1 Combining Geometrical and Statistical Models for Video-Based Face Recognition

Amit K. Roy-Chowdhury and Yilei Xu
University of California, Riverside,
{amitrc,yxu}@ee.ucr.edu

1.1 ABSTRACT

A number of methods in tracking and recognition have successfully exploited low-dimensional representations of object appearance learned from a set of examples. In all these approaches, the construction of the underlying low-dimensional manifold relied upon obtaining different instances of the object's appearance and then using statistical data analysis tools to approximate the appearance space. This requires collecting a very large number of examples and the accuracy of the method depends upon the examples that have been chosen. In this chapter, we show that it is possible to estimate low-dimensional manifolds that describe object appearance using a combination of analytically derived geometrical models and statistical data analysis. Specifically, we derive a quadrilinear space of object appearance that is able to represent the effects of illumination, motion, identity and shape. We then show how efficient tracking algorithms like inverse compositional estimation can be adapted to the geometry of this manifold. Our proposed method significantly reduces the amount of data that needs to be collected for learning the manifolds and makes the learned manifold less dependent upon the actual examples that were used. Based upon this novel manifold, we present a framework for face recognition from *video sequences* that is robust to large changes in facial pose and lighting conditions. The method can handle situations where the pose and lighting conditions in the training and testing data are *completely disjoint*. We show detailed performance analysis results and recognition scores on a large video dataset.

1.2 INTRODUCTION

Low dimensional representations of object appearance have proved to be one of the successful strategies in computer vision for applications in tracking, modeling and recognition. Active appearance models (AAMs) [8, 16], multilinear models [10, 23, 9, 24], and other low-dimensional manifold representations [14] fall in this genre. In all these approaches, the construction of the underlying low-dimensional manifold relies upon obtaining different instances of the object's appearance under various conditions (e.g., pose, lighting, identity and deformations) and then using statistical data analysis and machine learning tools to approximate the appearance space. This approach requires obtaining a large number of examples of the object's appearance and the accuracy of the method depends upon the examples that have been chosen for the training phase. Representation of appearances that have not been seen during the training phase can be inaccurate. In mathematical modeling terms, this is a *data-driven* approach.

In this chapter, we show that it is possible to learn complex manifolds of object appearance using a combination of *analytically* derived geometrical models and statistical data analysis. We term this as a "Geometry-Integrated Appearance Manifold" (GAM). Specifically, we derive a *quadrilinear* manifold of object appearance that is able to represent the combined effects of illumination, motion, identity and deformation. The basis vectors of this manifold depend upon the 3D geometry of the object. We then show how to adapt the inverse compositional (IC) algorithm to efficiently and accurately track objects on this manifold through changes of pose, lighting and deformations. Our proposed method *significantly reduces the amount of data that needs to be collected for learning the appearance manifolds during the training phase and makes the learned manifold less dependent upon the actual examples that were used*. The process for construction of this appearance manifold is relatively simple, has a solid theoretical basis, and provides a high level of accuracy and computational speed in tracking and novel view synthesis. Depending upon the application, it may be possible to derive the manifold in a completely analytical manner, an example being tracking a rigid object (e.g., vehicle) through pose and lighting changes. In other examples, like face recognition, a combination of analytical approaches and statistical data analysis will be used for learning the manifold.

Based upon the GAM, we present a novel framework for pose and illumination invariant, video-based face recognition. This video-based face recognition system works by (i) learning joint illumination and motion models from video using the GAM, (ii) synthesizing novel views based on the learned parameters, and (iii) designing measurements that can compare two time sequences while being robust to outliers. We can handle a variety of lighting conditions, including the presence of multiple point and extended light sources, which is natural in outdoor environments (where face recognition performance is still relatively poor[31, 19, 20]). We can also handle gradual and sudden changes of lighting patterns over time. The pose and illumination conditions in the gallery and probe can be completely disjoint. We show experimentally that our method achieves high identification rates under extreme changes of pose and illumination.

1.2.1 Novel Contributions and Relation to Past Work

There are three main parts of this paper - learning GAMs using a combination of geometrical models and statistical analysis, adapting the IC algorithm for tracking and view synthesis using these manifolds, and developing the framework for video-based face recognition using the GAM.

• *Learning GAMs:* The analytically derived geometrical models represent the effects of motion, lighting and 3D shape in describing the appearance of an object [4, 21, 27]. The statistical data analysis approaches are used to model the other effects like identity (e.g., faces of different people) and non-rigidity which are not easy to represent analytically. First, lighting is modeled using a spherical harmonics based linear subspace representation [4, 21]. This is then combined with a recent result by the authors that proved the appearance of an image is bilinear in the 3D motion and illumination parameters, with the 3D shape determining the basis vectors of the space [27]. The variations of this analytically derived bilinear basis over identity and deformation are then learned using multilinear SVD [13], and they together form a *quadrilinear* space of illumination, motion, identity and deformation. The GAM can be visualized (see Figure 1.1) as a collection of locally linear tangent planes along the pose dimension, where each tangent plane represents 3D motion in a local region around each pose. See Figure 1.1.

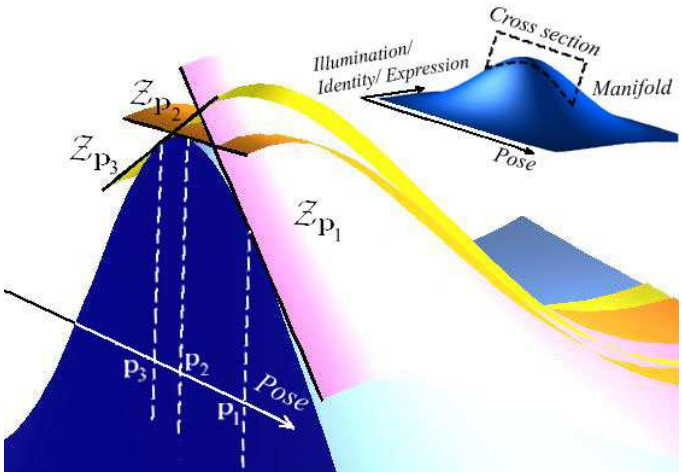


Fig. 1.1 Pictorial representation of variation of a GAM cross-section. Only two axes are shown for simplicity. At each pose, we have the manifold for illumination, identity and deformation. Around each pose, we have the tangent plane to the manifold.

The major difference of GAMs with other methods for computing appearance manifolds and subspaces [23, 9, 14, 16] is that the object appearance space is derived using a combination of analytical models and data analysis tools, while the

previous approaches rely purely on data analysis. This significantly reduces the data collection procedures for computing such manifolds and allows representations of appearances that were not included in the learning phase. We will provide some concrete numerical examples to justify this in the experimental section. Thus our method combines the precision and generalizability of model-based approaches with the robustness provided by statistical learning methods to deviations from the model predictions.

• *Probabilistic Inverse Compositional Tracking and Synthesis on GAMs:* We show how to track and synthesize novel views of an object using the learned GAMs. This is done by adapting the inverse compositional (IC) algorithm to the geometry of the manifold and embedding it within a stochastic framework. This can account for changes in pose, lighting, shape and non-rigidity of the object, as well as local errors in two-frame motion estimation. The inverse compositional (IC) approach [3] is an efficient implementation of the Lucas-Kanade image alignment method and works by moving the expensive computation of gradients and Hessians out of an iterative loop. Due to 3D motion estimation in our case, the expensive computations of derivatives need to take place only at a few discrete poses (not once every frame).

Our tracking algorithm provides 3D estimates of motion, illumination model parameters, and identity and deformation parameters, thus going beyond illumination-invariant 2D tracking [12, 10]. It does not require a texture mapped 3D model of the object as in [28], which can be a severe restriction in many application scenarios, like face recognition. For tracking faces, it is more computationally efficient than 3DMM approaches [6] since it approximates the pose appearance space as a series of locally linear tangent planes, while 3DMM works by finding the best fit on the non-linear manifold (requiring computationally expensive transformations). There is a small, but not significant (for most applications), tradeoff in accuracy in the process.

• *Video-based Face Recognition Using GAMs:* The probabilistic IC tracking on the GAMs described above is then used for video-based face recognition. We assume that a 3D model of each face in the gallery is available. For our experiments, the 3D model is estimated from images, but any 3D modeling algorithm, including directly acquiring the model through range sensors, can be used for this purpose. Given a probe sequence, we track the face automatically in the video sequence under arbitrary pose and illumination conditions using the probabilistic IC tracking on the GAMs. This tracking requires only a *generic* 3D shape model. The learned illumination parameters are used to synthesize video sequences for each gallery under the motion and illumination conditions in the probe. The distance between the probe and synthesized sequences is then computed for each frame. Different distance measurements are explored for this purpose. Next, the synthesized sequence that is at a minimum distance from the probe sequence is computed and is declared to be the identity of the person.

1.2.2 Review of Face Recognition

A broad review of face recognition is available in [31]. Recently there have been a number of algorithms for pose and/or illumination invariant face recognition, many

of which are based on the fact that the image of an object under varying illumination lies in a lower-dimensional linear subspace. In [30], the authors proposed a 3D Spherical Harmonic Basis Morphable Model (SHBMM) to implement a face recognition system given one single image under arbitrary unknown lighting. Another 3D face morphable model (3DMM) based face recognition algorithm was proposed in [5], but they used the Phong illumination model, estimation of whose parameters can be more difficult in the presence of multiple and extended light sources. A novel method for multilinear independent component analysis was proposed in [23] for pose and illumination invariant face recognition. All of the above methods deal with recognition in a single image or across discrete poses and do not consider continuous video sequences. Video-based face recognition requires integrating the tracking and recognition modules and exploitation of the spatio-temporal coherence in the data. The authors in [14] deal with the issue of video-based face recognition, but concentrate mostly on pose variations. Similarly [15] used adaptive Hidden Markov Models for pose-varying video-based face recognition. A probabilistic framework that fuse the temporal information in a prove video by investigating the propagation of the posterior distribution of the motion and identity was proposed in [32]. Another work used adaptive appearance model, adaptive motion model and adaptive particle filter for simultaneously tracking and recognizing people in video [33]. The authors in [18] proposed to perform face recognition by computing the Kullback-Leibler divergence between testing image sets and a learned manifold density. Another work in [1] learn manifolds of face variations for face recognition in video. A method for video-based face verification using correlation filters was proposed in [26], but the pose in the gallery and probe have to be similar.

1.2.3 Organization of the chapter

The rest of the chapter is organized as follows. Section 1.3 presents the GAM-based object representation using the analytically derived illumination and motion basis and machine learned basis of identity and deformation. Robust and efficient tracking algorithms using this object representation is presented in Section 1.4. Then, we propose an integrated tracking and recognition framework for video-based face recognition in Section 1.5. Experimental results and analysis are presented in Section 1.6. Section 1.7 concludes the chapter and highlights future work.

1.3 METHOD FOR LEARNING GAMS

We will start with the illumination representation of [4] and combine it with motion and shape using the results in [27] in order to derive an analytical representation of a low dimensional manifold of object appearance with variations in pose and lighting. We will then apply N-mode SVD,a multilinear generalization of SVD, to learn the variation of this manifold due to changes of identity and object deformations. We will show that the image appearance due to variations of illumination, pose, and deformation, is quadrilinear and compute the basis functions of this space.

1.3.1 An Analytically Derived Manifold for Motion and Illumination

Recently, it was shown that for moving objects it is possible to approximate the sequence of images by a bilinear subspace of nine illumination coefficients and six motion variables [27]. Representing by $\mathbf{T} = \begin{bmatrix} T_x & T_y & T_z \end{bmatrix}^T$ the translation of the centroid of the object, by $\mathbf{\Omega} = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$ the rotation about the centroid, and by $\mathbf{l} \in \mathbb{R}^{N_l}$ ($N_l \approx 9$ for Lambertian objects with attached shadow) the illumination coefficients in a spherical harmonics basis (see [4] for details), the authors in [27] showed that under small motion, the reflectance image at $t_2 = t_1 + \delta t$ can be expressed as

$$I(\mathbf{u}, t_2) = \sum_{i=1}^{9} l_i b_i^{t_2}(\mathbf{u}), \tag{1.1}$$

$$\text{where } \; b_i^{t_2}(\mathbf{u}) = b_i^{t_1}(\mathbf{u}) + \mathbf{A}(\mathbf{u}, \mathbf{n})\mathbf{T} + \mathbf{B}(\mathbf{u}, \mathbf{n})\mathbf{\Omega}. \tag{1.2}$$

In the above equations, $\mathbf{u}$ represents the image point projected from the 3D surface with surface normal $\mathbf{n}$, and $\{b_i^{t_1}(\mathbf{u})\}$ are the original basis images before motion. $\mathbf{A}$ and $\mathbf{B}$ contain the structure and camera intrinsic parameters, and are functions of $\mathbf{u}$ and the 3D surface normal $\mathbf{n}$. For each pixel $\mathbf{u}$, both $\mathbf{A}$ and $\mathbf{B}$ are $N_l \times 3$ matrices. (The exact forms of $\mathbf{A}$ and $\mathbf{B}$ are not necessary for understanding this paper, hence we skip this. The interested reader can see [27].)

It will be useful for us to represent this result using tensor notation as

$$\hat{\mathcal{I}}_{t_2} = \left(\mathcal{B} + \mathcal{C} \times_2 \begin{pmatrix} \mathbf{T} \\ \mathbf{\Omega} \end{pmatrix} \right) \times_1 \mathbf{l}, \tag{1.3}$$

where $\times_n$ is called the *mode-n product* [13]. [1] For an image of size $M \times N$, $\mathcal{C}$ is a tensor of size $N_l \times 6 \times M \times N$. For each pixel (p, q) in the image, $\mathcal{C}_{klpq} = \begin{bmatrix} \mathbf{A}(\mathbf{u}, \mathbf{n}) & \mathbf{B}(\mathbf{u}, \mathbf{n}) \end{bmatrix}$ of size $N_l \times 6$, $\mathcal{B}$ is a sub-tensor of dimension $N_l \times 1 \times M \times N$, comprised of the basis images b_i, and $\mathcal{I}$ is a sub-tensor of dimension $1 \times 1 \times M \times N$, representing the image.

1.3.2 Learning Identity and Deformation Manifold

The above bilinear space of 3D motion and illumination is derived by using the knowledge of the 3D model of the object (tensor $\mathcal{C}$ contains the surface normals). However, the 3D shape is a function of the identity of the object (e.g., the identity of a face) and possible non-rigid deformations. The challenge now is to general-

[1] The *mode-n product* of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n \times \ldots \times I_N}$ by a vector $\mathbf{v} \in \mathbb{R}^{1 \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{v}$, is the $I_1 \times I_2 \times \ldots \times 1 \times \ldots \times I_N$ tensor

$$(\mathcal{A} \times_n \mathbf{v})_{i_1 \ldots i_{n-1} 1 i_{n+1} \ldots i_N} = \sum_{i_n} a_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} v_{i_n}.$$

ize the above analytical model so that it can be used to represent a wide variety of appearances within a class of objects.

We achieve this by learning multilinear appearance models [23, 25] directly from data. Multilinear 3D shape models have been proposed in [24] to learn the shape variation due to identity and expression. For our case, rather than directly modeling the appearance images, we will model the bilinear bases of motion and illumination derived analytically in Section 1.3.1, and then combine all these different variations to obtain a multilinear model of object appearance.

Using $[\bullet]_v$ to denote the vectorization operation, we can vectorize $\mathcal{B}$ and $\mathcal{C}$ in (1.3), and concatenate them, as

$$\mathbf{v} = \begin{bmatrix} [\mathcal{B}]_v \\ [\mathcal{C}]_v \end{bmatrix}. \tag{1.4}$$

This $\mathbf{v}$ is the vectorized bilinear basis for one shape (i.e., one object) with dimension $I_v \times 1$, where $I_v = 7N_lMN$ (N_lMN for $\mathcal{B}$ and $6N_lMN$ for $\mathcal{C}$). Given the 3D shape of I_i objects with I_e different deformations, we can compute this vectorized bilinear basis $\mathbf{v}$ for every combination. For faces, using the 3DMM [6] approaches, these instances can be obtained by choosing different coefficients of the corresponding linear basis functions. With the application to faces in mind, we will sometimes use the words deformation and expression interchangably.

We use $\mathbf{v}_e^i$ to represent the vectorized bilinear basis of identity i with expression e. Let us rearrange them into a training data tensor $\mathcal{D}$ of size $I_i \times I_e \times I_v$ with the first dimension for identity, second dimension for expression (deformation) and the third dimension for the vectorized, analytically derived bilinear basis for each training sample. Applying the *N-Mode SVD* algorithm [13], the training data tensor can be decomposed as

$$\begin{aligned} \mathcal{D} &= \mathcal{Y} \times_1 \mathbf{U}_i \times_2 \mathbf{U}_e \times_3 \mathbf{U}_v \\ &= \mathcal{Z} \times_1 \mathbf{U}_i \times_2 \mathbf{U}_e, \\ \text{where} \quad & \mathcal{Z} = \mathcal{Y} \times_3 \mathbf{U}_v. \end{aligned} \tag{1.5}$$

$\mathcal{Y}$ is known as the core tensor of size $N_i \times N_e \times N_v$, and N_i and N_e are the number of bases we use for the identity and expression. With a slight abuse of terminology, we will call $\mathcal{Z}$, which is decomposed only along the identity and expression dimension with size $N_i \times N_e \times I_v$, to be the core tensor. $\mathbf{U}_i$ and $\mathbf{U}_e$, with sizes of $I_i \times N_i$ and $I_e \times N_e$, are the left matrices of the SVD of

$$\mathcal{D}_{(1)} = \begin{pmatrix} \mathbf{v}_1^{1\mathbf{T}} & \cdots & \mathbf{v}_{I_e}^{1\ \mathbf{T}} \\ & \cdots & \\ \mathbf{v}_1^{I_i\mathbf{T}} & \cdots & \mathbf{v}_{I_e}^{I_i\mathbf{T}} \end{pmatrix}$$

$$\text{and } \mathcal{D}_{(2)} = \begin{pmatrix} \mathbf{v}_1^{1\mathbf{T}} & \cdots & \mathbf{v}_1^{I_i\mathbf{T}} \\ & \cdots & \\ \mathbf{v}_{I_e}^{1\ \mathbf{T}} & \cdots & \mathbf{v}_{I_e}^{I_i\mathbf{T}} \end{pmatrix}, \tag{1.6}$$

where the subscripts of tensor $\mathcal{D}$ indicate the tensor unfolding operation[2] along the first and second dimension. According to the *N-mode SVD algorithm* and equation (5), the core tensor $\mathcal{Z}$ can be expressed as

$$\mathcal{Z} = \mathcal{D} \times_1 \mathbf{U}_i^{\mathbf{T}} \times_2 \mathbf{U}_e^{\mathbf{T}}. \tag{1.7}$$

1.3.3 The GAM of Lighting, Motion, Identity and Deformation

The core tensor $\mathcal{Z}$ contains the basis of identity and expression (or deformation) for $\mathbf{v}$ as

$$\mathbf{v}_i^{eT} = \mathcal{Z} \times_1 \mathbf{c}_i^{\mathbf{T}} \times_2 \mathbf{c}_e^{\mathbf{T}}, \tag{1.8}$$

where $\mathbf{c}_i$ and $\mathbf{c}_e$ are the coefficient vectors encoding the identity and expression. As $\mathbf{v}_i^e$ are the vectorized, bilinear basis functions of the illumination and 3D motion, the core tensor $\mathcal{Z}$ is *quadrilinear* in illumination, motion, identity and expression. As an example, this core tensor $\mathcal{Z}$ can describe all the face images of identity $\mathbf{c}_i$ with expression $\mathbf{c}_e$ and motion ($\mathbf{T}$, $\mathbf{\Omega}$) under illumination $\mathbf{l}$.

Due to the small motion assumption in the derivation of the analytical model of motion and illumination in Section 1.3.1, the core tensor $\mathcal{Z}$ can only represent the image of the object whose pose is close to the pose $\mathbf{p}$ under which the training samples of $\mathbf{v}$ are computed. To emphasize that $\mathcal{Z}$ is a function of pose $\mathbf{p}$, we denote it as $\mathcal{Z}_\mathbf{p}$ in the following derivation. Since $\mathbf{v}$ is obtained by concatenating $[\mathcal{B}]_v$ and $[\mathcal{C}]_v$, $\mathcal{Z}_\mathbf{p}$ also contains two parts, $\mathcal{Z}_\mathbf{p}^\mathcal{B}$ with size $(N_i \times N_e \times N_l MN)$ and $\mathcal{Z}_\mathbf{p}^\mathcal{C}$ with size $(N_i \times N_e \times 6N_l MN)$. The first part encodes the variation of the image due to changes of identity, deformation and illumination at the pose $\mathbf{p}$, and the second part encodes the variation due to motion around $\mathbf{p}$, i.e., the tangent plane of the manifold along the motion direction. Rearranging the two sub-tensors according to the illumination and motion basis into sizes of $N_l \times 1 \times N_i \times N_e \times MN$ and $N_l \times 6 \times N_i \times N_e \times MN$ (this step is needed to undo the vectorization operation of equation (1.4)), we can represent the quadrilinear basis of illumination, 3D motion, identity, and deformation along the first, second, third and forth dimensions respectively. The image with identity $\mathbf{c}_i$ and expression $\mathbf{c}_e$ after motion ($\mathbf{T}$,$\mathbf{\Omega}$) around pose $\mathbf{p}$ under illumination $\mathbf{l}$ can be obtained by

$$\mathcal{I} = \mathcal{Z}_\mathbf{p}^\mathcal{B} \times_1 \mathbf{l} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e + \mathcal{Z}_\mathbf{p}^\mathcal{C} \times_1 \mathbf{l} \times_2 \begin{pmatrix} \mathbf{T} \\ \mathbf{\Omega} \end{pmatrix} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e. \tag{1.9}$$

Note that we did not need examples of the object at different lighting conditions and motion in order to construct this manifold - these parts of the manifold came from the analytical expressions in (1.3).

[2]Assume an Nth-order tensor $\mathcal{A} \in \mathbf{C}^{I_1 \times I_2 \times \ldots \times I_N}$. The matrix unfolding $\mathbf{A}_{(n)} \in \mathbf{C}^{I_n \times (I_{n+1} I_{n+2} \ldots I_N I_1 I_2 \ldots I_{n-1})}$ contains the element $a_{i_1 i_2 \ldots i_N}$ at the position with row number i_n and column number equal to $(i_{n+1} - 1) I_{n+2} I_{n+3} \ldots I_N I_1 I_2 \ldots I_{n-1} + (i_{n+2} - 1) I_{n+3} I_{n+4} \ldots I_N I_1 I_2 \ldots I_{n-1} + \cdots + (i_N - 1) I_1 I_2 \ldots I_{n-1} + (i_1 - 1) I_2 I_3 \ldots I_{n-1} + \cdots + i_{n-1}$.

To represent the manifold at all the possible poses, we do not need such a tensor at every pose. Effects of 3D translation can be removed by centering and scale normalization, while in-plane rotation to a pre-defined pose can mitigate the effects of rotation about the z-axis. Thus, the image of object under arbitrary pose, $\mathbf{p}$, can always be described by the multilinear object representation at a pre-defined $(\mathbf{T}_x^{pd}, \mathbf{T}_y^{pd}, \mathbf{T}_z^{pd}, \mathbf{\Omega}_z^{pd})$, with only $\mathbf{\Omega}_x$ and $\mathbf{\Omega}_y$ depending upon the particular pose. Thus, the image manifold under any pose can be approximated by the collection of a few tangent planes on distinct $\mathbf{\Omega}_x^j$ and $\mathbf{\Omega}_y^j$, denoted as $\mathbf{p}_j$.

1.4 ROBUST AND EFFICIENT TRACKING ON GAMS

We now show how the GAMs of object appearance can be applied for estimation of 3D motion and lighting, which we broadly refer to as tracking. These estimates of motion and lighting can be used for novel view synthesis which will then be used for video-based face recognition in Section 1.5.

A simple method for estimating motion and illumination is by minimizing a cost function directly derived from (1.3) as

$$(\hat{\mathbf{l}}_t, \hat{\mathbf{m}}_t) = \arg\min_{\mathbf{l},\mathbf{m}} \|\mathcal{I}_t - \left(\mathcal{B}_{\hat{\mathbf{p}}_{t-1}} + \mathcal{C}_{\hat{\mathbf{p}}_{t-1}} \times_2 \mathbf{m}\right) \times_1 \mathbf{l}\|^2 + \alpha||\mathbf{m}||^2, \qquad (1.10)$$

where $\hat{x}$ denotes an estimate of x. Since the motion between consecutive frames is small, but illumination can change suddenly, we add a regularization term $\alpha||\mathbf{m}||^2$ to the above cost function. The estimates of motion and lighting can be obtained by alternate minimization along these two directions (this is a valid local minimization due to the bilinearity of the two terms) as

$$\hat{\mathbf{l}} = (\mathcal{B}_{\hat{\mathbf{p}}_{t-1}(1)} \mathcal{B}^T_{\hat{\mathbf{p}}_{t-1}(1)})^{-1} \mathcal{B}_{\hat{\mathbf{p}}_{t-1}(1)} \mathcal{I}^T_{t(1)}, \qquad (1.11)$$

and

$$\begin{aligned} \hat{\mathbf{m}} \quad = \quad & \left((\mathcal{C}_{\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l})_{(2)} (\mathcal{C}_{\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l})^T_{(2)} + \alpha \mathbf{I}\right)^{-1} \\ & (\mathcal{C}_{\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l})_{(2)} (\mathcal{I}_t - \mathcal{B}_{\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l})^T_{(2)}, \end{aligned} \qquad (1.12)$$

where $\mathbf{I}$ is an identity matrix of dimension 6×6.

This is essentially a model-based estimation approach that requires a texture-mapped 3D model of the object to be tracked. This is expected as the method works only with the analytically derived model which cannot represent variations of identity within a single class of objects. It was the approach presented by the authors of [28]. By using our GAMs, this restriction can be overcome. Moreover, we can achieve this in a computationally efficient manner by using the inverse compositional algorithm. As mentioned earlier, our tracking method is faster than 3DMM-based approaches [6], while sacrificing little in accuracy.

1.4.1 Inverse Compositional (IC) Estimation on GAMs

The iteration involving alternate minimization over motion and illumination in the above approach is essentially a gradient descent method. In each iteration, as pose is updated, the gradients, i.e. the tensors $\mathcal{B}$ and $\mathcal{C}$, need to be recomputed, which is computationally expensive. The inverse compositional algorithm [3] works by moving these computational steps out of the iterative updating process. In addition, the constraint of knowing the 3D model of the object can be relaxed by reconstructing $\mathcal{B}$ and $\mathcal{C}$ from the core tensors $\mathcal{Z}^{\mathcal{B}}$ and $\mathcal{Z}^{\mathcal{C}}$.

From equation (1.9), the the cost function for estimation of 3D motion and lighting in (1.10) can be rewritten as

$$(\hat{\mathbf{l}}_t, \hat{\mathbf{m}}_t) = \arg \min_{\mathbf{l},\mathbf{m},\mathbf{c}_i,\mathbf{c}_e} \left|\left| \mathcal{I}_t - \left(\mathcal{Z}^{\mathcal{B}}_{\hat{\mathbf{p}}_{t-1}} + \mathcal{Z}^{\mathcal{C}}_{\hat{\mathbf{p}}_{t-1}} \times_2 \mathbf{m} \right) \times_1 \mathbf{l} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e \right|\right|^2 + \alpha ||\mathbf{m}||^2. \tag{1.13}$$

This cost function is quadrilinear in illumination, motion, identity and deformation variables. The optimization of (1.13) can be done by alternatively optimizing over each parameter of $\mathbf{l}, \mathbf{m}, \mathbf{c}_i$, and $\mathbf{c}_e$, while keeping the others fixed. This takes advantage of the fact that we know that the space is multilinear. Starting from an initial pose estimate (where the manifold is approximated by a tangent), we will first optimize over illumination, identity and expression dimensions, and then apply the inverse compositional algorithm for optimization over motion. In Figure 1.2, we show a pictorial scheme of this optimization process.

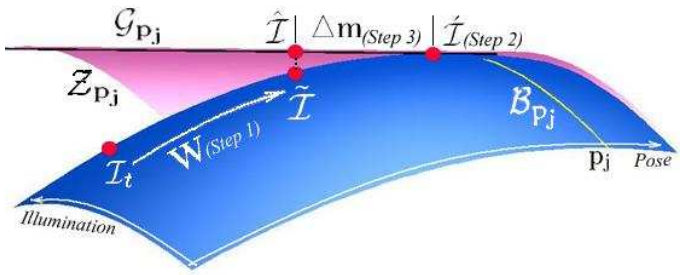


Fig. 1.2 Pictorial representation of the inverse compositional tracking scheme on GAMs.

IC Warping Function: Consider an input frame $I_t(\mathbf{u})$ at time instance t with image coordinate $\mathbf{u}$. We introduce a warp operator $\mathbf{W} : \mathbb{R}^{\mathbf{2}} \rightarrow \mathbb{R}^{\mathbf{2}}$ such that, if the pose of $I_t(\mathbf{u})$ is $\mathbf{p}$, the pose of $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{u}, \mathbf{m}))$ is $\mathbf{p} + \mathbf{m}$. Basically, $\mathbf{W}$ represents the displacement in the image plane due to a pose transformation of the 3D model. Denote the pose transformed image $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{u}, \mathbf{m}))$ in tensor notation $\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{m})}$. Using this warp operator and ignoring the regularization term, we can restate the cost

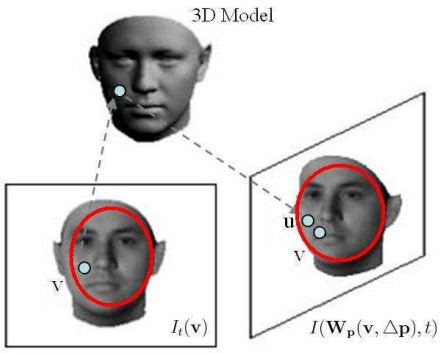


Fig. 1.3 Illustration of the warping function $\mathbf{W}$. A point $\mathbf{v}$ in image plane is projected onto the surface of the 3D object model. After the pose transformation with $\triangle\mathbf{p}$, the point on the surface is back projected onto the image plane at a new point $\mathbf{u}$. The warping function maps from $\mathbf{v} \in \mathbb{R}^\mathbf{2}$ to $\mathbf{u} \in \mathbb{R}^\mathbf{2}$. The red ellipses show the common part in both frames that the warping function $\mathbf{W}$ is defined upon.

function (1.13) in the inverse compositional framework as

$$(\hat{\mathbf{l}_t}, \hat{\mathbf{m}_t}) = \arg\min_{\mathbf{l},\mathbf{m},\mathbf{c}_i,\mathbf{c}_e} \|\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(-\mathbf{m})} - \mathcal{Z}^{\mathcal{B}}_{\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e\|^2 + \alpha||\mathbf{m}||^2. \tag{1.14}$$

Given the other parameters of the quadrilinear manifold, the cost function can be minimized over $\mathbf{m}$ by iteratively solving for increments $\triangle\mathbf{m}$ in

$$\|\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(-\mathbf{m})} - \left(\mathcal{Z}^{\mathcal{B}}_{\hat{\mathbf{p}}_{t-1}} + \mathcal{Z}^{\mathcal{C}}_{\hat{\mathbf{p}}_{t-1}} \times_2 \triangle\mathbf{m}\right) \times_1 \mathbf{l} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e\|^2 + \alpha||\mathbf{m} + \triangle\mathbf{m}||^2 \tag{1.15}$$

In each iteration, $\mathbf{m}$ is updated such that $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, -\mathbf{m}) \leftarrow \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, -\mathbf{m}) \circ \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, \triangle\mathbf{m})^{-1}$. [3] Using the additivity of pose transformation for small $\triangle\mathbf{m}$, $\mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(\mathbf{u}, \triangle\mathbf{m})^{-\mathbf{1}}, -\mathbf{m}) = \mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(\mathbf{u}, -\triangle\mathbf{m}), -\mathbf{m}) = \mathbf{W}_{\hat{\mathbf{p}}_\mathbf{t-1}}(\mathbf{u}, -\triangle\mathbf{m} - \mathbf{m})$. Thus, the above update is essentially $\mathbf{m} \leftarrow \mathbf{m} + \triangle\mathbf{m}$.

In [3], the authors proved that, for the inverse compositional algorithm to be provably equivalent to the Lucas-Kanade algorithm to the first order approximation of $\triangle\mathbf{m}$, the set of warps $\{\mathbf{W}\}$ must form a group, i.e. every warp $\mathbf{W}$ must be invertible. If the change of pose is small enough, the visibility for most of the pixels will remain the same - thus $\mathbf{W}$ can be considered approximately invertible. However, if the pose change becomes too big, some portion of the object will become invisible after the pose transformation, and $\mathbf{W}$ will no longer be invertible.

[3] The compositional operator $\circ$ means the second warp is composed into the first warp, i.e. $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, -\mathbf{m}) \equiv \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, \triangle\mathbf{m})^{-1}, -\mathbf{m})$.

Since the GAM along the motion direction is composed of a set of tangent planes at a few discrete poses (see Figure 1.1), the computations for $\triangle\mathbf{m}$ need to happen only at these poses (called cardinal poses). Thus all frames that are close to a particular pose $\mathbf{p}_j$ will use the $\mathcal{B}$ and $\mathcal{C}$ at that pose, and the warp $\mathbf{W}$ should be performed to normalize the pose to $\mathbf{p}_j$. While most of the existing inverse compositional methods move the expensive update steps out of the iterations for two-frame matching, we go even further and perform these expensive computations only once every few frames. This is by virtue of the fact that we estimate 3D motion.

1.4.1.1 The IC Algorithm on GAMs Consider a sequence of image frames $\mathcal{I}_t$, $t = 0, ..., N-1$.
Assume that we know the pose and illumination estimates for frame $t-1$, i.e., $\hat{\mathbf{p}}_{t-1}$ and $\hat{\mathbf{l}}_{t-1}$.
• *Step 1*. For the new input frame $\mathcal{I}_t$, find the closest $\mathbf{p_j}$ to the pose estimates at $t-1$, i.e. $\hat{\mathbf{p}}_{t-1}$. Assume motion $\mathbf{m}$ to be zero, and illumination condition $\hat{\mathbf{l}}_t = \hat{\mathbf{l}}_{t-1}$. Apply the pose transformation operator $\mathbf{W}$ to get the pose normalized version of the frame $\tilde{\mathcal{I}}^{\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t}-\mathbf{1}}}(\mathbf{p_j}-\hat{\mathbf{p}}_{\mathbf{t}-\mathbf{1}}-\mathbf{m})}$, i.e., $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, \mathbf{p_j} - \hat{\mathbf{p}}_{t-1} - \mathbf{m}))$. This is shown in Figure 1.2, where the input frame $\mathcal{I}_t$ on the manifold is first warped to $\tilde{\mathcal{I}}$ which is within a nearby region of pose $\mathbf{p_j}$.
• *Step 2*. Use (18) to alternately estimate $\hat{\mathbf{l}}$, $\hat{\mathbf{c}}_i$ and $\hat{\mathbf{c}}_e$ of the pose normalized image $\mathcal{I}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u},\mathbf{p_j}-\hat{\mathbf{p}}_{t-1}-\mathbf{m})}$ as follows.
Using (1.8), $\mathcal{B}_{\mathbf{p_j}}$ can be written as

$$\mathcal{B}_{\mathbf{p_j}} = \left[\mathcal{Z}^{\mathcal{B}}_{\mathbf{p_j}} \times_3 \mathbf{c}_i \times_4 \mathbf{c}_e\right]_v^{-1}. \tag{1.16}$$

Denoting the basis for the identity and expression as $\mathcal{E}$ and $\mathcal{F}$, we can similarly compute them as

$$\begin{aligned}\mathcal{E}_{\mathbf{p_j}} &= \left[\mathcal{Z}^{\mathcal{B}}_{\mathbf{p_j}} \times_1 \mathbf{l} \times \mathbf{c}_e\right]_v^{-1},\\ \mathcal{F}_{\mathbf{p_j}} &= \left[\mathcal{Z}^{\mathcal{C}}_{\mathbf{p_j}} \times_1 \mathbf{l} \times_3 \mathbf{c}_i\right]_v^{-1}.\end{aligned} \tag{1.17}$$

Thus the illumination coefficients can be estimated using least squares (since the illumination bases after motion (1.2) are not orthogonal), while the identity and expression coefficients can be estimated by projection of the image onto the corresponding basis as

$$\begin{aligned}\hat{\mathbf{l}} &= (\mathcal{B}_{\mathbf{p}_j}\mathcal{B}^{\mathbf{T}}_{\mathbf{p}_j})^{-1}\mathcal{B}^{\mathbf{T}}_{\mathbf{p}_j}\mathcal{I}_{(1)},\\ \hat{\mathbf{c_i}} &= \mathcal{E}^{\mathbf{T}}_{\mathbf{p}_j} I_{(1)}, \hat{\mathbf{c_e}} = \mathcal{F}^{\mathbf{T}}_{\mathbf{p}_j} I_{(1)}.\end{aligned} \tag{1.18}$$

Iteratively solving for $\hat{\mathbf{l}}$, $\hat{\mathbf{c}}_i$ and $\hat{\mathbf{c}}_e$, the cost function(1.14) is minimized over illumination, identity and expression directions. In Figure 1.2, the curve $\mathcal{B}_{\mathbf{p_j}}$ shows the

manifold of the image at pose $\mathbf{p_j}$ with motion as zero, but varying illumination, identity or deformation. By iteratively minimizing along the illumination, identity, and deformation directions, we are finding the point

$$\acute{\mathcal{I}} = \mathcal{Z}^{\mathcal{B}}_{\mathbf{p}_j} \times_1 \hat{\mathbf{l}} \times_3 \hat{\mathbf{c}}_i \times_4 \hat{\mathbf{c}}_e \tag{1.19}$$

on the curve $\mathcal{B}_{\mathbf{p_j}}$ which has the minimum distance to the pose normalized point $\tilde{\mathcal{I}}$.

• *Step 3*. With the estimated $\hat{\mathbf{l}}$, $\hat{\mathbf{c}}_i$ and $\hat{\mathbf{c}}_e$ from Step 2, use (1.21) to estimate the motion increment $\triangle\mathbf{m}$. Update $\mathbf{m}$ with $\mathbf{m} \leftarrow \mathbf{m} + \triangle\mathbf{m}$. This can be done as follows.

Rewrite the cost function in (1.15) at the cardinal pose $\mathbf{p}_j$ as

$$\left|\left|\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{p}_j - \hat{\mathbf{p}}_{t-1} - \mathbf{m})} - \left(\acute{\mathcal{I}} + \mathcal{G}^{\mathbf{T}}_{\mathbf{p_j}} \triangle\mathbf{m}\right)\right|\right|^2 + \alpha||\mathbf{m} + \triangle\mathbf{m}||^2,$$
$$\text{where } \mathcal{G}_{\mathbf{p_j}} = \left[\mathcal{Z}^{\mathcal{C}}_{\mathbf{p_j}} \times_1 \hat{\mathbf{l}} \times \hat{\mathbf{c}}_i \times \hat{\mathbf{c}}_e\right]^{-1}_v. \tag{1.20}$$

$\mathcal{G}_{\mathbf{p_j}}$ is the motion basis at pose $\mathbf{p_j}$ with fixed $\hat{l}$, $\hat{\mathbf{c}}_i$ and $\hat{\mathbf{c}}_e$. Recall that $\mathcal{Z}^{\mathcal{C}}_{\mathbf{p_j}}$ is a tensor of size $N_l \times 6 \times N_i \times N_e \times MN$ - thus $\mathcal{G}_{\mathbf{p_j}}$ degenerates to a matrix of size $6 \times MN$. In Figure 1.2, we compute the tangent along the motion direction, shown as the black line $\mathcal{G}_{\mathbf{p_j}}$, from the core tensor shown as the pink surface $\mathcal{Z}$.

Taking the derivative of (1.20) with respect to $\triangle\mathbf{m}$, and setting it to be zero, we have

$$\triangle\mathbf{m} = \left[\mathcal{G}_{\mathbf{p_j}}\mathcal{G}^{\mathbf{T}}_{\mathbf{p_j}} + \alpha\mathbf{I}\right]^{-1}(\mathcal{G}_{\mathbf{p_j}}(\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{p}_j - \hat{\mathbf{p}}_{t-1} - \mathbf{m})} - \acute{\mathcal{I}}) - \alpha\mathbf{m}), \tag{1.21}$$

and the motion estimates $\mathbf{m}$ should be updated with the increments $\mathbf{m} \leftarrow \mathbf{m} + \triangle\mathbf{m}$. The overall computational cost is reduced significantly by making the gradient $\mathcal{G}_{\mathbf{p_j}}$ independent of the updating variable $\mathbf{m}$. In Figure 1.2, $\triangle\mathbf{m}$ is shown to be the distance from point $\acute{\mathcal{I}}$ to $\hat{\mathcal{I}}$, the projection of $\tilde{\mathcal{I}}$, onto the motion tangent.

• *Step 4*. Use the updated $\mathbf{m}$ from Step 3 to update the pose normalized image as $\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{p_j} - \hat{\mathbf{p}}_{\mathbf{t-1}} - \mathbf{m})}$, i.e. $I(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{u}, \mathbf{p_j} - \hat{\mathbf{p}}_{t-1} - \mathbf{m}), t)$.

• *Step 5*. Repeat Steps 2, 3 and 4 for that input frame till the difference error ε between the pose normalized image $\tilde{\mathcal{I}}_t^{\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{p_j} - \hat{\mathbf{p}}_{t-1} - \mathbf{m})}$ and the rendered image $\acute{\mathcal{I}}$ can be reduced below an acceptable threshold.

• Step 6. Set t = t + 1. Repeat Steps 1, 2, 3, 4 and 5. Continue till t = N - 1.

1.4.2 Probabilistic IC (PIC) Estimation

To ensure that the tracking is robust to estimation errors, we embed the IC approach within a probabilistic framework. For ease of explanation, let us denote the current cardinal pose to be $\mathbf{p}_j$, and the nearby cardinal poses as $\mathbf{p}_{j-1}$ and $\mathbf{p}_{j+1}$. Denote the nearest-neighbor partition region on the multilinear manifold for cardinal pose

$\mathbf{p}_j$ to be $\Theta_{\mathbf{p}_j}$. Given the estimated pose at the previous time instance $\hat{\mathbf{p}}_{t-1}$, the average velocity $\bar{\mathbf{m}}$ and variation σ_m^2 of it within a recent history, we can model the distribution of the current pose $\mathbf{p}_t \sim \mathcal{N}(\hat{\mathbf{p}}_{t-1} + \bar{\mathbf{m}}, \sigma_m^2)$, where $\mathcal{N}$ is the normal distribution. Assume the likelihood distribution of ε at pose $\mathbf{p}_j$ (difference between pose normalized image and rendered image) in step (5) of the inverse compositional algorithm is $p(\varepsilon|\mathbf{p}_t \in \Theta_{\mathbf{p}_j}) \sim \mathcal{N}(0, \sigma)$. Using Bayes rule, we get

$$P(\mathbf{p}_t \in \Theta_{\mathbf{p}_j}|\varepsilon) = \frac{p(\varepsilon|\mathbf{p}_t \in \Theta_{\mathbf{p}_j}) \int_{\Theta_{\mathbf{p}_j}} p(\mathbf{p}_t) d\mathbf{p}_t}{P\varepsilon}. \tag{1.22}$$

Similarly, we can compute $P(\mathbf{p}_t \in \Theta_{\mathbf{p}_{j-1}}|\varepsilon)$ and $P(\mathbf{p}_t \in \Theta_{\mathbf{p}_{j+1}}|\varepsilon)$. Denoting the estimate of motion, illumination, identity, and expression with the tangent at $\mathbf{p}_j$ as $\hat{\mathbf{x}}_t^{\mathbf{p}_j}$, the final estimate can be obtained as

$$\hat{\mathbf{x}}_t = E(\mathbf{x}_t|\varepsilon) = \frac{\sum_{i=j-1}^{j+1} \hat{\mathbf{x}}_t^{\mathbf{p}_i} P(\mathbf{p}_t \in \Theta_{\mathbf{p}_i}|\varepsilon)}{\sum_{i=j-1}^{j+1} P(\mathbf{p}_t \in \Theta_{\mathbf{p}_i}|\varepsilon)}. \tag{1.23}$$

1.5 FACE RECOGNITION FROM VIDEO

We now explain the video-based face recognition algorithm using GAMs. The use of GAMs is motivated by the fact that in video we will encounter changes of pose, lighting and appearance.

In our method, the gallery is represented by a textured 3D model of the face. The model can be built from a single image [6], a video sequence [22] or obtained directly from 3D sensors [7]. In our experiments, the face model will be estimated from the gallery video sequence for each individual. Face texture is obtained by normalizing the illumination of the first frame in the gallery sequence to an ambient condition, and mapping it onto the 3D model. Given a probe sequence, we will estimate the motion and illumination conditions using the algorithms described in Section 1.4. Note that the tracking does not require a person-specific 3D model - a generic face model is usually sufficient. Given the motion and illumination estimates, we will then render images from the 3D models in the gallery. The rendered images can then be compared with the images in the probe sequence. For this purpose, we will design robust measurements for comparing these two sequences. A feature of these measurements will be their ability to integrate the identity over all the frames, ignoring some frames that may have the wrong identity.

Let $I_i, i = 0, ..., N-1$ be the ith frame from the probe sequence. Let $S_{i,j}, i = 0, ..., N-1$ be the frames of the synthesized sequence for individual j, where $j = 1, ..., M$ and M is the total number of individuals in the gallery. Note that the number of frames in the two sequences to be compared will always be the same in our method. By design, each corresponding frame in the two sequences will be under the same pose and illumination conditions, dictated by the accuracy of the estimates of these parameters from the probes sequences. Let d_{ij} be the Euclidean

distance between the i^{th} frames I_i and $S_{i,j}$. Then we obtain the identity of the probe as

$$ID = \arg\min_j \min_i d_{ij}. \tag{1.24}$$

The measurement in (1.24) computes the distance between the frames in the probe sequence and each synthesized sequence that are the most similar and chooses the identity as the individual with the smallest distance.

As the images in the synthesized sequences are pose and illumination normalized to the ones in the probe sequence, d_{ij} can be computed directly using the Euclidean distance. Other distance measurements, like [11, 17], can be considered in situations where the pose and illumination estimates may not be reliable or in the presence of occlusion and clutter. We will look into such issues in our future work.

1.5.1 Video-Based Face Recognition Algorithm:

Using the above notation, let $I_i, i = 0, ..., N-1$ be N frames from the probe sequence. Let $G_1, ..., G_M$ be the 3D models with texture for each of M galleries.
• **Step 1.** Register a 3D generic face model to the first frame of the probe sequence. This is achieved using the method in [29][4]. Estimate the illumination and motion model parameters for each frame of the probe sequence using the method described in Section 1.4.1.1.
• **Step 2.** Using the estimated illumination and motion parameters, synthesize, for each gallery, a video sequence using the generative model of (1.1). Denote these as $S_{i,j}, i = 1, ..., N$ and $j = 1, ..., M$.
• **Step 3.** Compute d_{ij} as above.
• **Step 4.** Obtain the identity using a suitable distance measure as in (1.24).

1.6 EXPERIMENT RESULTS

As discussed above, the advantages of using the GAMs are (i) ease of construction due to the need for significantly less number of training images, (ii) ability to represent objects at all poses and lighting conditions from only a few examples during training, and (iii) accuracy and efficiency of tracking and recognition. We will now show results to justify these claims.

[4]We use a semi-automatic registration algorithm to initialize the IC tracking. It requires first manually choosing seven landmark points, followed by automatically registering the 3D face model onto the image to estimate the initial pose.

1.6.1 Constructing GAM of faces:

In the case of faces, we will need at least one image for every person. We then fit the 3DMM to estimate the face model and compute the vectorized tensor $\mathbf{v}$ at a predefined collection of poses $\mathbf{p}_j$. For each expression, we will need at least one image per person. Thus for N_i people with N_e expressions, we need $N_i N_e$ images. In our experiments, $N_i = 100$ and $N_e = 7$ thus requiring 700 images for all the people and every expression. In contrast, [14] requires 300 frames per person for training purposes while modeling only pose variation. Similarly, in [23], 225 frames of 15 poses and 15 under different illumination patterns are used for each identity (expression variation is not considered). Moreover, the GAM can model the appearance space not only at these discrete poses, but also the manifold in a local region around each pose. In our experiments, the pose collection $\mathbf{p}_j$ is chosen to be every 15° along the vertical rotational axis, and every 20° along the horizontal rotational axis. In Figure 1.4, we show some basis images of the face GAM along illumination, 3D motion, identity and expression dimensions. As we can show only 3 dimensions, identity is fixed to one particular person.

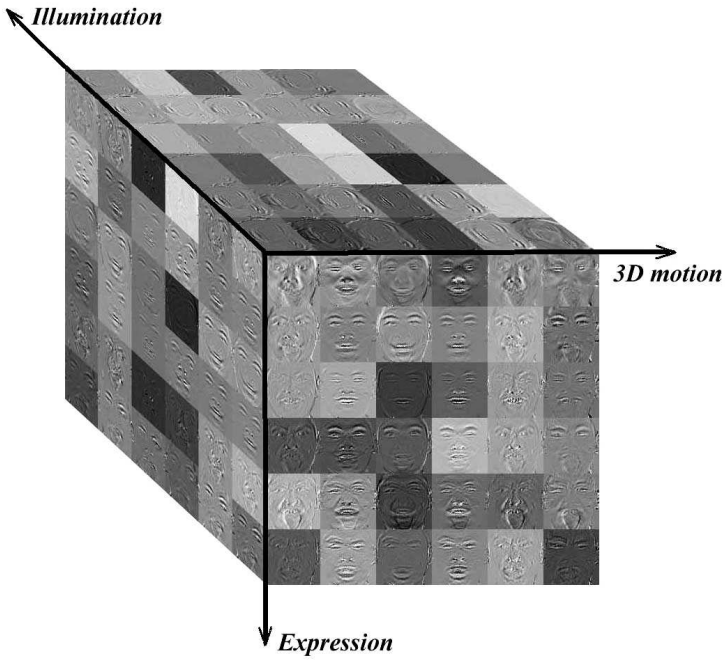


Fig. 1.4 The basis images of the face GAM on illumination, expression, and the 3D motion around the frontal cardinal pose for a specific person.

1.6.2 Accuracy of the motion and illumination estimates on GAM

We will now show some results on the accuracy of tracking on the GAM with known ground truth. We use the 3DMM [6] to randomly generate a face. The generated face model is rotated along the vertical axis at some specific angular velocity, and the illumination is changing both in direction (from right-bottom corner to the left-top corner) and in brightness (from dark to bright to dark). In Figure 1.5, the images show the back projection of some feature points on the 3D model onto the input frames using the estimated motion under three different illumination conditions. In Figure 1.6, (a) shows the comparison between the estimated motion (in blue) and the ground truth (in red). The maximum error in pose estimates is 3.57° and the average error is 1.22°. Figure 1.6 (b) shows the norm of the error between the ground truth illumination coefficients and the estimated ones from the GAM, normalized with the ground truth. The maximum error is 5.5% and the average is 2.2%. The peaks in the error plot are due to the change of the cardinal pose $\mathbf{p}_j$ (the tangent planes along the pose dimension).

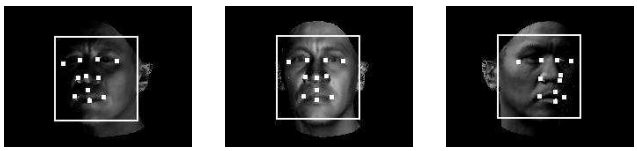
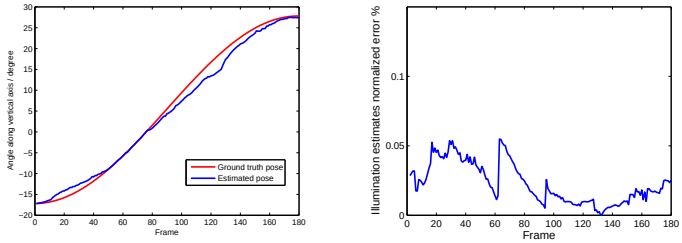


Fig. 1.5 The back projection of the feature points on the generated 3D face model using the estimated 3D motion onto some input frames.



(a) (b)

Fig. 1.6 (a): 3D estimates (blue) and ground truth (red) of pose against frames. (b): The normalized error of the illumination estimates vs. frame numbers.

1.6.3 PIC Tracking on GAM using Real Data

Figure 1.7 shows results of face tracking under large changes of pose, lighting, expression and background using the PIC approach. The images in the first row show tracking under illumination variations with global and local changes. The images

Fig. 1.7 Examples of face tracking using GAMs under changes of pose, lighting and expressions.

in the second row show tracking on the GAM with some expressions under varying illumination conditions. We did not require a texture-mapped 3D model as in [28]. Compared to 3DMM, we achieve almost the same accuracy while requiring one-tenth the computational time per frame.

1.6.4 Face Database and Experimental Setup

Our database consists of videos of 57 people. Each person was asked to move his/her head as they wished (mostly rotate their head from left to right, and then from down to up), and the illumination was changed randomly. The illumination consisted of ceiling lights, lights from the back of the head and sunlight from a window on the left side of the face. Random combinations of these were turned on and off and the window was controlled using dark blinds. There was no control over how the subject moves his/her head or on facial expression. An example of some of the images in the video database is shown in Figure 1.8. The images are scale normalized and centered. Some of the subjects had expression changes also, e.g., the last row of the Figure 1.8. The average size of the face was about 70 x 70, with the minimum size being 50 x 50. Videos are captured with uniform background. We recorded 2 to 3 sessions of video sequences for each individual. All the video sessions are recorded within one week. The first session is used as the gallery for constructing the 3D textured model of the head, while the remaining are used for testing. We used a simplified version of the method in [22] for this purpose. We would like to emphasize that any other 3D modeling algorithm would also have worked. Texture is obtained by normalizing the illumination of the first frame in each gallery sequence to an ambient illumination condition, and mapping onto the 3D model.

As can be seen from Figure 1.8, the pose and illumination varies randomly in the video. For each subject, we designed three experiments by choosing different probe sequences:
Expt. A: A video was used as the probe sequence with the average pose of the face in the video being about 15° from frontal;
Expt. B: A video was used as the probe sequence with the average pose of the face in the video being about 30° from frontal;
Expt. C: A video was used as the probe sequence with the average pose of the face

Fig. 1.8 Sample frames from the video sequence collected for our database (best viewed on a monitor).

in the video being about 45° from frontal.

Each probe sequence has about 20 frames around the average pose. The variation of pose in each sequence was less than 15°, so as to keep pose in the experiments disjoint. To show the benefit of video-based methods over image-based approaches, we designed three new Expts. D, E and F by taking random single images from A, B and C respectively. We restricted our face recognition experiments to the pose and illumination variations only (which can be expressed analytically), with the bilinear representation of [27].

1.6.5 Recognition Results

We plot the Cumulative Match Characteristic (CMC) [31, 19] for experiments A, B, and C with measurement (1.24) in Figure 1.9. Our proposed algorithm gives relatively high performance. In Expt. A, where pose is 15° away from frontal, all the videos with large and arbitrary variations of illumination are recognized correctly. In Expt. B, we achieve about 95% recognition rate, while for Expt. C it is 93% using the distance measure (1.24). Irrespective of the illumination changes, the recognition rate decreases consistently with large difference in pose from frontal (which is the gallery), a trend that has been reported by other authors [5, 30]. *Note that the pose and illumination conditions in the probe and gallery sets can be completely disjoint.*

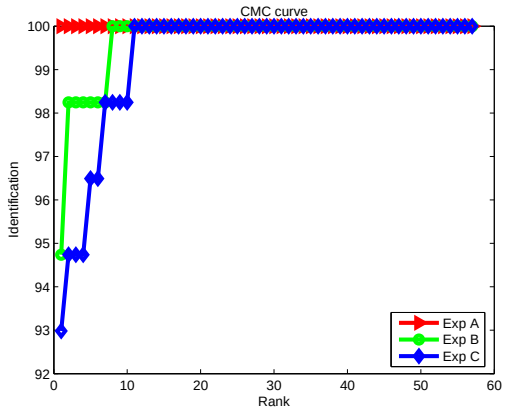


Fig. 1.9 CMC curve for video-based face recognition experiments A to C with distance measure in (1.24).

1.6.6 Comparison with other Approaches

The area of video-based face recognition is less standardized than image-based approaches. There is no standard dataset on which both image and video-based methods have been tried, thus we do the comparison on our own dataset. This dataset can be used for such comparison by other researchers in the future.

1.6.7 Comparison with 3DMM based approaches

3DMM has achieved a significant impact in the biometrics area, and obtained impressive results in pose and illumination varying face recognition. It is similar to our proposed approach in the sense that both methods are 3D approaches, estimate the pose, illumination, and do synthesis for recognition. However, 3DMM method uses the Phong illumination model, thus it cannot model extended light sources (like the sky) accurately. To overcome this, Samaras etc. [30] proposed the SHBMM (3D Shperical Harmonics Basis Morphable Model) that integrates the spherical harmonics illumination representation into the 3DMM. Although it is possible to repeatedly apply 3DMM or SHBMM approach to each frame in the video sequence, it is inefficient. Registration of the 3D model to each frame will be needed, which requires a lot of computation and manual work. None of the existing 3DMM approaches integrate tracking and recognition. Also, 3DMM-based methods cannot achieve real-time pose/illumination estimation, which can be achieved with the inverse compositional version of our tracking method. Our proposed method, which integrates 3D motion into SHBMM, is a unified approach for modeling lighting and motion in a video sequence.

We now compare our proposed approach against the SHBMM method of [30], which were shown give better results than 3DMM in [5]. We will also compare our

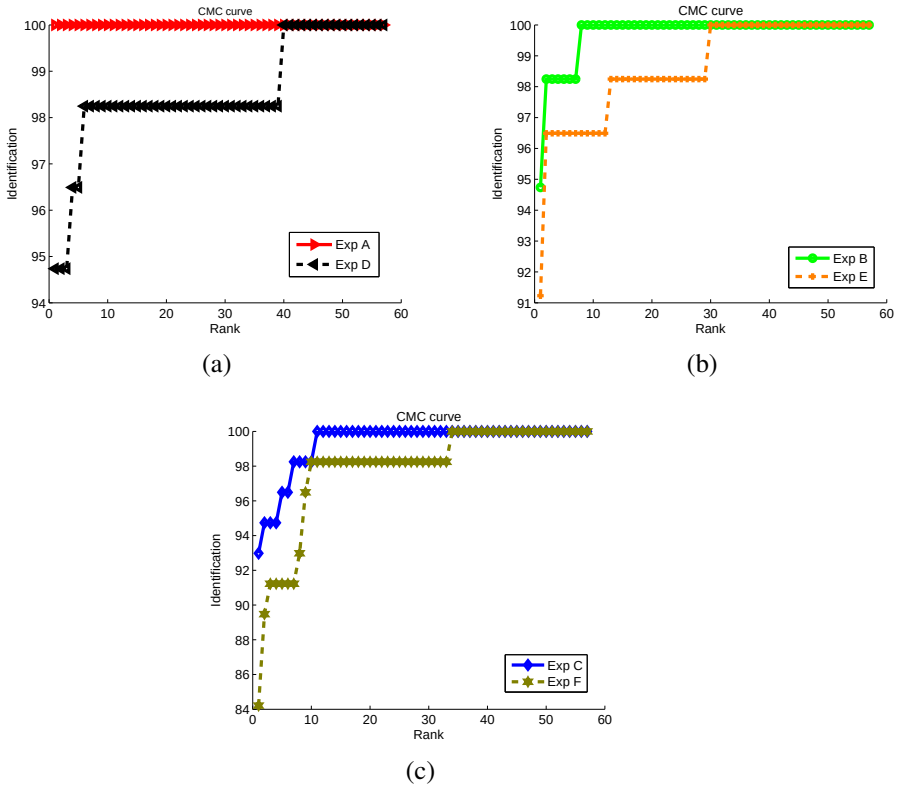


Fig. 1.10 Comparison between the CMC curves for the video-based face experiments A to C with distance measurement (1.24) against SHBMM method of [30].

results with the published results of SHBMM method [30] in the later part of this section.

Recall that we designed three new Expts. D, E and F by taking random single images from A, b and C respectively. In Figure 1.10, we plot the CMC curve with measurement 1 in equation (1.24) (which has the best performance for Expt. A, B and C) for the Expts. D, E, F and compare them with the ones of the Expt. A, B, and C. For this comparison, we randomly chose images from the probe sequences of Expts. A, B, C and computed the recognition performance over multiple such random sets. Thus the Expts. D, E and F average the image-based performance over different conditions. By analyzing the plots in Figure 1.10, we see that the recognition performance with the video-based approach is consistently higher than the image-based one, both in Rank 1 performance as well as the area under the CMC curve. This trend is magnified as the average facial pose becomes more non-frontal. Also, we expect that registration errors, in general, will affect image-based methods more than video-based methods (since robust tracking maybe able to overcome some of the registration errors, as shown in section 4.4).

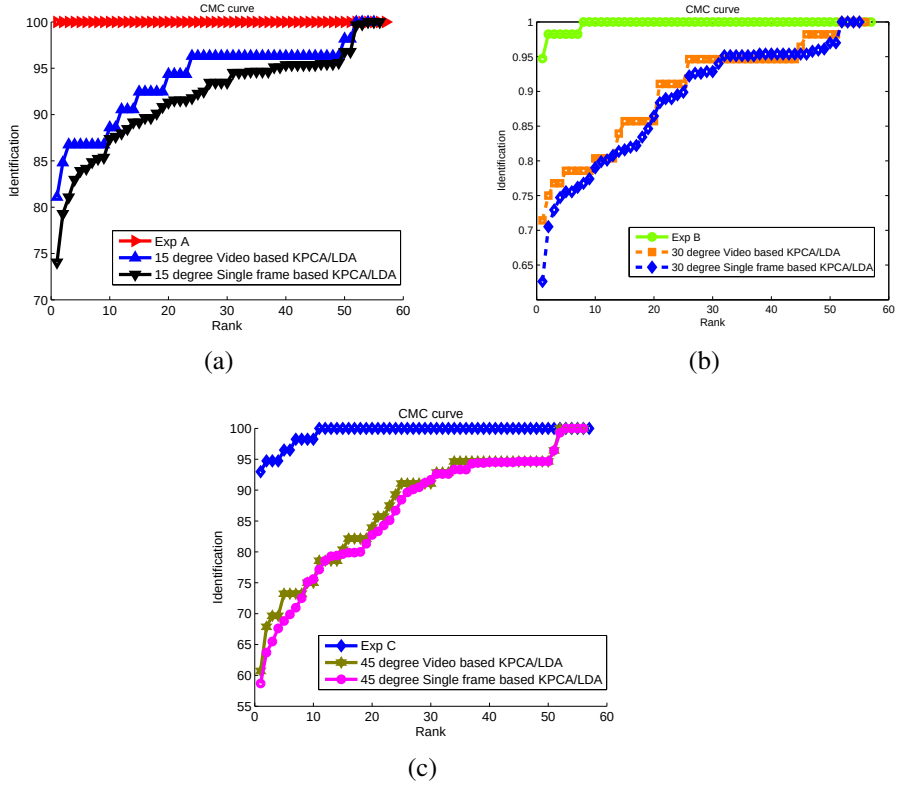


Fig. 1.11 Comparison between the CMC curves for the video-based face experiments A to C with distance measurement in (1.24) against KPCA+LDA based 2D approaches [2].

It is interesting to compare these results against the results in [30], for image-based recognition. The size of the databases in both cases is close (though ours is slightly smaller). Our recognition rate with a video sequence at average 15 degrees facial pose (with a range of 15 degrees about the average) is 100%, while the average recognition rate for approximately 20 degrees (called side view) in [30] is 92.4%. For the Exp. B and C, [30] does not have comparable cases and goes directly to profile pose (90 degrees), which we don't have. Our recognition rate at 45° average pose is 93%. In [30], the quoted rates at 20° is 92% and at 90° is 55%. Thus the trend of our video-based recognition results are significantly higher than image-based approaches that deal with both pose and illumination variations.

1.7 CONCLUSION

In this chapter, we showed how to combine geometrical and statistical models for video-based face recognition. We showed that it is possible to estimate low-dimensional

manifolds that describe object appearance with a small number of training samples using a combination of analytically derived geometrical models and statistical data analysis. We derived a quadrilinear space of object appearance that is able to represent the effects of illumination, motion, identity and deformation, and termed it as the Geometry-Integrated Appearance Manifold. Based upon the GAM, we have proposed a method for video-based face recognition. We also collected a face video database consisting of 57 people with large and arbitrary variation in pose and illumination, and demonstrated the effectiveness of the method on this new database. We showed specific examples on how to construct this manifold, analyzed the accuracy of the pose and lighting estimates, and presented the the video-based face recognition results upon our own dataset. Detailed analysis of recognition performance are also carried out. Future work will focus on extending GAMs to objects with large deformations and its application in video-based face recognition with large expression variations.

REFERENCES

1. Ognjen Arandjelovic and Roberto Cipolla. An illumination invariant face recognition system for access control using video. *British Machine Vision Conference*, 2004.

2. A.J. Smola B. Scholkopf and K.-R. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10:1299-1319, 1998.*, 1998.

3. S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, February 2004.

4. R. Basri and D.W. Jacobs. Lambertian Reflectance and Linear Subspaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(2):218–233, February 2003.

5. V. Blanz, P. Grother, P. Phillips, and T. Vetter. Face Recognition Based on Frontal Views Generated From Non-Frontal Images. In *Computer Vision and Pattern Recognition*, 2005.

6. V. Blanz and T. Vetter. Face recognition based on fitting a 3D morphable model. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(9):1063–1074, September 2003.

7. K.W. Bowyer and Chang. A survey of 3D and Multimodal 3D+2D Face Recognition. In *Face Processing: Advanced Modeling and Methods*. Academic Press, 2005.

8. T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active Appearance Models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001.

9. A.M. Elgammal and C.S. Lee. Separating style and content on a nonlinear manifold. In *Computer Vision and Pattern Recognition*, pages I: 478–485, 2004.

10. D. Freedman and M. Turek. Illumination-Invariant Tracking via Graph Cuts. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

11. Trevor Darrell Gregory Shakhnarovich, John W. Fisher. Face recognition from long-term observations. *European Conference on Computer Vision*, 2002.

12. G. D. Hager and P.N. Belhumeur. Efficient Region Tracking With Parametric Models of Geometry and Illumination. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, 1998.

13. L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A Multillinear Singular Value Decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

14. K.C. Lee, J. Ho, M.H. Yang, and D.J. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *Computer Vision and Pattern Recognition*, pages I: 313–320, 2003.

15. Xiaoming Liu and Tsuhan Chen. Video-based face recognition using adaptive hidden markov models. *IEEE Computer Vision and Pattern Recognition*, 2003.

16. Iain Matthews and Simon Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135 – 164, November 2004.

17. J. Fisher R. Cipolla O. Arandjelovic, G. Shakhnarovich and T. Darrell. Face recognition with image sets using manifold density divergence. *IEEE Computer Vision and Pattern Recognition*, 2005.

18. John Fisher Roberto Cipolla Ognjen Arandjelovic, Gregory Shakhnarovich and Trevor Darrell. Face recognition with image sets using manifold density divergence. *IEEE Computer Vision and Pattern Recognition*, 2005.

19. P.J. Phillips, P.J. Grother, R.J. Micheals, D.M. Blackburn, E. Tabassi, and J.M. Bone. Face recognition vendor test 2002: Evaluation report. Technical Report NISTIR 6965, http://www.frvt.org, 2003.

20. P. J. Phillips et al. Overview of the face recognition grand challenge. In *Computer Vision and Pattern Recognition*, 2005.

21. R. Ramamoorthi and P. Hanrahan. On the relationship between radiance and irradiance: determining the illumination from images of a convex Lambertian object. *Journal of the Optical Society of America A*, 18(10), Oct 2001.

22. A. Roy-Chowdhury and R. Chellappa. Face Reconstruction From Monocular Video Using Uncertainty Analysis and a Generic Model. *Computer Vision and Image Understanding*, 91(1-2):188–213, July-August 2003.

23. M.A.O. Vasilescu and D. Terzopoulos. Multilinear Independent Components Analysis. In *Computer Vision and Pattern Recognition*, 2005.

24. D. Vlasic, M. Brand, H. Pfister, and J. Popovic. Face transfer with multilinear models. *ACM Transactions on Graphics(TOG)*, pages 426–433, 2005.

25. H. Wang and N. Ahuja. Facial expression decomposition. *IEEE International Conference on Computer Vision*, 2:958 – 965, 2003.

26. C. Xie, B.V.K. Vijaya Kumar, S. Palanivel, and B. Yegnanarayana. A still-to-video face verification system using advanced correlation filters. In *First International Conference on Biometric Authentication*, 2004.

27. Y. Xu and A. Roy-Chowdhury. Integrating the Effects of Motion, Illumination and Structure in Video Sequences. In *Proc. of IEEE International Conference on Computer Vision*, 2005.

28. Y. Xu and A. Roy-Chowdhury. Learning illumination models while tracking. In *Third Intl. Symposium on 3D Processing, Visualization and Transmission*, 2006.

29. Y. Xu and A. Roy-Chowdhury. Pose and illumination invariant registration and tracking for video-based face recognition. *IEEE Computer Society Workshop on Biometrics (in association with CVPR)*, 2006.

30. L. Zhang and D. Samaras. Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 351–363, March 2006.

31. W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld. Face Recognition: A Literature Survey. *ACM Transactions*, 2003.

32. S. Zhou, V. Krueger, and R. Chellappa, Probabilistic recognition of human faces from video *Computer Vision and Image Understanding (CVIU) (special issue on Face Recognition)*, Vol. 91, pp. 214-245, 2003.

33. S. Zhou, R. Chellappa, and B. Moghaddam, Visual tracking and recognition using appearance-adaptive models in particle filters *IEEE Transactions on Image Processing (TIP)*, Vol. 11, pp. 1434- 1456, November 2004.