Facial Expression Analysis using Nonlinear Decomposable Generative Models

Chan-Su Lee and Ahmed Elgammal

Computer Science, Rutgers University
Piscataway NJ 08854, USA
{chansu, elgammal}@cs.rutgers.edu

Abstract. We present a new framework to represent and analyze dynamic facial motions using a decomposable generative model. In this paper, we consider facial expressions which lie on a one dimensional closed manifold, i.e., start from some configuration and coming back to the same configuration, while there are other sources of variability such as different classes of expression, and different people, etc., all of which are needed to be parameterized. The learned model supports tasks such as facial expression recognition, person identification, and synthesis. We aim to learn a generative model that can generate different dynamic facial appearances for different people and for different expressions. Given a single image or a sequence of images, we can use the model to solve for the temporal embedding, expression type and person identification parameters. As a result we can directly infer intensity of facial expression, expression type, and person identity from the visual input. The model can successfully be used to recognize expressions performed by different people never seen during training. We show experiment results for applying the framework for simultaneous face and facial expression recognition.

Sub-categories: 1.1 Novel algorithms, 1.6 Others: modeling facial expression

1 Introduction

The appearance of a face performing a facial expression is an example of a dynamic appearance that has global and local deformations. There are two interesting components in dynamic facial expressions: face identity (face geometry and appearance characterizing the person) and facial motion (deformation of face geometry through the expression and its temporal characteristics). There has been extensive research related to face recognition [24] emanating from interest in applications in security and visual surveillance. Most of face recognition systems focused on still face images, i.e., capturing identity through facial geometry and appearance. There have been also interests on expression invariant face recognition [15, 14, 2]. Individual differences of facial expression like expressiveness can be useful as a biometric to enhance accuracy in face recognition [8]. On the other hand, facial expression analysis gain interest in computer vision with applications in human emotion analysis for HCI and affective computing. Most studies of facial expression recognition have focused on static display of intense expressions even though facial dynamics are important in interpreting facial expression precisely [1].

Our objective in this paper is to learn dynamic models for facial expressions that enable simultaneous recognition of faces and facial expressions. We learn a dynamic generative model that factors out different face appearance corresponding to different people and in the same time parameterizes different expressions.

Despite the high dimensionality of the image space in facial expressions, facial motions lie intrinsically on much lower dimensional subspaces. Therefore, researchers have tried to exploit subspace analysis in face recognition and facial expression analysis. PCA has been widely used in appearance modeling to discover subspaces for face appearance variations as in [21, 10]. When dealing with dynamic facial expressions, image data lie on low dimensional nonlinear manifolds embedded in the high dimensional input space. Embedding expression manifolds to low dimensional spaces provides a way to explicitly model such manifolds. Linear subspace analysis can achieve a linear embedding of the motion manifold in a subspace. However, the dimensionality of the subspace depends on the variations in the data and not on the intrinsic dimensionality of the manifold. Nonlinear dimensionality reduction approaches can achieve much lower dimensionality embedding of nonlinear manifolds through changing the metric from the original space to the embedding space based on local structure of the manifold, e.g. [17, 19]. Nonlinear dimensionality reduction has been recently exploited to model the manifold structure in face recognition, facial expression analysis [3]. However, all these approaches (linear and nonlinear) are data-driven, i.e., the visual input is used to model motion manifolds. The resulting embeddings are data-driven and, therefore, the resulting embedded manifolds vary due to person facial geometry, appearance, facial deformation, and dynamics in facial expressions, which affect collectively the appearance of facial expressions. The embedding of the same facial expression performed by different people will be quite different and it is hard to find a unified representation of the manifold. But, conceptually all these manifolds (for the same expression) are the same. We can think of it as the same expression manifold which is twisted differently in the input space based on person's facial appearance. They are all topologically equivalent, i.e., homeomorphic to each other and we can establish a bijection between any pair of them. Therefore, we utilize a conceptual manifold representation to model facial expression configuration and learn mappings between the conceptual unified representation and each individual data manifold.

Different factors affect the face appearance. There had been efforts to decompose multiple factors affecting appearance from face and facial expression data. Bilinear models were applied to decompose person-dependent factor and the pose-dependent factor as the style and content from pose-aligned face images of different people [20] and facial expression synthesis [4]. Multilinear analysis, or higher-order singular value decomposition [11], were applied to aligned face images with variation of people, illumination and expression factors and applied for face recognition [22]. In this model, face images are decomposed into tensor multiplication of different people basis, illumination basis and expression basis. Facial expressions were also analyzed using multilinear analysis for feature space similar to active appearance model to recognize face and facial expression simultaneously [23]. All these approaches have limitations in capturing nonlinearity of facial expression as the subspaces are expansion of linear subspace of facial images. In addition, all these approaches deal with static facial expressions and do not model dynamics in facial expression.

In this paper, we learn nonlinear mappings between a conceptual embedding space and facial expression image space and decompose the mapping space using multilinear analysis. The mapping between sequences of facial expression and embedding points contains characteristics of the data invariant to temporal variations and change with different people facial expression and different types of facial expressions. We decompose the mapping space into person face appearance factor, which is person dependent and consistent for each person, and expression factor, which depends on expression type and common to all people with the same expression. In addition, we explicitly decompose the intrinsic face configuration during the expression, as a function of time in the embedding space, from other conceptually orthogonal factors such as facial expressions and person face appearances. As a result, we learn a nonlinear generative model of facial expression with modeling dynamics in low dimensional embedding space and decomposing of multiple factors in facial expressions.

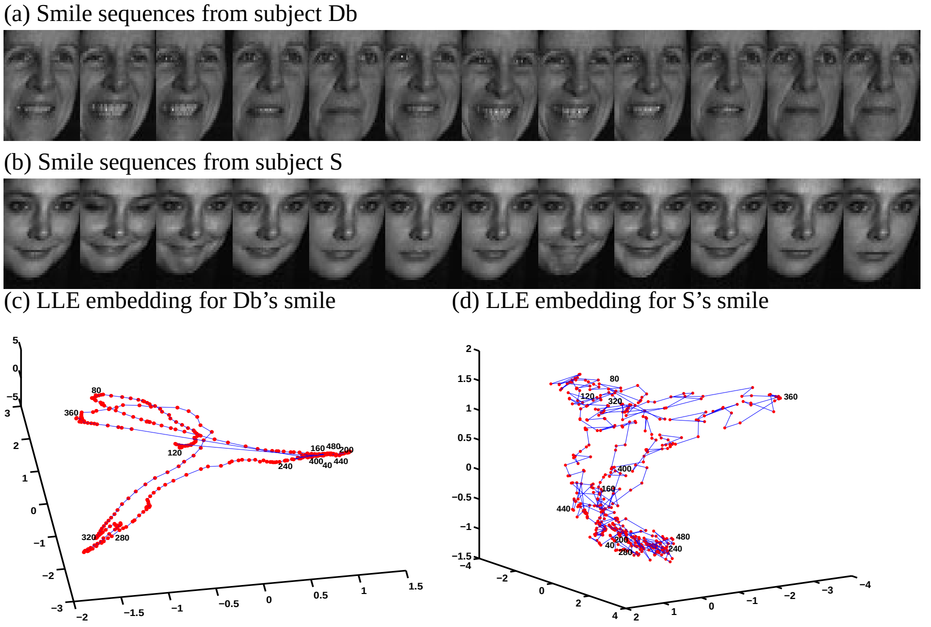
Contribution: In this paper we consider facial expressions which lie on a one dimensional closed manifold, i.e., start from some configuration and coming back to the same configuration. We introduce a framework to learn decomposable generative models for dynamic appearance of facial expressions where the motion is constrained to one dimensional closed manifolds while there are other sources of variability such as different classes of expression, and different people, etc., all of which are needed to be parameterized. The learned model supports tasks such as facial expression recognition, person identification, and synthesis. Given a single image or a sequence of images, we can use the model to solve for the temporal embedding, expression type and person identification parameters. As a result we can directly infer intensity of facial expression, expression type, and person face from the visual input. The model can successfully be used to recognize expressions performed by different people never seen in the training.

2 Facial Expression Manifolds and Nonlinear Decomposable Generative Models

We investigate low dimensional manifolds and propose conceptual manifold embedding as a representation of facial expression dynamics in Sec. 2.1. In order to preserve nonlinearity of facial expression in our generative model, we learn nonlinear mapping between embedding space and image space of facial expression in Sec. 2.2. The decomposable compact parameterization of the generative model is achieved using multilinear analysis of the mapping coefficients in Sec. 2.3.

2.1 Facial Expression Manifolds and Conceptual Manifold Embedding

We use conceptual manifold embedding for facial expressions as a uniform representation of facial expression manifolds. Conceptually, each expression sequence forms a one-dimensional closed trajectory in the input space as the expression starts from a neutral face and comes back to the neutral face. Data-driven low dimensional manifolds using nonlinear dimensionality reduction algorithms such as LLE [17] and Isomap [19]



(a) Smile sequences from subject Db

(b) Smile sequences from subject S

(c) LLE embedding for Db's smile

(d) LLE embedding for S's smile

Fig. 1. Facial expression manifolds in different subjects: (a) and (b): Facial expression image sequences. (2 cycles 480 frames):40th, 80th, 120th, 160th, 200th, 240th, 280th, 320th, 360th, 400th, 440th, 480th frames. (c) and (d): Nonlinear manifold embeddings of facial expression sequences by LLE.

vary in different people and in different expression types. Fig. 1 shows low dimensional manifold representation of facial expression sequences when we applied LLE to high dimensional vector representations of image sequences of facial expressions. The facial expression data with twice repetitions of the same type expression are captured and normalized for each person as shown in Fig. 1 (a) and (b). Fig. 1 (c) and (d) show the manifolds found by applying the LLE algorithm. The manifolds are elliptical curves with distortions according to the person face and expressions. Isomap and other nonlinear dimensionality reduction algorithms show similar results. Sometimes the manifold does not show smooth curves due to noise in the tracking data and images. In addition, the embedding manifolds can be very different in some case. It is hard to find representations comparable each manifold for multiple expression styles and expression types. Conceptually, however, all data driven manifolds are equal. They are all topologically equivalent, i.e., homeomorphic to each other, and to a circular curve. Therefore, we can use the unit circle in 2D space as a conceptual embedding space for facial expressions.

A set of image sequences which represent a full cycle of the facial expressions are used in conceptual embedding of facial expressions. Each image sequence is of a certain person with a certain expression. Each person has multiple expression image sequences. The image sequences are not necessarily to be of the same length. We denote each sequence by $\boldsymbol{Y}^{se} = \{\boldsymbol{y}_1^{se} \cdots \boldsymbol{y}_{N_{se}}^{se}\}$ where e denotes the expression label and s is person face label. Let N_e and N_s denote the number of expressions and the number of people respectively, i.e., there are $N_s \times N_e$ sequences. Each sequence is temporally embedded at equidistance on a unit circle such that $\boldsymbol{x}_i^{se} = [cos(2\pi i/N_{se}) \;\; sin(2\pi i/N_{se})], i = 1 \cdots N_{se}$. Notice that by temporal embedding on a unit circle we do not preserve the metric in input space. Rather, we preserve the topology of the manifold.

2.2 Nonlinear Mapping between Embedding Space and Image Space

Nonlinear mapping between embedding space and image space can be achieved through raidal basis function interpolation [6]. Given a set of distinctive representative and arbitrary points $\{\boldsymbol{z}_i \in \mathbb{R}^2, i = 1 \cdots N\}$ we can define an empirical kernel map[18] as $\psi_N(\boldsymbol{x}) : \mathbb{R}^2 \rightarrow \mathbb{R}^N$ where

$$\psi_N(\boldsymbol{x}) = [\phi(\boldsymbol{x}, \boldsymbol{z}_1), \cdots, \phi(\boldsymbol{x}, \boldsymbol{z}_N)]^{\mathsf{T}}, \tag{1}$$

given a kernel function $\phi(\cdot)$. For each input sequence $\boldsymbol{Y}^{se}$ and its embedding $\boldsymbol{X}^{se}$ we can learn a nonlinear mapping function $f^{se}(\boldsymbol{x})$ that satisfies $f^{se}(\boldsymbol{x}_i) = \boldsymbol{y}_i, i = 1 \cdots N_{se}$ and minimizes a regularized risk criteria. Such function admits a representation of the form

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} w_i \phi(\boldsymbol{x}, \boldsymbol{z}_i),$$

i.e., the whole mapping can be written as

$$f^{se}(\boldsymbol{x}) = \boldsymbol{B}^{se} \cdot \psi(\boldsymbol{x}) \tag{2}$$

where $\boldsymbol{B}$ is a $d \times N$ coefficient matrix. If radial symmetric kernel function is used, we can think of equation 2 as a typical Generalized Radial basis function (GRBF) interpolation [16] where each row in the matrix $\boldsymbol{B}$ represents the interpolation coefficients for corresponding element in the input. i.e., we have d simultaneous interpolation functions each from 2D to 1D. The mapping coefficients can be obtained by solving the linear system

$$[\boldsymbol{y}_1^{se} \cdots \boldsymbol{y}_{N_{se}}^{se}] = \boldsymbol{B}^{se} [\psi(\boldsymbol{x}_1^{se}) \cdots \psi(\boldsymbol{x}_{N_{se}}^{se})]$$

Where the left hand side is a $d \times N_{se}$ matrix formed by stacking the images of sequence se column wise and the right hand side matrix is an $N \times N_{se}$ matrix formed by stacking kernel mapped vectors. Using these nonlinear mapping, we can capture nonlinearity of facial expression in different people and expressions. More details about fitting the model can be found in [6].

2.3 Decomposition of Nonlinear Mapping Space

Each nonlinear mapping is affected by multiple factors such as expressions and person faces. Mapping coefficients can be arranged into high order tensor according to expression type and person face. We applied multilinear tensor analysis to decompose the mapping into multiple orthogonal factors. This is a generalization of the nonlinear style and content decomposition as introduced in [7]. Multilinear analysis can be achieved by higher-order singular value decomposition (HOSVD) with *unfolding*, which is a generalization of singular value decomposition (SVD) [11]. Each of the coefficient matrices $\boldsymbol{B}^{se} = [\boldsymbol{b}_1 \boldsymbol{b}_2 \cdots \boldsymbol{b}_N]$ can be represented as a coefficient vector $\boldsymbol{b}^{se}$ by column stacking (stacking its columns above each other to form a vector). Therefore, $\boldsymbol{b}^{se}$ is an $N_c = d{\cdot}N$ dimensional vector. All the coefficient vectors can then be arranged in an order-three facial expression coefficient tensor $\mathcal{B}$ with dimensionality $N_s \times N_e \times N_c$. The coefficient tensor is then decomposed as

$$\mathcal{B} = \mathcal{Z} \times_1 \boldsymbol{S} \times_2 \boldsymbol{E} \times_3 \boldsymbol{F} \tag{3}$$

where $\boldsymbol{S}$ is the mode-1 basis of $\mathcal{B}$, which represents the orthogonal basis for the person face. Similarly, $\boldsymbol{E}$ is the mode-2 basis representing the orthogonal basis of the expression and $\boldsymbol{F}$ represents the basis for the mapping coefficient space. The dimensionality of these matrices are $N_s \times N_s$, $N_e \times N_e$, $N_c \times N_c$ for $\boldsymbol{S}$,$\boldsymbol{E}$ and $\boldsymbol{F}$ respectively. $\mathcal{Z}$ is a core tensor, with dimensionality $N_s \times N_e \times N_c$ which governs the interactions among different mode basis matrices. Similar to PCA, it is desired to reduce the dimensionality for each of the orthogonal spaces to retain a subspace representation. This can be achieved by applying higher-order orthogonal iteration for dimensionality reduction [12].

Given this decomposition and given any N_s dimensional person face vector $\boldsymbol{s}$ and any N_e dimensional expression vector $\boldsymbol{e}$ we can generate coefficient matrix $\boldsymbol{B}^{se}$ by unstacking the vector $\boldsymbol{b}^{se}$ obtained by tensor product $\boldsymbol{b}^{se} = \mathcal{Z} \times_1 \boldsymbol{s} \times_2 \boldsymbol{e}$. Therefore we can generate any specific instant of the expression by specifying the configuration parameter $\boldsymbol{x}_t$ through the kernel map defined in equation 1. Therefore, the whole model for generating image $\boldsymbol{y}_t^{se}$ can be expressed as

$$\boldsymbol{y}_t^{se} = \textit{unstacking}(\mathcal{Z} \times_1 \boldsymbol{s} \times_2 \boldsymbol{e}) \cdot \psi(\boldsymbol{x}_t) \ .$$

This can be expressed abstractly also in the generative form by arranging the tensor $\mathcal{Z}$ into a order-four tensor $\mathcal{C}$

$$\boldsymbol{y}_t = \mathcal{C} \times_1 \boldsymbol{s} \times_2 \boldsymbol{e} \times_3 \psi(\boldsymbol{x}) \times_4 \boldsymbol{L} \ , \tag{4}$$

where dimensionality of core tensor $\mathcal{C}$ is $N_s \times N_e \times N \times d$, $\psi(\boldsymbol{x})$ is a basis vector for kernel mapping with dimension N for given $\boldsymbol{x}$ and $\boldsymbol{L}$ is collection of basis vectors of all pixel elements with dimension $d \times d$. We can analyze facial expression image sequence by estimation of the parameters in this generative model.

3 Facial Expression Analysis and Synthesis using Generative Models

There are two main approaches in representing facial motions for facial expression analysis: model-based or appearance-based. Geometric features are extracted with the aid of 2D or 3D face models in model-based approaches. 3D deformable generic face model [5] or multistate facial component models [13] are used to extract facial features. Active appearance model are employed to use both shape and textures in [10][23]. Our generative model use pixel intensity itself as an appearance representation as we want,

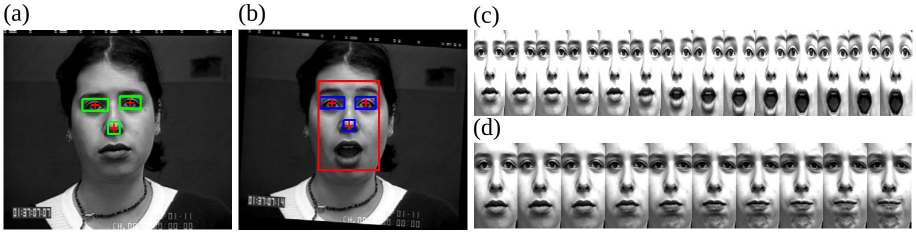


Fig. 2. Cropping and normalizing face images to a standard front face: (a) Selected templates (eyes and a nose tip). (b) Template detection, affine transformation and selected cropping region. (c) A normalized sequence where templates are selected from the first frame. (d) A normalized sequence from another expression of the same person.

not only to analyze, but also to synthesize facial expressions in the image space. The final representation of facial expressions in our generative model, however, is a compact person face vector and an expression vector that are invariant to temporal characteristics and low dimensional embedding that represents temporal characteristics.

The generative model supports both sequence-based and frame-based recognition of facial expressions. Facial expression recognition system can be categorized into frame-based and sequence-based methods [8] according to the use of temporal information. In frame-based methods, the input image is treated independently either a static image or a frame of a sequence. Frame-based method does not use temporal information in the recognition process. In sequence-based methods, the HMMs are frequently used to utilize temporal information in facial expression recognition [5]. In our generative model, the temporal characteristics are modeled in low dimensional conceptual manifolds and we can utilize the temporal characteristics of the whole sequence by analyzing facial expression based on the mapping between the low dimensional embedding and the whole image sequence. We also provide methods to estimate expression parameters and face parameters from single static image.

3.1 Preprocessing: Cropping and normalizing face images

The alignment and normalization of captured faces using a standard face is an important preprocessing in facial expression recognition to achieve robust recognition of facial expressions in head motion and lighting condition change. We interactively select two eyes and a nose tip locations, which are relatively consistent during facial expressions, from one face image for each subject. Based on the selected templates, we perform detection of each template location from subsequent facial image by finding maximum correlation of the template images in the given frame image. We cropped images based on eye locations and nose similar to [14] after affine transformation to align the location of eyes and nose tip to a standard front face. Fig. 2 (a) shows interactively selected two eyes and a nose tip templates. A cropping region is decided after detection of template locations and affine transformation for every new image as shown in (b). Fig. 2 (c) shows normalization results in the sequence where the first frame is used to select templates and (d) in another sequence with different expression of the same subject

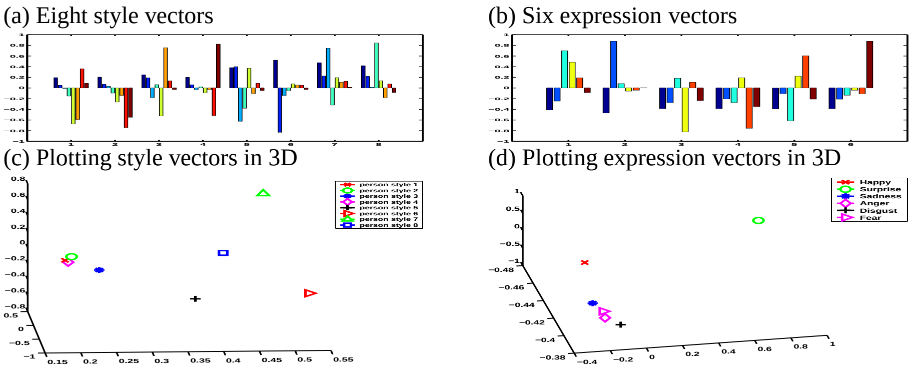

(a) Eight style vectors
(b) Six expression vectors
(c) Plotting style vectors in 3D
(d) Plotting expression vectors in 3D

Fig. 3. Facial expression analysis for eight subjects with six expressions from Cohn-Kanade dataset

without new template selections. We further processed the normalization of brightness when necessary. As a result, we can recognize facial expression robustly with changes of head location and small changes of head orientation from a frontal view.

3.2 Facial Expression Representation

Our generative model represents facial expressions using three state variables of the generative model: person face vector $\boldsymbol{s}$, expression vector $\boldsymbol{e}$, and embedding manifold point $\boldsymbol{x}$, whose dimensions are N_s, N_e and 2 without further dimensionality reduction using orthogonal iteration. The embedding can be parameterized by one dimensional vector as the conceptual embedding manifold, unit circle, is one dimensional manifold in two dimensional space. The total number of dimensions of the parameters to represent a facial image is $N_s + N_e + 1$ after we learn the generative model. Fig. 3 shows examples of person face vectors (a) and expression vectors (b) when we learn the generative model from eight people with six different expressions related to basic emotions from Cohn-Kanade AU coded facial expression database [9], where $N_s = 8$ and $N_e = 6$. Plottings in three dimensional space using the first three parameters of face class vectors (c) and facial expression class vectors (d) give insight to the similarity among different person faces and different facial expression classes. Interestingly, plotting of the first three parameters of six basic expressions in Fig. 3 (d) shows embedding similar to the conceptual distance of six expressions in the image space. The surprise expression class vector is located far from other expressions, which is connected to distinguishable different visual motions in surprise. Anger, fear, disgust, and sadness are relatively close to each other than other expressions since they are distinguished visually using more subtle motions. The expression vector captures characteristics of image space facial expression in low dimensional space.

3.3 Sequence-based Facial Expression Recognition

Given a sequence of images representing a facial expression, we can solve for the expression class paramter, $\boldsymbol{e}$, and person face parameter, $\boldsymbol{s}$. First, the sequence is embed-

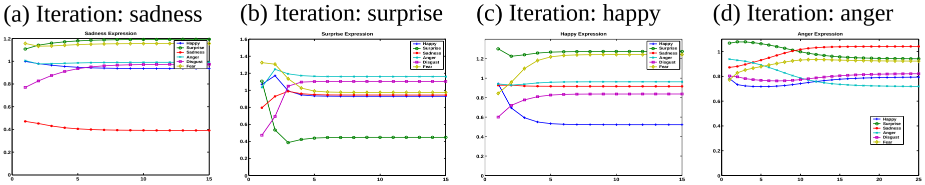

(a) Iteration: sadness (b) Iteration: surprise (c) Iteration: happy (d) Iteration: anger

Fig. 4. The convergence of estimated expression parameters in iterations

ded to a unit circle and aligned to the model as described in Sec. 2. Then, mapping coefficients $\boldsymbol{B}$ are learned from the aligned embedding to the input. Given such coefficients, we need to find the optimal $\boldsymbol{s}$ and $\boldsymbol{e}$, which minimize the error

$$E(s,e) = \|\boldsymbol{b} - \mathcal{Z} \times_1 \boldsymbol{s} \times_2 \boldsymbol{e}\| \ , \tag{5}$$

where $\boldsymbol{b}$ is the vector representation of matrix $\boldsymbol{B}$ by column stacking. If the person face parameter $\boldsymbol{s}$ is known, we can obtain a closed form solution for $\boldsymbol{e}$. This can be achieved by evaluating the tensor product $\mathcal{G} = \mathcal{Z} \times_1 \boldsymbol{s}$ to obtain tensor $\mathcal{G}$. Solution for $\boldsymbol{b}$ can be obtained by solving the system $\boldsymbol{b} = \mathcal{G} \times_2 \boldsymbol{e}$ for $\boldsymbol{e}$ which can be written as a typical linear system by unfolding $\mathcal{G}$ as a matrix. Therefore the expression estimation $\boldsymbol{e}$ can be obtained by

$$e = (\mathcal{G}_2)^+ \boldsymbol{b} \tag{6}$$

where $\mathcal{G}_2$ is the matrix obtained by mode-2 unfolding of $\mathcal{G}$ and $+$ denotes the pseudo inverse using singular value decomposition (SVD). Similarly we can analytically solve for $\boldsymbol{s}$ if the expression parameter, $\boldsymbol{e}$, is known by forming a tensor $\mathcal{H} = \mathcal{Z} \times_2 \boldsymbol{e}$:

$$\boldsymbol{s} = (\mathcal{H}_1)^+ \boldsymbol{b} \tag{7}$$

where $\mathcal{H}_1$ is the matrix obtained by mode-1 unfolding of $\mathcal{H}$

Iterative estimations of $\boldsymbol{e}$ and $\boldsymbol{s}$ using equations 6 and 7 would lead to a local minima for the error in 5. Fig. 4 shows examples of expression estimation in iteration using new sequences. Y axis shows Euclidian distance between the estimated expression vector and six expression class vectors in the generative model in Sec. 4.1. Usually the estimation parameters of expressions converge into one of expression class vectors within several iterations. Fig. 4 (d) shows a case when more than ten iterations are required to reach stable solution in the estimation of expression vector.

3.4 Frame-based Facial Expression Recognition

When the input is a single face image, it is desired to estimate temporal embedding or the face configuration in addition to expression and person face parameters in the generative model. Given an input image $\boldsymbol{y}$, we need to estimate configuration, $\boldsymbol{x}$, expression parameter $\boldsymbol{e}$, and person face parameter $\boldsymbol{s}$ which minimize the reconstruction error

$$E(\boldsymbol{x},\boldsymbol{s},\boldsymbol{e}) = || \ \boldsymbol{y} - \mathcal{C} \times_1 \boldsymbol{s} \times_2 \boldsymbol{e} \times_3 \psi(\boldsymbol{x}) \ || \tag{8}$$

We can use a robust error metric instead of Euclidian distance in error measurements. In both cases we end up with a nonlinear optimization problem.

We assume optimal estimated expression parameter for a given image can be written as a linear combination of expression class vectors in the training data. i.e., we need to solve for linear regression weights α such that $\boldsymbol{e} = \sum_{k=1}^{K_e} \alpha_k \boldsymbol{e}^k$ where each $\boldsymbol{e}^k$ is one of K_e expression class vectors in the training data. Similarly for the person face, we need to solve for weights β such that $\boldsymbol{s} = \sum_{k=1}^{K_s} \beta_k \boldsymbol{s}^k$ where each $\boldsymbol{s}^k$ is one of K_s face class vectors.

If the expression vector and the person face vector are known, then equation 8 is reduced to a nonlinear 1-dimensional search problem for configuration x on the unit circle that minimizes the error. On the other hand, if the configuration vector and the person face vector are known, we can obtain expression conditional class probabilities $p(\boldsymbol{e}^k|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{s})$ which is proportional to observation likelihood $p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{s}, \boldsymbol{e}^k)$. Such likelihood can be estimated assuming a Gaussian density centered around $\mathcal{C} \times_1 \boldsymbol{s}^k \times_2 \boldsymbol{e} \times_3 \psi(\boldsymbol{x})$, i.e.,

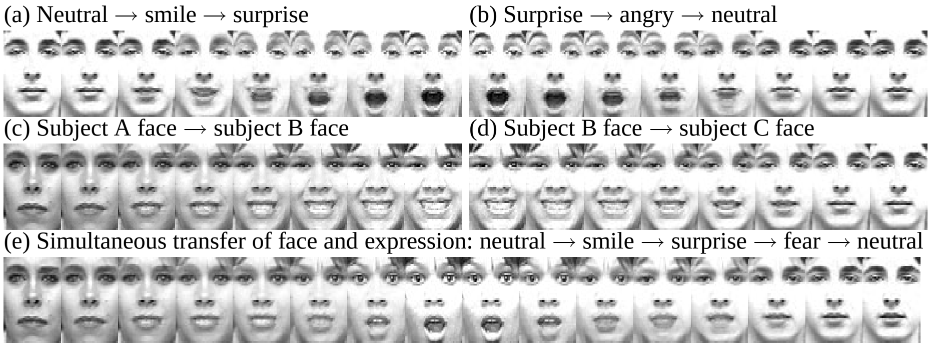
$$p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{s}, \boldsymbol{e}^k) \approx N(\mathcal{C} \times_1 \boldsymbol{s}^k \times_2 \boldsymbol{e} \times_3 \psi(x), \Sigma^{\boldsymbol{e}^k}).$$

Given expression class probabilities we can set the weights to $\alpha_k = p(\boldsymbol{e}^k \mid \boldsymbol{y}, \boldsymbol{x}, \boldsymbol{s})$. Similarly, if the configuration vector and the expression vector are known, we can obtain face class weights by evaluating image likelihood given each face class $\boldsymbol{s}^k$ assuming a Gaussian density centered at $\mathcal{C} \times_1 \boldsymbol{s}^k \times_2 \boldsymbol{e} \times_3 \psi(\boldsymbol{x})$.

This setting favors an iterative procedures for solving for $\boldsymbol{x}, \boldsymbol{e}, \boldsymbol{s}$. However, wrong estimation of any of the vectors would lead to wrong estimation of the others and leads to a local minima. For example wrong estimation of the expression vector would lead to a totally wrong estimate of configuration parameter and therefore wrong estimate for person face parameter. To avoid this we use a deterministic annealing like procedure where in the beginning the expression weights and person face weights are forced to be close to uniform weights to avoid hard decisions about expression and face classes. The weights gradually become discriminative thereafter. To achieve this, we use a variable expression and person face class variances which are uniform to all classes and are defined as $\boldsymbol{\Sigma}^{\boldsymbol{e}} = T_e \sigma_e^2 \boldsymbol{I}$ and $\boldsymbol{\Sigma}^{\boldsymbol{s}} = T_s \sigma_s^2 \boldsymbol{I}$ respectively. The parameters T_e and T_s start with large values and are gradually reduced and in each step and a new configuration estimate is computed. Several iterations with decreasing T_e and T_s allow estimations of the expression vector, the person face vector and face configuration iteratively and allow estimations of expression and face from a single image.

3.5 Facial Expression Synthesis

Our model can generate new facial expressions by combinations of new facial expression parameter and person face parameter. As we have decomposed the mapping space that captures nonlinear deformation in facial expressions, the linear interpolation of the face style and facial expression still somewhat captures nonlinearity in the facial expression. In addition, we can control the parameters for person face and facial expression separately as a result of the multilinear decomposition. A new person face vector and a new facial expression vector can be synthesized by linear interpolation of existing


(a) Neutral → smile → surprise

(b) Surprise → angry → neutral

(c) Subject A face → subject B face

(d) Subject B face → subject C face

(e) Simultaneous transfer of face and expression: neutral → smile → surprise → fear → neutral

Fig. 5. Facial expression synthesis: First row: Expression transfer. Second row: Person face transfer during smile expression. Third row: simultaneous transfer of facial expression and person face

person face class vectors and expression class vectors using parameter α_i, and β_j as follows:

$$\boldsymbol{e}^{new} = \alpha_1 \boldsymbol{e}_1 + \alpha_2 \boldsymbol{e}_2 + \cdots + \alpha_{N_e} \boldsymbol{e}_{N_e} \ \ , \boldsymbol{s}^{new} = \beta_1 \boldsymbol{s}_1 + \beta_2 \boldsymbol{s}_2 + \cdots + \beta_{N_s} \boldsymbol{s}_{N_s} \ , \tag{9}$$

where $\sum_i \alpha_i = 1$, and $\sum_j \beta_j = 1$, and $\alpha_i \geq 0$ and $\beta_i \geq 0$ in order to be linear interpolation in the convex set of the original expression classes and face classes. Here α_i and β_j are control parameters whereas they are estimated in recognition as in Sec. 3.4. We can also control these interpolation parameters according to temporal information or configuration. A new facial expression image can be generated using new style and expression parameters.

$$\boldsymbol{y}_t^{new} = \mathcal{C} \times_1 \boldsymbol{s}_t^{new} \times_2 \boldsymbol{e}_t^{new} \times_3 \psi(\boldsymbol{x}_t) \tag{10}$$

Fig. 5 shows examples of the synthesis of new facial expressions and person faces. During synthesis of the new images, we combine control parameter t to embedding coordinate x and interpolation parameter α and β. In case of Fig. 5 (a), the t changed $0 \rightarrow 1$ and new expression parameter $\boldsymbol{e}_t^{new} = (1-t)\boldsymbol{e}^{smile} + t\boldsymbol{e}^{surprise}$. As a result, the facial expression starts from neutral expression of smile and animates new expression as t changes and when $t = 1$, the expression become a peak expression of surprise. In case of (b), the t changed $1 \rightarrow 0$. In the same way, we can synthesize new faces during smile expressions as in (c) and (d). Fig. 5 (e) is the simultaneous control of the person face and expression parameters. This shows the potential of synthesis of new facial expression in the image space using our generative model.

4 Experimental Results

4.1 Person independent recognition of facial expression: Cohn-Kanade facial expression data set

We test the performance of facial expression analysis by our generative model using Cohn-Kanade AU coded facial expression database [9]. We first collected eight subjects with all six basic expression sequences, which are 48 expression sequences whose frame number varies between 11 and 33 to target display. We performed normalization by cropping image sequence based on template eyes and nose images as explained in Sec. 3.1. We embed the sequence into a half circle in the conceptual manifold as we counted the sequence of the data as half of one cycle among neutral $\rightarrow$ target expression $\rightarrow$ neutral expression. Eight equal-distance centers are used in learning GRBF with thin-plate spline basis. We used a full dimension to represent each style and expression. Fig. 3 shows the representation of expression vectors and person face vectors after learning the generative models from these eight subjects with six expressions. Fig. 5 shows examples of facial expression synthesis using this generative model.

Sequence-based expression recognition: The performance of person independent facial expression recognition is tested by leave-one-out cross-validation method using whole sequences in the database [9]. We learned a generative model using 42 sequences of seven subjectsand and tested six sequences of one subject whose data are not used for learning the generative model. We tested the recognition performance by selecting the nearest expression class vector after iterations by sequence-based expression recognition in Sec. 3.3. Table 1 shows the confusion matrix for 48 sequences. The result shows potentials of the estimated expression vectors as feature vectors for other advanced classifiers like SVM.

Table 1. Person-independent average confusion matrix by sequence-based expression recognition

Emotion	Happy	Surprise	Sadness	Anger	Disgust	Fear
Happy	25%(2)	0	0	37.5%(3)	25%(2)	25%(2)
Surprise	12.5%(1)	62.5%(5)	12.5%(1)	0	0	12.5%(1)
Sadness	0	0	37.5%(3)	25%(2)	12.5%(1)	25%(2)
Anger	12.5%(1)	0	37.5%(3)	50%(4)	0	0
Disgust	12.5%(1)	12.5%(1)	12.5%(1)	25%(2)	12.5%(1)	25%(2)
Fear	0	0	0	50%(4)	0	50%(4)

Frame-based expression recognition: Using the generative model, we can estimate person face parameters and expression parameters for a given expression image or sequence of images based on frame-by-frame estimation. We collected additional data that have five different expressions from 16 subjects. We used the generative model learned by eight subjects with six expressions to estimate expression parameters and person face parameters using deterministic annealing in Sec. 3.4. Fig. 6 (a) (b) (c) shows expression weight values α of every frame in three different expression sequences. The weights become more discriminative as expressions get closer to target expressions. We can recognize the expression using the maximum weight expression class in every frame. Table 2 shows recognition results when we classified facial expression using maximum expression weight of the last frame from 80 sequences.

4.2 Dynamic facial expression and face recognition

We used CMU-AMP facial expression database which are used for robust face recognition in variant facial expressions [14]. We collected sequences of ten people with three

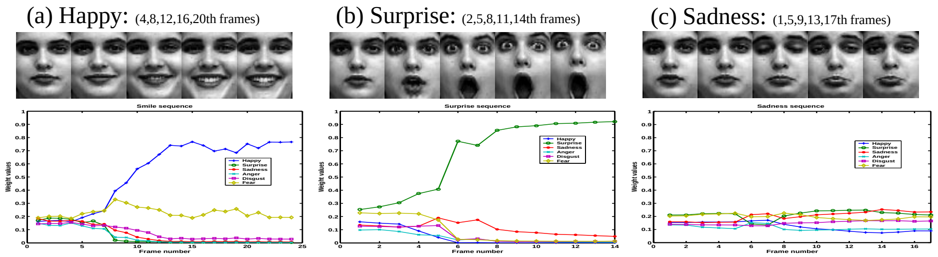

(a) Happy: (4,8,12,16,20th frames)
(b) Surprise: (2,5,8,11,14th frames)
(c) Sadness: (1,5,9,13,17th frames)

Fig. 6. Estimated expression weights in frame-based estimations

Table 2. Person-independent average confusion matrix by frame-based recognition: classification only last frame maximum weight expression

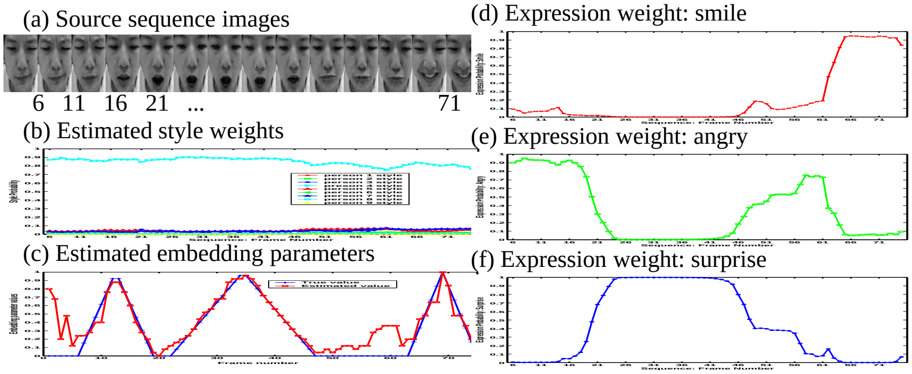
Emotion	Happy	Surprise	Sadness	Anger	Disgust	Fear
Happy	93.3%(14)	0	0	0	0	6.7%(1)
Surprise	0	100%(16)	0	0	0	0
Sadness	0	7.1%(1)	28.6%(4)	7.1%(1)	35.7%(5)	21.4%(3)
Anger	9.1%(1)	0	18.2%(2)	27.3%(3)	45.4%	0
Disgust	9.1%(1)	0	9.1%(1)	18.2%(2)	63.6%(7)	0
Fear	25%(3)	0	8.3%(1)	0	8.3%(1)	58.3%(7)

expressions (smile, anger, surprise) by manual segmentation from the whole sequences. We learned a generative model from nine people. The last one person data are used to test recognition of expression as a new person. The unit circle is used to embed each expression sequence.

We used the learned generative model to recognize facial expression, and person identity at each frame from the whole sequence using the frame-based algorithm in section 3.4. Fig. 7 (a) shows example frames of a whole sequence and the three different expression probabilities obtained in each frame (d)(e)(f). The person face weights, which are used to person identification, consistantly show dominant weights for the subject face as in Fig. 7 (b). Fig. 7 (c) shows that the estimated embedding parameters are close to the true embedding from manually selected sequences. We used the learned model to recognize facial expressions from sequences of a new person whose data are not used during training. Fig. 8 shows recognition of expressions for the new person. The model can generalize for the new person and can distinguish three expressions from the whole sequence.

5 Conclusion

In this paper we presented a framework for learning a decomposable generative model for facial expression analysis. Conceptual manifold embedding on a unit circle is used to model the intrinsic facial expression configuration on a closed 1D manifold. The embedding allows modeling any variations (twists) of the manifold given any number factors such as different people, different expression, etc; since all resulting manifolds are still topologically equivalent to the unit circle. This is not achievable if data-driven


(a) Source sequence images

(b) Estimated style weights

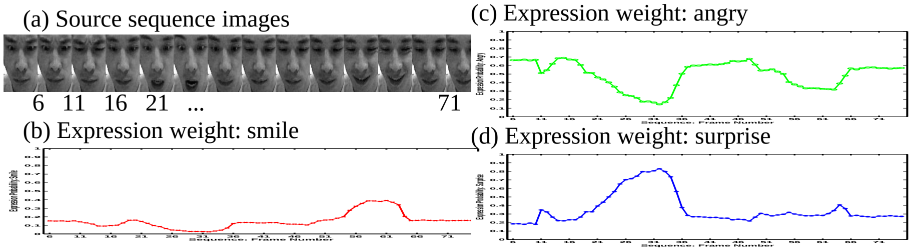
(c) Estimated embedding parameters

(d) Expression weight: smile

(e) Expression weight: angry

(f) Expression weight: surprise

Fig. 7. Facial expression analysis with partially trained segements


(a) Source sequence images

(b) Expression weight: smile

(c) Expression weight: angry

(d) Expression weight: surprise

Fig. 8. Expression recognition for a new person.

embedding is used. The use of a generative model is tied to the use of conceptual embedding since the mapping from the manifold representation to the input space will be well defined in contrast to a discriminative model where the mapping from the visual input to manifold representation is not necessarily a function. We introduced a framework to solve facial expression factors, person face factors and configurations in iterative methods for the whole sequence and in deterministic annealing methods for a given frame. The estimated expression parameters can be used as feature vectors for expression recognition using advanced classification algorithms like SVM. The frame-by-frame estimation of facial expression shows similar weights when expression image is close to the neutral face and more discriminative weights when it is near to target facial expressions. The weights of facial expression may be useful not only for facial expression recognition but also for other characteristics like expressiveness in the expression.

References

1. Z. Ambadar, J. W. Schooler, and J. F. Cohn. Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions. *Psychological Science*, 16(5):403–410, 2005.
2. A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Expression-invariant 3d face recognition. In *AVBPA, LNCS 2688*, pages 62–70, 2003.

3. Y. Chang, C. Hu, and M. Turk. Probabilistic expression analysis on manifolds. In *Proc. of CVPR*, pages 520–527, 2004.
4. E. S. Chuang, H. Deshpande, and C. Bregler. Facial expression space learning. In *Pacific Conference on Computer Graphics and Applications*, pages 68–76, 2002.
5. I. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang. Facial expression recognition from video sequences: Temporal and static modeling. *CVIU*, pages 160–187, 2003.
6. A. Elgammal. Nonlinear manifold learning for dynamic shape and dynamic appearance. In *Workshop Proc. of GMBV*, 2004.
7. A. Elgammal and C.-S. Lee. Separating style and content on a nonlinear manifold. In *Proc. of CVPR*, volume 1, pages 478–485, 2004.
8. A. K. Jain and S. Z. Li, editors. *Handbook of Face Recognition*, chapter 11. Face Expression Analysis. Springer, 2005.
9. T. Kanade, Y. Tian, and J. F. Cohn. Comprehensive database for facial expression analysis. In *Proc. of FGR*, pages 46–53, 2000.
10. A. Lanitis, C. J. Taylor, and T. F. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Trans. PAMI*, 19(7):743–756, 1997.
11. L. D. Lathauwer, B. de Moor, and J. Vandewalle. A multilinear singular value decomposiiton. *SIAM Journal On Matrix Analysis and Applications*, 21(4):1253–1278, 2000.
12. L. D. Lathauwer, B. de Moor, and J. Vandewalle. On the best rank-1 and rank-(r1, r2, ..., rn) approximation of higher-order tensors. *SIAM Journal On Matrix Analysis and Applications*, 21(4):1324–1342, 2000.
13. Y. li Tian, T. Kanade, and J. F. Cohn. Recognizing action units for facial expression analysis. *IEEE Trans. PAMI*, 23(2), 2001.
14. X. Liu, T. Chen, and B. V. Kumar. Face authentication for multiple subjects using eigenflow. *Pattern Recognitioin*, 36:313–328, 2003.
15. A. M. Martinez. Recognizing expression variant faces from a single sample image per class. In *Proc. of CVPR*, pages 353–358, 2003.
16. T. Poggio and F. Girosi. Networks for approximation and learning. *Proceedings of the IEEE*, 78(9):1481–1497, 1990.
17. S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
18. B. Schlkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.
19. J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.
20. J. B. Tenenbaum and W. T. Freeman. Separating style and content with biliear models. *Neural Computation*, 12:1247–1283, 2000.
21. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
22. M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *7th European Conference on Computer Vision*, pages 447–460, 2002.
23. H. Wang and N. Ahuja. Facial expression decomposition. In *Proc. of ICCV*, volume 2, pages 958–965, 2003.
24. W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, 2003.