Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 24 (2006) 605–614
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Manifold based analysis of facial expression

Ya Chang [a,*], Changbo Hu [b], Rogerio Feris [a], Matthew Turk [b]

[a] Computer Science Department, University of California, Santa Barbara, CA 93106, USA
[b] Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Received 14 December 2004; received in revised form 30 June 2005; accepted 23 August 2005

Abstract

We propose a novel approach for modeling, tracking, and recognizing facial expressions on a low-dimensional expression manifold. A modified Lipschitz embedding is developed to embed aligned facial features in a low-dimensional space, while keeping the main structure of the manifold. In the embedded space, a complete expression sequence becomes a path on the expression manifold, emanating from a center that corresponds to the neutral expression. As an offline training stage, facial contour features are first clustered in this space, using a mixture model. For each cluster in the low-dimensional space, a specific ASM model is learned, in order to avoid incorrect matching due to non-linear image variations. A probabilistic model of transitions between the clusters and paths in the embedded space is then learned. Given a new expression sequence, we use ICondensation to track facial features, while recognizing facial expressions simultaneously, within the common probabilistic framework. Experimental results demonstrate that our probabilistic facial expression model on the manifold significantly improves facial deformation tracking and expression recognition. We also synthesize image sequences of changing expressions through the manifold model.

1. Introduction

Facial expression is one of the most powerful ways that people coordinate conversation and communicate emotions and other mental, social, and physiological cues. Computational facial expression analysis is an active and challenging research topic in computer vision, impacting important applications in areas such as human–computer interaction and data-driven animation.

Facial expressions can be classified in various ways—in terms of non-prototypic expressions such as 'raised brows.' prototypic expressions such as emotional labels (e.g. 'happy'), or facial actions such as the action units defined in facial action coding system (FACS) [1]. Some psychologists claim that there are six kinds of universally recognized facial expressions: happiness, sadness, fear, anger, disgust, and surprise [2]. Existing expression analyzers [3–5] usually classify the examined expression into one of the basic emotion categories. These six basic categories are only a small subset of all facial expressions expressible by the human face. For 'blended' expressions, it may be more reasonable to classify them quantitatively into multiple emotion categories. Considering the intensity scale of the different facial expressions, each person has his/her own maximal intensity of displaying a particular facial action. It is useful to recognize the temporal intensity change of expressions in videos. Some surveys [6,7] gave a detailed review of existing methods on facial expression analysis and recognition.

A key challenge in automatic facial expression analysis is to identify a global representation for all possible facial expressions that affords semantic analysis. In this paper, we explore the space of expression images and propose the manifold of expressions as a foundation for expression analysis, using non-linear dimensionality reduction to embed facial deformations in a low-dimensional space. Non-linear dimensionality reduction has attracted attention for a long time in computer vision and visualization research [8,9]. Images lie in a very high-dimensional space, but a class of images generated by latent variables lies on a manifold in this space. For human face images, the latent variables may be the illumination, identity, pose and facial deformations.

An N-dimensional representation of the face (where N could be the number of pixels in the image or the number of parameters in a face model, for example) can be considered a point in an N-dimensional face space, and the variability of facial expression can be represented as low-dimensional manifolds embedded in this space. People change facial expressions continuously over time. Thus all images of an individual's facial expressions represent a smooth manifold in

* Corresponding author.
E-mail address: yachang@cs.ucsb.edu (Y. Chang).

0262-8856/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2005.08.006

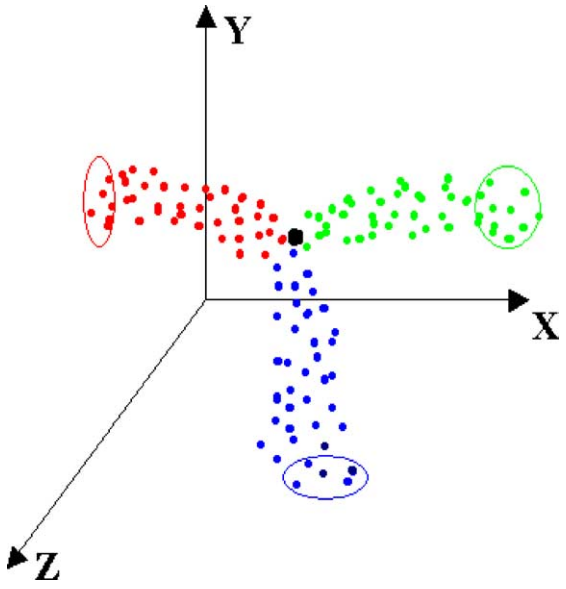

Fig. 1. Illustration of a 3D expression manifold. The reference center is defined by the neutral face. Image sequences from three different expressions are shown. The further a point is away from the reference point, the higher is the intensity of that expression.

the *N*-dimensional face space with the 'neutral' face as the central reference point. The intrinsic dimension of this manifold is much lower than *N*.

On the manifold of expressions, similar expressions are points in the local neighborhood on the manifold. Sequences of basic emotional expressions become paths on the manifold extended from the reference center, as illustrated in Fig. 1. The blends of expressions lie between those paths, so they can be defined analytically by the positions of the basic paths. The analysis of the relationships between different facial expressions is facilitated on the manifold.

It is a formidable task to learn the complete structure of the manifold of expressions in a high-dimensional image space. To overcome this problem, our core idea is to embed the non-linear manifold in a low-dimensional space and recognize facial expression from video sequences probabilistically. Fig. 2 illustrates the overall structure of the system.

Non-linear embedding methods such as ISOMAP [10], local linear embedding (LLE) [11], charting a manifold [12], and global coordinate of local linear models [13] are promising in handling high-dimensional non-linear data. Recently, researchers have applied manifold methods to face recognition [14–16] and facial expression representation [17–19].

Rather than working in the image space (which is very sensitive to illumination changes), we describe the face as a set of points along facial feature contours, as shown in Fig. 3. We applied a modified Lipschitz embedding [20,21] to embed the face contour representation in the high-dimensional space into a low-dimensional space, while keeping the main structure of the manifold. Lipschitz embedding leads to good preservation of clusters in practical cases [22,23].

After the embedding, the expression sequences in the gallery become paths emanating from the center, which is defined by the neutral expression. In an offline training stage, a Gaussian mixture model is applied to cluster data in the low-dimensional expression space. For each cluster, a specific ASM model is learned to allow more robustness with respect to non-linear image variations. We learn the probabilistic model of transition between those paths from the gallery videos.

Given a probe video sequence, based on our learned model, we track facial features using ICondensation [24], while recognizing facial expressions in the same probabilistic framework. The probe set includes videos of random expression changes, which may not begin or end with a neutral expression. The duration and the intensity of the expressions are varied. The transition between different expressions is represented as the evolution of the posterior probability of the basic paths. Our empirical study demonstrates that the probabilistic approach can recognize expression transitions effectively. We also synthesize image sequences of changing expressions through the manifold model.

Differing from traditional methods that consider expression tracking and recognition in separate stages, we address these

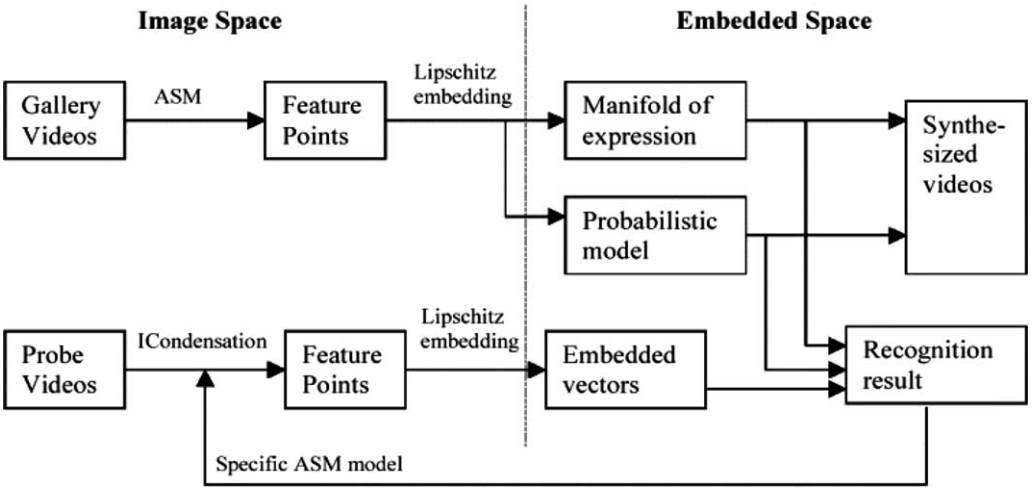

Fig. 2. System diagram.

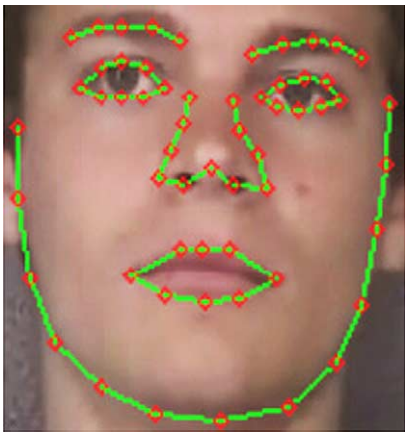

Fig. 3. The shape model, defined by 58 facial landmarks.

tasks in a common probabilistic framework, which enables them to be solved in a cooperative manner.

The remainder of this paper is organized as follows. In Section 2 we discuss related work. We then discuss the properties of Lipschitz embedding in Section 3. Section 4 covers the learning of our proposed representation, while Section 5 describes the framework to track and recognize facial deformation. In Section 6, we show how to synthesize facial expressions using our model. Section 7 reports our experimental results, and Section 8 presents conclusions and future work.

2. Related work

In the past decade, many techniques have been proposed to automatically classify expressions in still images, using methods based on Neural Networks [25,26], Gabor wavelets [5,27] and rule-based methods [3], to mention just a few. However, in recent years, more attention has been given to modeling facial deformation in dynamic scenarios [28–30], which allows the integration of information temporally across the video sequence, potentially increasing recognition rates over single-image approaches. Bassili [31] suggested that motion in the image of a face would allow expression to be identified with minimal information about the spatial arrangement of features. Cohen et al. [28] proposed a new architecture of HMMs to segment and recognize facial expression from video sequence automatically. Probabilistic video analysis has gained significant attention since the seminal work of Isard and Blake [24]. They introduced a time series state space model parameterized by a tracking motion vector. For such dynamic scenarios, current methods work in two separate stages: tracking and recognition. The tracking module extracts features over time, while the recognition module processes this information for expression classification.

Many researchers have explored the nature of the space of facial expressions. Schmidt and Cohn [32] measured 195 spontaneous smiles from 95 individuals through facial electromyographic (EMG) data and found consistency in zygomaticus major muscle activity over time. Zhang et al. [5] used a two-layer perceptron to classify facial expressions. They found that five to seven hidden perceptrons are probably enough to represent the space of feature expressions. Tenenbaum and Freeman [33] talked about separating style and content by using symmetry or asymmetry bilinear models. Chuang et al. [34] also showed that the space of facial expression can be modeled with a bilinear model. They fit two formulations of bilinear models, asymmetric and symmetric, to facial expression data.

More recently, Seung and Lee [9] suggested representing the variability of images as low-dimensional manifolds embedded in image space. Tenenbaum et al. [10] introduced Isomap to find meaningful low-dimensional structures hidden in the high-dimensional data that is guaranteed to converge asymptotically to the true structure. Roweis and Saul [11] showed that locally linear embedding is able to learn the global structure of non-linear manifolds, such as those generated by images of faces with only pose and illumination change. Elad and Kimmel [35] used the invariant signature of manifolds for object recognition.

Lyons et al. [27] conducted a quantitative low-dimensional analysis from image features for coding facial expressions. They used non-linear non-metric multidimensional scaling of Gabor-labelled elastic graphs. Wang et al. [36] demonstrated the importance of applying non-linear dimensionality reduction in the field of non-rigid object tracking. In fact, representing the object state as a globally coordinated low-dimensional vector improves tracking efficiency and reduces local minimum problems in optimization. They learn the object's intrinsic structure in a low dimension manifold with density modeled by a mixture of factor analyzers. Our work also models the intrinsic structure of facial expressions for tracking, but extends it to include recognition in a unified probabilistic framework.

Many systems obtain facial motion information by computing dense flow between successive image frames. But flow estimates are easily disturbed by the variation of lighting and non-rigid motion, and they are also sensitive to the inaccuracy of image registration and motion discontinuities [29]. Model-based approaches, such as active shape models (ASM) [37] and active appearance models (AAM) [38], have been successfully used for tracking facial deformation. The ASM method detects facial landmarks through a local-based search constrained by a global shape model, statistically learned from training data. The AAM algorithm elegantly combines shape and texture models, assuming a linear relationship between appearance and parameter variation. Both methods, however, tend to fail in the presence of non-linear image variations such as those caused by large facial expression changes. In our approach, we use specific ASM models for each cluster in the embedded space. On-line model selection is done probabilistically in a cooperative manner with expression classification, thus improving tracking reliability.

Zhou et al. [39] proposed a generic framework to track and recognize human faces simultaneously by adding an identity variable to the stale vector in the sequential importance sampling method. The posterior probability of the identity variable is then estimated by marginalization. Their work, however, does not consider tracking and recognition of facial

deformation, the main focus of this paper. We were also inspired by the work of Lee et al. [14], who present a method for modeling and recognizing human faces in video sequences. They use an appearance model composed of pose manifolds and a matrix of transition probabilities to connect them. In our work, we consider transition probabilities among clusters in the embedded space, effectively capturing the dynamics of expression changes and exploiting the temporal information for recognition.

3. Lipschitz embedding

Lipschitz embedding [20,21] is a powerful embedding method used widely in image clustering and image search. For a finite set of input data S, Lipschitz embedding is defined in terms of a set R of subsets of S, $R=\{A_1,A_1,\ldots,A_k\}$. The subsets A_i are termed the reference sets of the embedding. Let $d(o;A)$ be an extension of the distance function d to a subset $A\subset S$, such that $d(o,A)=\min_{x\in A}\{d(o,x)\}$. An embedding with respect to R is defined as a mapping F such that $F(o)=(d(o;A_1); d(o;A_2);\ldots, d(o;A_k))$. In other words, Lipschitz embedding defines a coordinate space where each axis corresponds to a subset $A_i\subset S$ of the objects, and the coordinate values of object O are the distances from O to the closest element in each of A_i.

With a suitable definition of the reference set R, the distance of all pairs of data points in the embedding space is bounded [40]. So Lipschitz embedding works well when there are multiple clusters in the input data set [22,23]. In our algorithm, we preserve the intrinsic structure of the expression manifold by combining Lipschitz embedding and the main feature of Isomap [10]. Given a video gallery covering six basic facial expressions, there are six 'paths' from the neutral image to the six sets of images with the basic expressions at apex on the manifold. In Fig. 1, the apex sets in 3D space are illustrated as the points within the circles. Each path is composed of many small steps (the difference between consecutive frames). Different paths contain information on how the expressions evolve. The comparative positions between those paths correspond to the relationship between different expressions.

The distance function in Lipschitz embedding reflects the distance between points on the manifold. The crucial property that we aim to retain is proximity; i.e. which points are close to each other and which are far from each other. Due to the essential non-linear structure of the expression manifold, the classical approaches of multidimensional scaling (MDS) [41] and PCA fail to detect the true degrees of freedom of the face data set. Tenenbaum et al. [10] seek to preserve the intrinsic geometry of the data by capturing the geodesic manifold distance between all pairs of data points. For neighboring points, input-space distance provides a good approximation to geodesic distance. For distant points, geodesic distance can be approximated by adding up a sequence of 'short hops' between neighboring points. This shortest path can be computed efficiently by the Dijkstra Algorithm [42]. The details of geodesic distance computation can be found in [10].

For our experiments, we used six reference sets, each of which contains images of only one kind of basic facial expression at its apex. The embedded space is six-dimensional. The distance function is the geodesic manifold distance. After we apply the modified Lipschitz embedding to the gallery set, there are six basic paths in the embedded space, emanating from the center that corresponds to the neutral image. The images with blended expression lie between the basic paths. In the embedded space, expressions can be recognized by using the probabilistic model described in Section 5.

4. Learning dynamic facial deformation

We are interested in embedding the facial deformations of a person in a very low-dimensional space, which reflects the intrinsic structure of facial expressions. From training video sequences of different people undergoing different expressions, a low-dimensional manifold is learned, with a subsequent probabilistic modeling used for tracking and recognition. The goal of the probabilistic model is to exploit the temporal information in video sequences. Expression recognition is performed on the manifold constructed for each individual.

4.1. The training database

For preliminary testing, we collected data from two subjects who were asked to perform six basic facial expressions multiple times. To reduce the influence of illumination variation, we preprocessed the training data video sequence by detecting a set of 2D facial landmarks in each image, which defines the shape of a face in each particular frame. We use the active shape model algorithm to accomplish this task. With a good manual initialization and separate training models prepared specifically for each expression image set, we can extract the face shape precisely. Fig. 3 shows the facial points in our shape model. The detailed facial deformation such as wrinkles and dimpling are ignored. But the positions of the feature points still provide plenty of information to recognize expression correctly based on our experiments. We expect better recognition results for a facial model with higher spatial resolution when more details of facial deformation can be captured.

The whole training dataset, comprising different video sequences of different people undergoing different facial expressions, is then specified by a set $X=\{x_1,\ldots,x_n\}$, where $x_i\in R^{2D}$ denotes a set of D facial points in a particular frame, and n denotes the total number of images in the training data. Unlike traditional manifold embedding methods, where data can be in any temporal order, our training images are temporally ordered according to the video sequences, thus allowing the learning of dynamics on the manifold, as we will show later.

To embed the high dimension data set $X=\{x_1,\ldots,x_n\}$ with $x_i\in R^{2D}$ to a space with low dimension $d<2D$, we use our modified Lipschitz embedding algorithm, as described in the previous section. Our goal is to find the latent variable $Y=\{y_1, y_2,\ldots,y_n\}$, where $y_i\in R^d$. This latent variable encodes the knowledge of the data set and controls the data variations.

4.2. Mixture model on the embedded space

In the lower dimensional embedded space, we describe the distribution of the data using a Gaussian mixture model (GMM). The expectation maximization (EM) algorithm is used to estimate the distribution. The following equation describes the density model, where $p(y)$ is the probability that a point in the low-dimensional space is generated by the model, k is the number of Gaussians, $p(\omega=i)$ constitutes the mixture coefficients and $N(\mu_i, C_i)$ describes each Gaussian distribution with mean μ_i and covariance matrix C_i:

$$p(y) = \sum_{i=1}^{K} p(\omega = i)N(\mu_i, C_i) \tag{4.1}$$

Fig. 4 shows the result of projecting our training data (set of facial shapes) onto a three-dimensional space using our modified Lipschitz embedding.

The appearance of different subjects could be aligned through a common 3D face model. Currently, we build a separate manifold for each subject. These manifolds share a similar 'skeleton' shape, but vary in reference set positions and path directions. With warped appearance data, the subjects from different subjects can be aligned through linear or non-linear alignment [17].

4.3. Cluster-based active shape models

If we were to train an active shape model from all the images in a data set together, the significant variation in the data set would not be modeled well and the tracking performance would be poor. Instead, we train a set of ASM models for each image cluster; that is, for each set of images corresponding to a mixture center (with a defined covariance) of the GMM in the embedded space.

We also propose a method to select and probabilistically integrate the ASM models in the ICondensation framework. We will show in Section 5 that online model selection allows tracking to be robust under large expression variations.

In ASM, a shape vector S is represented in the space spanned by a set of eigenvectors learned from the training data. As a result, S may be expressed as:

$$S = \bar{S} + Us \tag{4.2}$$

where $\bar{S}$ is the mean shape, U is the matrix consisting of eigenvectors and s constitutes the shape parameters, which are estimated during ASM search. In Section 4, we will describe how tracking is achieved using the learned ASM models.

4.4. Learning dynamics on the manifold

Based on the manifold representation, we can learn a dynamic model, defined as the transition probability $p(y_t|y_{t-1})$. Let $\omega \in \{1,\ldots,k\}$ be a discrete random variable denoting the cluster center and let $r \in \{1,\ldots,n_r\}$ be a discrete random variable denoting the expression class. For this work, $n_r = 6$, meaning that r can assume six basic expressions. We have been using the prototypical universal expressions of fear, disgust, happiness, sadness, anger and surprise, though the method does not depend on this particular grouping.

The dynamic model can be factorized in the following way:

$$\begin{aligned} p(y_t|y_{t-1}) &= \sum_{w_t} p(y_t|y_{t-1}, \omega_t)p(\omega_t|y_{t-1}) \\ &= \sum_{\omega_t,\omega_{t-1}} p(y_t|y_{t-1}, \omega_t)p(\omega_t|\omega_{t-1})p(\omega_{t-1}|y_{t-1}) \end{aligned} \tag{4.3}$$

where

$$p(\omega_t|\omega_{t-1}) = \sum_{r_{t-1}} p(\omega_t|\omega_{t-1}, r_{t-1})p(r_{t-1})$$

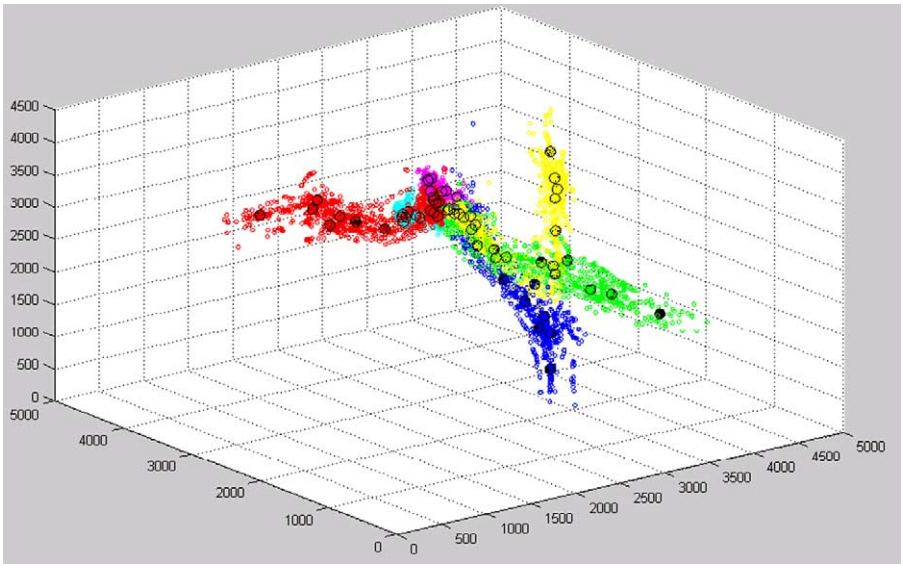

Fig. 4. An expression manifold projected on its first three dimensions. Points with different colors represent images with different expression. Anger, red; disgust, green; fear, blue; sad, cyan; smile, pink; surprise, yellow. The black points represent the mixture centers.

This assumes that ω_t and y_{t-1} are conditionally independent given ω_{t-1}.

For each state of r_{t-1} (i.e. each expression class), the cluster transition dynamics $P(\omega_t|\omega_{t-1}, r_{t-1})$ can be learned from the training data. $P(y_t|y_{t-1}, \omega_t)$ is the dynamic model for a known cluster center. The dynamics in a fixed cluster are similar for each expression. Since the intra-cluster variations are much smaller than the inter-cluster variations, we approximate the truth by assuming the dynamics in a fixed cluster is the same for each expression. If each cluster contains only one point, the difference between our approximation and the truth becomes zero.

Similar to Wang et al. [36], we also model the within cluster transition as a first order Gaussian auto-regressive process (ARP) by:

$$p(y_t|y_{t-1}, \omega_t) = N(A_{\omega_t} y_{t-1} + D_{\omega_t}, \mathrm{BB^T}) \tag{4.4}$$

which can be represented in generative form as

$$y_t = A_{\omega_t} y_{t-1} + D_{\omega_i} + Bw_k \tag{4.5}$$

where A_{ω_t} and D_{ω_t} are the deterministic parameters of the process, BB^{T} is the covariance matrix, and w_k is independent random white noise.

For AR parameter learning, we use the same method as Blake and Isard [44]. Combining Eqs. (4.3), (4.4) and (4.5), we get:

$$\begin{aligned} p(y_t|y_{t-1}) &= \sum_{\omega_t \omega_{t-1} r_{t-1}} p(y_t|y_{t-1}, \omega_t) p(\omega_t|\omega_{t-1}, r_{t-1}) p(r_{t-1}) p(\omega_{t-1}|y_{t-1}) \\ &= \sum_{\omega_t} N(A_{\omega_t} y_{t-1} + D_{\omega_t}, \mathrm{BB}^T) \alpha(\omega_t; y_{t-1}) \end{aligned} \tag{4.6}$$

where

$$\alpha(\omega_t; y_{t-1}) = \sum_{\omega_{t-1}, r_{t-1}} P(\omega_t|\omega_{t-1}, r_{t-1}) P(r_{t-1}) P(\omega_{t-1}|y_{t-1}) \tag{4.7}$$

Wang et al. [36] pointed out that the equations above model a mixture of Gaussian diffusion (MGD), whose mixture term is controlled by the random variable ω_t. In our work, the mixture term is also controlled by the expression recognition random variable.

5. Probabilistic tracking and recognition

In the previous section, we showed how to learn a facial expression model on the manifold as well as its associated dynamics. Now, we show how to use this representation to achieve robust online facial deformation tracking and recognition. Our probabilistic tracking is based on the ICondensation algorithm [24], which is described next, followed by expression classification. Both tracking and recognition are described in the same probabilistic framework, which enables them to be carried out in a cooperative manner.

5.1. ICondensation tracking

Our object state is composed of rigid and non-rigid parts, defined by $s=(x,y,\theta,\mathrm{sc};y_1 \ldots y_d)$. The rigid part (x,y,θ,sc) represents the rigid face motion (position, orientation and scale), while the non-rigid part $(y_1 \ldots y_d)$ is the low-dimensional representation of facial deformation obtained by our modified Lipschitz embedding, as described in Section 3.

At time t, the conditional object state density is represented as a weighted set of samples $\{(s_t^{(n)}, \pi_t^{(n)}), \quad n=1, \ldots, N\}$, where $s_t^{(n)}$ is a discrete sample with associated weight $\pi_t^{(n)}$, where $\sum_n \pi_t^{(n)} = 1$. Below we illustrate one step of a sample's evolution.

After this step, the state with largest weight describes the tracking output in each frame, consisting of face pose (x,y,θ,sc) and deformation, which is obtained by projecting $(y_1 \ldots y_d)$ back to the original shape space, through a nearest-neighbor scheme.

Sequential importance sampling iteration:

Main objective: Generate sample set

$S_t = \{(s_t^{(n)}, \pi_t^{(n)}), \ n=1, \ldots, N\}$ at time t from sample set $S_{t-1} = \{(s_{t-1}^{(n)}, \pi_{t-1}^{(n)}), \ n=1, \ldots, N\}$ at time $t-1$.

Algorithm:

For each sample, $n=1$ to N:

(1) Create samples $\tilde{s}_t^n$

Choose one of the following sampling methods with a fixed probability:

(1) Generate sample from initialization prior.
(2) Generate sample from importance resampling, where the importance function is the posterior from time $t-1$;
(2) Predict s_t^n from $\tilde{s}_t^n$
   (a) If $\tilde{s}_t^n$ was generated from the prior probability, choose s_t^n from $\tilde{s}_t^n$ adding a fixed Gaussian noise.
   (b) If $\tilde{s}_t^n$ was generated from the posterior probability, apply the dynamic model for prediction. For the rigid state part, we use constant prediction, adding a small Gaussian noise. For the non-rigid part, we use the MGD noise model, where the weight of each component is controlled by the cluster center distribution $p(\omega_t)$ and expression classification distribution $p(r_t)$.
(3) Update the set of samples. The measurement of the sample s_t^n is $\pi_t^{(n)} = \lambda_t^{(i)} M(s_t^n)$, where $\lambda_t^{(i)}$ is the importance sampling correction term. M is the sample measurement function, described in the next subsection.
(4) After all the samples are generated and measured, normalize $\pi_t^{(n)}$ so that $\sum_n \pi_t^{(n)} = 1$ and store the sample set as $\{(s_t^{(n)}, \pi_t^{(n)}), \ n=1, \ldots, N\}$

5.1.1. Sample measurement

In order to measure a sample (function M in the algorithm above), we proceed in the following way. For each mixture center in the embedded space, a specific ASM model is selected to measure image observation. This measure is given by a residual error obtained after applying one step of ASM search (we refer to Cootes et al. [37] for details on the search process). Face pose initialization is given by the sample rigid part (x,y,θ,sc) and shape initialization is computed by

projecting the non-rigid part ($y_1 \ldots y_d$) of the sample back to the original shape space (using a nearest-neighbor scheme).

Once we have a residual error for each one of the mixture centers, the desired sample measurement is obtained by a weighted sum of these residuals, where the weights corresponds to the likelihood of the sample non-rigid part ($y_1 \ldots y_d$) in each Gaussian model.

This scheme allows tracking to be robust under large facial expression changes, as we will show in Section 7. Next we describe how to update expression classification in each frame, using a common probabilistic framework.

5.2. Expression recognition updating

We have already showed that the distribution of the discrete random variable r (the expression recognition variable) directly affects tracking (see sample prediction and dynamic model learning). Now we show how to update the posterior probability $p(r_t|y_{0:t})$ in every frame to identify facial deformation.

In the ICondensation tracking, by assuming statistical independence between all noise variables, Markov property and priors of the distributions $p(\omega_0|r_0)$, $p(r_0|y_0)$, $p(y_t|y_{t-1})$, on embedded space, our goal is to compute the posterior $p(r_t|y_{0:t})$. It is in fact a probability mass function (PMF) as well as a marginal probability of $p(r_t,\omega_t|y_{0:t})$. Therefore, the problem is reduced to computing the posterior probability $p(r_{0:t},\omega_{0:t}|y_{0:t})$.

$$
\begin{aligned}
&p(r_{0:t}, \omega_{0:t}|y_{0:t}) \\
&\quad = p(r_{0:t-1}, \omega_{0:t-1}|y_{0:t-1}) \\
&\qquad \times \frac{p(y_t|r_{0:t-1}, \omega_{0:t-1})p(r_t|r_{t-1})p(\omega_t|\omega_{t-1})}{p(y_t|y_{0:t-1})} \\
&\quad = p(r_0, \omega_0|y_0) \prod_{l=1}^{t} \frac{p(y_l|r_l, \omega_l)p(r_l|r_{l-1})p(\omega_l|\omega_{l-1})}{p(y_l|y_{0:l-1})} \qquad (5.1)
\end{aligned}
$$

By marginalizing over $\omega_{0:t}$ and $r_{0:t-1}$, we obtain:

$$
\begin{aligned}
p(r_t = l|y_{0:t}) = &\int_{\omega_0}\int_{r_0}\cdots\int_{\omega_{t-1}}\int_{r_{t-1}}\int_{\omega_t} p(r_0, \omega_0|y_0) \\
&\times \prod_{l=1}^{t} \frac{p(y_l|r_l, \omega_l)p(r_l|r_{l-1})p(\omega_l|\omega_{l-1})}{p(y_l|y_{0:l-1})} \qquad (5.2) \\
&\mathrm{d}\omega_t \mathrm{d}r_{t-1} \mathrm{d}\omega_{t-1} \ldots \mathrm{d}\omega_0 \mathrm{d}r_0
\end{aligned}
$$

This equation can be computed by prior distributions and the product of the likelihood $\prod_{l=1}^{t} p(y_l|r_l, \omega_l)$.

6. Synthesis of dynamic expressions

The manifold model can also be used to synthesize an image sequence with changing expressions. Given expression r, $r=1$, …,6, we keep the cluster indexing $l_1,\ldots,l_k$, and k is the number of the clusters, such that:

$$p(\omega^{l_1}|r = l) < p(\omega^{l_2}|r = l) \ldots < p(\omega^{l_k}|r = l)$$

For expression r, there are m gallery videos that begin from the neutral expression, pass the apex, and end with the neutral expression. We set the first video sequence as a template. Then we apply dynamic time warping [43] to the following $m-1$ image sequences. Thus we have a standard time index for all m videos. For every cluster along the path r, we can measure the duration of the cluster by the range of time index of the images within the cluster. Note we compute the time range for increasing and decreasing expression separately since a cluster may cover both types of images at the same time. The time range for each cluster is w_i, $i=1,\ldots,k$. The average time range of all clusters is $\bar{w}$.

The algorithm for synthesizing an image sequence from expression A to expression B is listed as following. The critical part is to find a trajectory that maximizes the probability of the transitions between the clusters A_r and B_r. The optimal trajectory is computed by dynamic programming [45]. The correlations between consecutive frames are maximized locally at the same time. To eliminate the jitter and redundancy in the image sequence, we keep a cache for recently appeared frames. If the same frame from the gallery appearances more than twice in the passed n frames, it should be removed from the final video sequence. n is an empirical window width.

Input:

The beginning expression category: $A \in \{1, \ldots, 6\}$
The ending expression category: $B \in \{1, \ldots, 6\}$
The length of synthesized video sequence: *fnum*
The embedded vectors of k cluster centers:
$d_i \in R^6$, $i = 1, \ldots, k$

Output:

Image sequence P

Function:

floor (x) : return the maximum integer no more than x.
findnear (d, z) : return the nearest z points to the embedded vector d on the learned manifold.
GetRaw (d) : return the corresponding face image to the embedded vector d.
correlation (x, y): return the correlation between two images.

$T = floor\ (fnum / \bar{w})$;

$n_1 = A_r$; {the cluster with strongest expression A}

$n_T = B_r$; {the cluster with strongest expression B}
$[n_2, \ldots, n_{T-1}] = \arg\max(P(\omega^{n_2}|\omega^{n_1}) \ldots P(\omega^{n_T}|\omega^{n_{T-1}}))$;
$count = 0$;
for i = 1 **to** $T-1$
  $betweenc = (w_{n_i} + w_{n_{i+1}})/2$;
  $fbegin = findnear\ (d_i, 1)$;
  $fend = findnear\ (d_{i+1}, 1)$;
  $count = count + 1$;
  $P_{count} = GetRaw\ (fbegin)$;
  for $j = 1$ **to** $betweenc - 1$
    $dist = (fend - fbegin)/(betweenc - j)$;
    $candi = findnear(fbegin + dist, 5)$;
    $comp = GetRaw(fbegin)$;
    for $k = 1$ **to** 5
      $candi\_im(k) = GetRaw(candi(k))$;
      $corr(k) = correlation(comp, candi\_im(k))$;
      if $(k == 1)$
      then
        $se = 1$;
        $\max = corr\ (k)$;
      else
        if $(corr\ (k) > \max)$
          $se = k$;
          $\max = corr\ (k)$;
        end
      end
    end
    $count = count + 1$;
    $P_{count} = candi\_im\ (se)$;
    $fbegin = candi\ (se)$;
  end
end
return P_i, $i = 1, \ldots, count$

7. Experimental results

In this section, we present our experimental results on facial deformation tracking and recognition.

7.1. Data set

To learn the structure of the expression manifold, we need $O(10^3)$ images to cover basic expressions for each subject and to enable stable geodesic distance computation. Since there is no database with a sufficiently large amount of subject data available, we built our own small data set for the experiments. In our experiments, two subjects were instructed to perform a series of six kinds of prototypical facial expressions, representing happiness, sadness, anger, surprise, fear, and disgust. The subjects repeated the series seven times for the gallery set. The probe set includes a long sequence (more than 10^4 frames) where the subject can change his/her expression randomly. To simplify the problem, we assume constant illumination and near frontal view pose. The sequences were recorded at 30 fps and stored at a resolution of 320×240. All results in this paper were obtained on a Xeon 2.8 GHz CPU. The complete process, including alignment, embedding, and recognition, runs at 5 fps.

To generate the shape sequence from the training data set, we trained ten ASM models for different kinds of deformations. We manually select the model in this offline stage to robustly track facial deformation along the video sequences. The shape space dimension is 90. We used our modified Lipschitz algorithm to obtain a space with dimensionality $d=3$.

We realized the difference between posed expression and spontaneous expressions in terms of amplitude and dynamics [46]. The future system will test on spontaneous expression of subjects.

7.2. Tracking and expression recognition

We verified that our probabilistic method is able to track and recognize long sequences of subjects performing subtle and large expression changes. Fig. 5 shows two frames from a tracking and recognition test using a new video sequence. The overlaid graphical bars for each expression label in the figure indicate their respective recognition probabilities. A complete output video sequence is available at http://ilab.cs.ucsb.edu/demos/IVC-seq2.m2v.

We visualize the learned manifold at the same time at video http://ilab.cs.ucsb.edu/demos/IVC-seq1.m2v. The embedded vector of the current frame is represented as a black point. During the expression transition, the black point 'walks' from one expression path to another. The viewpoint of the manifold is changed concurrently for better visualization. Fig. 6 shows some sample images from the available video. The first image is during a transition from fear to surprise. The second image is during a transition from anger to disgust. The third image and the fourth image are sadness and happiness respectively. The bar figures indicate the expression transition correctly. The quantitative measurement of expression recognition for every frame is not available because the output is represented as the posterior probability of basic expressions, and we do not have ground truth for this kind of representation.

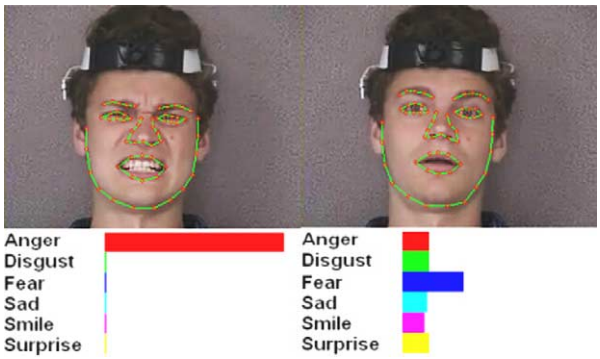

Fig. 5. Sample frames of our output tracking and recognition result in a video sequence.

We also quantitatively analyze the performance of our tracker with a standard ASM tracker. Fig. 7 shows a precision comparison, considering as ground truth a manual labeling of eye corners and lip corners. The same images were used to train both trackers. The difference is that our method automatically splits this data to train a set of models, which are probabilistically selected during tracking. This allows more robust performance under large facial expression changes.

7.3. Expression synthesis

With the manifold model, we synthesize image sequences of aligned face appearances with changing expressions. There are about 6000 images from 42 video sequences (seven for each basic expression) in each gallery set. The lengths of synthesized image sequences are around 200.

Figs. 8 and 9 show some selected images (every 20th frame) from the synthesized sequences. The trajectories with the maximum transition probability between clusters reflect the expression change correctly.

8. Conclusions

We proposed a novel framework for dynamic facial expression analysis. We now summarize our main contributions:

(1) A new representation for tracking and recognition of facial expressions, based on manifold embedding and probabilistic modeling in the embedded space. Our experimental results show that manifold methods provide an analytical way to analyze the relationship between different expressions, and to recognize blended expressions.
(2) A robust method for facial deformation tracking based on a set of ASM models, which are probabilistically selected during tracking, improving reliability under large expression changes.
(3) A probabilistic expression classification method, which integrates information temporally across the video sequence. In contrast with traditional methods that

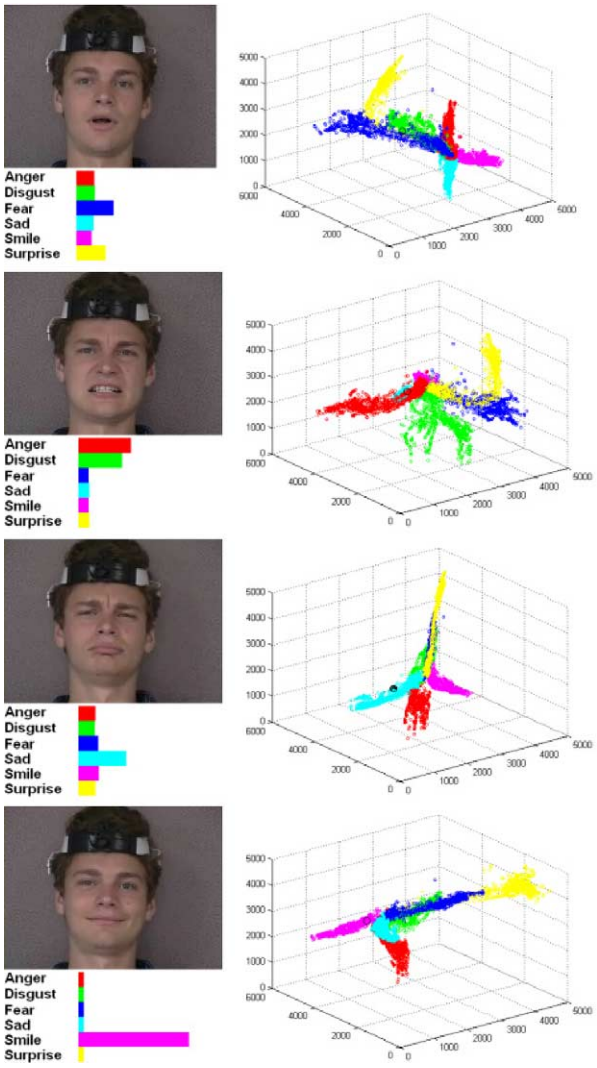

Fig. 6. Facial expression recognition result with manifold visualization.

consider expression tracking and recognition in separate stages, we address these tasks in a common probabilistic framework, which enables them to be solved in a cooperative manner.

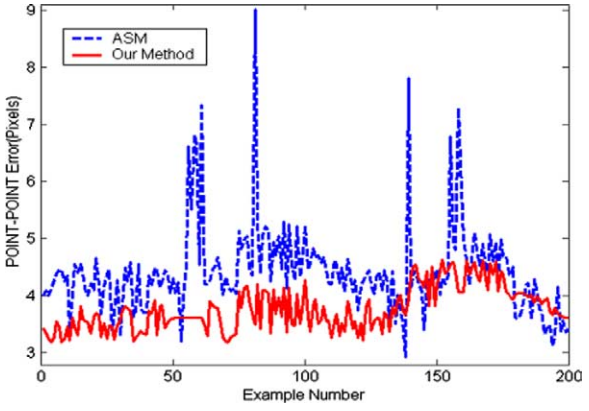

Fig. 7. Comparison of tracking precision between an ASM tracker and our method. We have obtained considerably improvement, mainly under the presence of images with large expression changes.

Fig. 8. Twelve frames selected from a transition from anger to happiness.

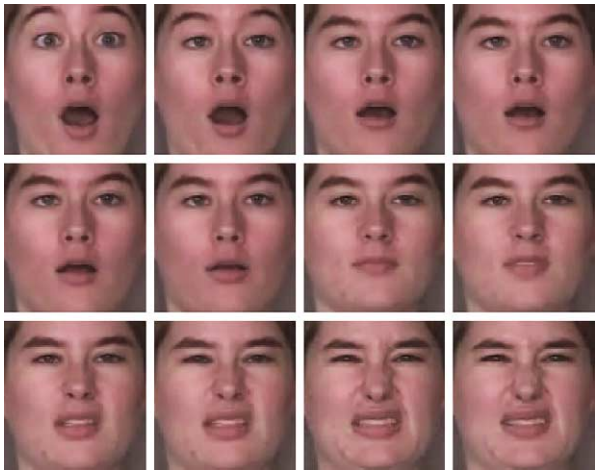

Fig. 9. Twelve frames selected from a transition from surprise to disgust.

The main limitation of the work is the 2D shape model ignored the detailed facial deformation, such as wrinkles, which will help recognize the nuances of facial expressions. 3D morphable face model will provide more information for robust expression recognition. The temporal approximation of dynamics within the cluster can be improved by decreasing the size of clusters with higher computational expense. We try to achieve the best tradeoff between accuracy and speed. The current data are directed expressions. We will test on the spontaneous expression in the future system.

We will evaluate and quantify the results more systematically with many more subjects in future work. Another future research direction is to consider variation on face pose and illumination [47,48], which will add more degrees of freedom to manifold of expression. How these factors affect the intrinsic geometry of expression manifold will be a challenging topic for future study.

Acknowledgements

This work has been supported by NSF ITR grant #0205740 and under the auspices of the US Department of Energy by the

Lawrence Livermore National Laboratory under contract No. W-7405-ENG-48. Thanks to the comments from the reviewers. They together make this paper a better one.

References

[1] P. Ekman, W. Friesen, Facial Action Coding System: Manual, Consulting Psychologist Press, Palo Alto, 1978.
[2] P. Ekman, Emotion in the Human Face, Cambridge University Press, New York, 1982.
[3] M. Black, Y. Yacoob, Recognizing facial expressions in image sequences using local parameterized models of image motion, International Journal of Computer Vision 25 (1) (1997) 23–48.
[4] I. Essa, A. Pentland, Coding, analysis interpretation, recognition of facial expressions, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 757–769.
[5] Z. Zhang, M. Lyons, M. Schuster, S. Akamatsu, Comparison between geometry-based and gabor wavelets-based facial expression recognition using multi-layer perceptron, Proceedings of Int. Conf. on Automatic Face and Gesture Recognition, 1998.
[6] M. Pantic, L. Rothkrantz, Automatic analysis of facial expressions: the state of the art, IEEE Trans. Pattern Analysis and Machine Intelligence 22 (12) (2000).
[7] B. Fasel, J. Luettin, Automatic facial expression analysis: a survey, Pattern Recognition (2003) 36.
[8] J.A. Russell, Core affect and the psychological construction of emotion, Psychological Review 110 (2003) 145–172.
[9] H. Sebastian Seung, Daniel.D. Lee, The manifold ways of perception, Science 290 (2000) 2268–2269.
[10] J. Tenenbaum, V. de Silva, J. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.
[11] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[12] M. Brand, Charting a manifold, Neural Information Processing Systems (2002) 15.
[13] S. Roweis, L. Saul, G. Hinton, Global coordination of local linear models, Neural Information Processing Systems 14 (2002) 889–896 NIPS'2001.
[14] K. Lee, J. Ho, M. Yang, D. Kriegman, Video-based face recognition using probabilistic appearance manifolds Proceedings of Int. Conf. on Computer Vision and Pattern Recognition, June 16-22, 2003.
[15] M.-H. Yang, Face recognition using extended isomap, International Conference on Image Processing (2002).
[16] Douglas Fidaleo, M. Trivedi, Manifold analysis of facial gestures for face recognition, ACM SIGMM Multimedia Biometrics Methods and Application Workshop, Nov. 8 2003.
[17] Y. Chang, C. Hu, M. Turk, Manifold of facial expression, Proceedings of IEEE International Workshop on Analysis and Modeling of Faces and Gestures, Nice, France, Oct. 17, 2003.
[18] Y. Chang, C. Ho, M. Turk, Probabilistic expression analysis on manifolds, Proceedings of Int. Conf. on Computer Vision and Pattern Recognition, Washington DC, June 2004.
[19] C. Hu, Y. Chang, R. Feris, M. Turk, Manifold based analysis of facial expression, Proceedings of IEEE Workshop on Face Processing in Video, Washington, DC, June 2004.
[20] J. Bourgain, On Lipschitz embedding of finite metric spaces in hilbert space, Israel J. Math. 52 (1/2) (1985) 46–52.
[21] W. Johnson, J. Lindenstrauss, Extension of Lipschitz mapping into a hilbert space, Contemporary Mathematics 26 (1984) 189–206.
[22] Hristescu G., Farach-Colton M., Cluster-Preserving Embedding of Proteins, Technical Report, Rutgers Univ., Piscataway, New Jersey, 1999.
[23] M. Linial, N. Linial, N. Tishby, G. Yona, Global self organization of all known protein sequences reveals inherent biological signatures, Journal of Molecular Biology 268 (2) (1997) 539–556.
[24] M. Isard, A. Blake, ICondensation: unifying low-level and high-level tracking in a stochastic framework, Proceedings of European Conference on Computer Vision, 1998.
[25] C. Padgett, G. Cottrell, Representing face images for emotional classification, Proceedings of Conf. Advances in Neural information Processing Systems, 1996, pp. 894–900.
[26] Y. Tian, T. Kanade, J.F. Cohn, Recognizing action units for facial expression analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2) (2001).
[27] M.J. Lyons, S. Akamatsu, M. Kamachi, J. Gyob, Coding facial expressions with gabor wavelets, Proceedings of Int. Conf. On Automatic Face and Gesture Recognition, 1998.
[28] I. Cohen, N. Sebe, L. Chen, A. Garg, T. Huang, Facial expression recognition from video sequences: temporal and static modeling, Computer Vision and Image Understanding 91 (1) (2003) 160–187.
[29] Y. Zhang, Q. Ji, Facial expression understanding in image sequences using dynamic and active visual information fusion, Proceedings of Int. Conf. On Computer Vision, Nice, France, 2003.
[30] Y. Li, S. Gong, H. Liddell, Video-based online face recognition using identity surfaces, Proceedings of IEEE International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems, Vancouver, Canada, July 2001, pp. 40–46.
[31] J.N. Bassili, Emotion recognition: The role of facial movement and the relative importance of upper and lower areas of the face, Journal of Personality and Social Psychology 37 (1979) 2049–2059.
[32] K. Schmidt, J. Cohn, Dynamics of facial expression: normative characteristics and individual difference, Intl. Conf. on Multimedia and Expo, 2001.
[33] J.B. Tenenbaum, W.T. Freeman, Separating style and content with bilinear models, Neural Computation Journal 12 (1999) 1247–1283.
[34] E. Chuang, H. Deshpande, C. Bregler, Facial expression space learning Pacific Graphics 2002.
[35] A. Elad, R. Kimmel, On bending invariant signatures for surfaces, IEEE Transaction on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1285–1295.
[36] Q. Wang, G. Xu, H. Ai, Learning object intrinsic structure for robust visual tracking, Proceedings of Int. Conf. on Computer Vision and Pattern Recognition, June 16–22, 2003.
[37] T. Cootes, C. Taylor, D. Cooper, J. Graham, Active shape models — their training and applications, Computer Vision and Image Understanding 61 (2) (1995).
[38] T. Cootes, G. Edwards, C. Taylor, Active appearance models, IEEE Transaction on Pattern Analysis and Machine Intelligence 23 (6) (2001) 681–685.
[39] S. Zhou, V. Krueger, R. Chellappa, Probabilistic recognition of human faces from video, Computer Vision and Image Understanding 91 (1) (2003).
[40] N. Linial, E. London, Y. Rabinovich, The geometry of graphs and some of its algorithmic applications, Combinatorica 15 (1995) 215–245.
[41] F.W. Young, R.M. Hamer, Multidimensional Scaling: History, Theory and Applications, Erlbaum, New York, 1987.
[42] E.W. Dijkstra, A note on two problems in connection with graphs, Numeriche Math 1 (1959) 269–271.
[43] C.S. Myers, L.R. Rabiner, A comparative study of several dynamic time-warping algorithms for connected word recognition, The Bell System Technical Journal 60 (7) (1981) 1389–1409.
[44] A. Blake, M. Isard, Active Contours, Springer, 1998.
[45] D. Bertsekas, Dynamic Programming and Optimal Control, Athena Scientific, Massachusetts, 2000.
[46] M. Stewart Bartlett, B. Braathen, G. Littlewort, E. Smith, J.R. Movellan, An approach to automatic recognition of spontaneous facial actions, Advances in Neural Information Processing Systems, 4 number 15, MIT Press, Cambridge, Massachusetts, 2003.
[47] Y. Li, S. Gong, H. Liddell, Recognizing trajectories of facial identities using kernel discriminate analysis, Proceedings of British Machine Vision Conference, 2001.
[48] S.Z. Li, R. Xiao, Z. Li, H. Zhang, Nonlinear mapping from multi-view face patterns to a gaussian distribution in a low dimensional space, Proceedings of IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS), 2001.