Robust Face Recognition from Multi-View Videos

Ming Du, *Student Member, IEEE,* Aswin C. Sankaranarayanan, *Member, IEEE,*
and Rama Chellappa, *Fellow, IEEE*

***Abstract*—Multi-view face recognition has become an active research area in the last few years. In this paper, we present an approach for video-based face recognition in camera networks. Our goal is to handle pose variations by exploiting the redundancy in multi-view video data. However, unlike traditional approaches that explicitly estimate the pose of the face, we propose a novel feature for robust face recognition in the presence of diffuse lighting and pose variations. The proposed feature is developed using the spherical harmonic representation of the face texture-mapped onto a sphere; the texture map itself is generated by back-projecting the multi-view video data. Video plays an important role in this scenario. First, it provides an automatic and efficient way for feature extraction. Second, the data redundancy renders the recognition algorithm more robust. We measure the similarity between feature sets from different videos using the Reproducing Kernel Hilbert Space. We demonstrate that the proposed approach outperforms traditional algorithms on a multi-view video database.**

***Index Terms*—Face recognition, Pose variations, Multi-camera networks, Spherical harmonics.**

I. Introduction

Single-view based object recognition is inherently affected by information loss that occurs during image formation. Although there exist many works addressing this problem, pose variation remains as one of the major nuisance factors for face recognition. In particular, self-occlusion of facial features, as the pose varies, raises fundamental challenges to designing robust face recognition algorithms. A promising approach to handle pose variations and its inherent challenges is the use of multi-view data.

In recent years, multi-camera networks have become increasingly common for biometric and surveillance systems. Having multiple viewpoints alleviates the drawbacks of a single viewpoint since the system has more information at its disposal. For example, in the context of face recognition, having multiple views increases the chances of the person being in a favorable frontal pose. However, to reliably and efficiently exploit the multi-view video data, we often need to estimate the pose of the person's head. This could be done explicitly by computing the actual pose of the person to a reasonable approximation, or implicitly by using a view selection algorithm. While there are many methods for multi-view pose estimation [1], [2], solving for the pose of a person's head is still a hard problem, especially when the resolution of the images is poor and the calibration of cameras (both external and internal) is not sufficiently precise to allow robust multi-view fusion. Such a scenario is especially true in the context of surveillance.

Face recognition using a multi-camera network is the focus of this paper. At this point, it is worth noting that the problem we study goes beyond face recognition across pose variations. In our setting, at a given time instant, we obtain multiple images of the face in different poses. Invariably these images could include a mix of frontal, non-frontal images of the face or in some cases, a mix of non-frontal images. This makes registration of the faces extremely important. Registration can be done once we decide to impose a 3D model onto the face. However, registration to a 3D model (essentially, aligning eyes to eyes, nose to nose, etc.) is very hard and computationally intensive for low-resolution imagery. Toward this end, we choose to use a spherical model of the face and a feature that is insensitive to pose variations.

In this paper, we propose a robust feature for multi-view recognition that is insensitive to pose variations[1]. For a given set of multi-view video sequences, we first use a particle filter to track the 3D location of the head using multi-view information. At each time instant or video frame, we then build the texture map associated with the face under the spherical model for the face. Given that we have the 3D location of the head from the tracking algorithm, we back-project the image intensity values from each of the views onto the surface of the spherical model, and construct a texture map for the whole face. We then compute a Spherical Harmonic (SH) transform of the texture map, and construct a robust feature that is based on the properties of the SH projection. Building rotational tolerances into our feature allows us to completely bypass the pose estimation step. For recognition with videos, we exploit the ensemble feature similarity which is measured by the limiting Bhattacharyya distance of features in the Reproducing Kernel Hilbert Space. The proposed approach outperforms traditional features and algorithms on a multi-view video database collected using a camera network.

The rest of this paper is organized as follows: We first discuss related work in Section II. In Section III, we review relevant SH theory and propose the robust feature. A particle filtering framework for multi-camera tracking and texture mapping is then described in Section IV. We then present a video-based recognition scheme in Section V. Finally, experimental results are presented in Section VI and conclusions in Section VII.

M. Du and R. Chellappa are with the Center for Automation Research, University of Maryland, College Park, MD, 20742 USA. e-mail: {mingdu, rama}@umiacs.umd.edu.

A. C. Sankaranarayanan is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA. e-mail: saswin@andrew.cmu.edu

[1]In many contexts such as camera pose estimation, pose typically refers to the 3D translation and 3D rotation of the camera/object. However, in face recognition, pose typically refers only to the 3D rotation of face with respect to a reference orientation. We follow this nomenclature. For most of this paper, we use the term pose and rotation interchangeably.

II. Related Work

The term *multi-view face recognition,* in a strict sense, only refers to situations where multiple cameras acquire the subject (or scene) simultaneously and an algorithm collaboratively utilizes the acquired images/videos. But the term has frequently been used to recognize faces across pose variations. This ambiguity does not cause any problem for recognition with (still) images; a group of images simultaneously taken with multiple cameras and those taken with a single camera but at different view angles are equivalent as far as pose variations are concerned. However, in the case of video data, the two cases diverge. While a multi-camera system guarantees the acquisition of multi-view data at any moment, the chance of obtaining the equivalent data by using a single camera is unpredictable. Such differences become vital in non-cooperative recognition applications such as surveillance. For clarity, we shall call the multiple video sequences captured by synchronized cameras a multi-view video, and the monocular video sequence captured when the subject changes pose, a single-view video. With the prevalence of camera networks, multi-view surveillance videos have become more and more common. Nonetheless, most existing multi-view video face recognition algorithms exploit single-view videos.

Still image-based recognition: There is a large body of research on still image-based multi-view face recognition. Existing algorithms include those based on view synthesis [3]–[7], 3D model construction [8]–[10], subspace or manifold analysis [11]–[13], regularized regression [14], stereo matching [15], [16] and local feature matching [17]–[21]. In recent years, local patch/feature based approaches have become popular due to their effectiveness in handling pose variations. Cao et al. [22] compare the local descriptors in a pose-adaptive way: they estimate the poses of the pair of input faces images and select an SVM classifier customized for that pose combination to perform verification. Yin et al. [23] generate a collection of generic intra-person variations for local patches. Given a pair of face images to verify, they look up in the collection to "align" the face part's appearance in one image to the same pose and illumination of the other image. This method will also require the poses and illumination conditions to be estimated for both face images. This "generic reference set" idea has also been used to develop the holistic matching algorithm in [24], where the ranking of look-up results forms the basis of matching measure. There are also works which handles pose variations implicitly without estimating the pose explicitly. For example, by modeling the location-augmented local descriptors using a Gaussian Mixture Model, Li et al. [25] perform probabilistic elastic matching on a pair of face images even when large pose variations exhibit.

Video-based recognition: Video contains more information than still images. A straightforward way to handle single-view videos is to take advantage of the data redundancy and perform view selection. Li et al. [26] employ a combination of skin-color detector and edge feature-based SVM regression to localize face candidates and estimate their poses. Then, for each of the candidates, a face detector specific to that pose is applied to determine if it is a face. Only the frontal faces are retained for recognition. The algorithm in [27] also relies on an SVM to select frontal faces from video for recognition. The continuity of pose variation in video has inspired the idea of modeling face pose manifolds [28], [29]. The typical method is to cluster the frames of similar pose and train a linear subspace to represent each pose cluster. Here, the piecewise linear subspace model is an approximation to the pose manifold. Wang et al. [30] grow each such linear subspace gradually from a seed sample to include more and more nearest neighbors, until the linearity condition is violated. The linearity is measured as the ratio of geodesic distance to Euclidean distance, and the distances are calculated between a candidate neighbor and each existing sample in the cluster. They define the manifold-manifold distance as the distance between the closest subspace pair from the two manifolds, and the subspace distance is defined as a weighted sum of canonical correlations and exemplar distance. Also assuming that all images of the same person sit on a manifold, Arandjelovic et al. [31] model face videos using Gaussian Mixture Models. The manifold-manifold distance is then measured using the KL divergence between the Gaussian mixtures. Single-view videos have also been modeled using Hidden Markov Models [32], or ARMA models [33]. 3D face models can be estimated from single-view videos as done in [10], [34], [35]. The 3D model can be then used in a model-based algorithm (e.g. [36]) to perform face recognition.

Multi-view-based recognition: In contrast to single-view/video-based face recognition, there are relatively a smaller number of approaches for recognition using multi-view videos. In [37], although both the gallery and the probe are multi-view videos, they are treated just like single-view sequences. Frames of a multi-view sequence are collected together to form a gallery or probe set. The frontal or near-frontal faces are picked by the pose estimator and retained, while others are discarded. The recognition algorithm is frame-based PCA and LDA fused by the sum rule. In [38], a three-layer hierarchical image-set matching technique is presented. The first layer associates frames of the same individual taken by the same camera. The second layer matches the groups obtained in the first layer among different cameras. Finally, the third layer compares the output of the second layer with the training set, which is manually clustered using multi-view videos. Though multi-view data is used to deal with occlusions when more than one subject is present, pose variations are not effectively addressed in this work. Ramnath et al. [39] extend the AAM framework to the multi-view video case. They demonstrate that when 3D constraints are imposed, the resulting 2D+3D AAM is more robust than the single view case. However, recognition was not attempted in this work. Liu and Chen [40] use geometrical models to normalize pose variations. By back-projecting a face image to the surface of an elliptical head model, they obtained a texture map which was then decomposed into local patches. The texture maps generated from different images were compared in a probabilistic fashion. Our work shares some similarities with theirs in the texture mapping stage. This method has been

extended to multi-view videos in [41]. The texture mapping procedure was further elaborated by adding a geometric deviation model to describe the mapping error. However, tracking, texture mapping and recognition steps were all carried out for each view independently.

As mentioned earlier, almost all of the above referenced algorithms incorporate a pose estimation or model registration step, or even assume that pose is known a priori. The problem naturally arises when we try to compare face appearances described by pose-sensitive features.

Video processing in multi-camera networks: Camera networks have been extensively used for surveillance and security applications [42]. Research in this field has been focused on distributed tracking, resource allocation, activity recognition and active sensing. Yoder et al. [43] track multiple faces in a wireless camera network. The observations of multiple cameras are integrated using a minimum variance estimator and tracked using a Kalman filter. Song and Roy-Chowdhury present a multi-objective optimization framework for tracking in a camera network in [44]. They adapt the feature correspondence computations by modeling the long-term dependencies between them and then obtain statistically optimal paths for each subject. Song et al. [45] incorporate the concept of consensus into distributed camera networks for tracking and activity recognition. The estimate made by each camera is shared with its local neighborhood, and the consensus algorithms combine the decisions from single cameras to make a network-level decision. A detailed survey on video processing in camera networks can be found in [46].

Spherical harmonics (SH) in machine vision: Basri and Jacobs [47] use SH to model Lambertian objects under varying illumination. Specifically, they proved that the reflectance function produced by convex, Lambertian objects under distant, isotropic lighting can be well approximated using the first nine SH basis functions. Ramamoorthi [48] revealed the connection between SH and PCA, showing that the principal components are equal to the SH basis functions under appropriate assumptions. Zhang and Samaras [49] proposed an algorithm to estimate the SH basis images for a face at a fixed pose from a single 2D image based on statistical learning. When the 3D shape of the face is available, the SH basis images can be estimated for test images with different poses. Yue et al. [50] adopted a similar strategy where the distribution of SH basis images is modeled as Gaussian and its parameters are learned from a 3D face database. Note that all these works are based on Lambertian reflectance model. As a result, they require a 3D face model and face pose estimation to infer the face appearance. In contrast, we use an SH-based feature to directly model face appearance rather than the reflectance function, and hence do not require a 3D face surface model or a pose estimation step.

III. Robust Feature

The robust feature presented here is based on the theory of spherical harmonics. Spherical harmonics are a set of orthonormal basis functions defined over the unit sphere, and can be used to linearly expand any square-integrable function on $\mathbb{S}^2$ as:

$$f(\theta,\phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} f_{lm} Y_{lm}(\theta,\phi), \tag{1}$$

where $Y_{lm}(\cdot,\cdot)$ defines the SH basis function of degree $l \geq 0$ and order $m \in (-l, -l+1, \ldots, l-1, l)$. f_{lm} is the coefficient associated with the basis function Y_{lm} for the function f. Note that we are using the spherical coordinate system. $\theta \in (0, \pi)$ and $\phi \in (0, 2\pi)$ are the zenith and azimuth angles, respectively. There are $2l+1$ basis functions for a given order l [51].

The SH basis function for degree l and order m has the following form:

$$Y_{lm}(\theta,\phi) = K_{lm} P_l^m(\cos\theta) e^{im\phi} \tag{2}$$

where K_{lm} denotes a normalization constant such that:

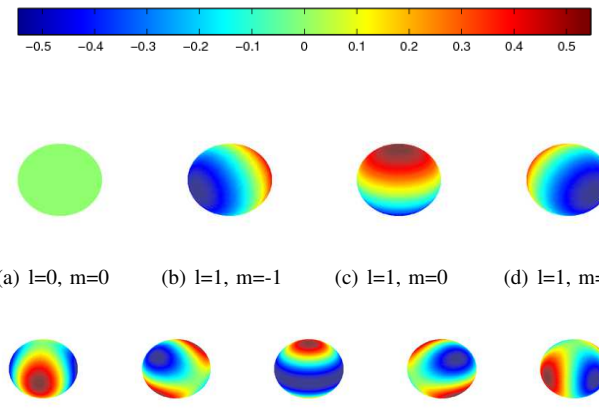
$$\int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} Y_{lm} Y_{lm}^* d\phi d\theta = 1 \tag{3}$$

Here, $P_l^m(x)$ are the associated Legendre functions.

In this paper, we are interested in modeling real-valued functions (eg. texture maps) and thus, we are more interested in the real Spherical Harmonics which are defined as

$$Y_l^m(\theta,\phi) = \begin{cases} Y_{l0} & \text{if } m = 0 \\ \frac{1}{\sqrt{2}}(Y_{lm} + (-1)^m Y_{l,-m}) & \text{if } m > 0 \\ \frac{1}{\sqrt{2}i}(Y_{l,-m} - (-1)^m Y_{lm}) & \text{if } m < 0 \end{cases} \tag{4}$$

The real SHs are also orthonormal and they share most of the important properties of the general Spherical Harmonics. For the rest of the paper, we will use the word "spherical harmonics" to refer exclusively to real SHs. We visualize the SH for degree l = 0, 1, 2 in Fig. 1.



(a) l=0, m=0 (b) l=1, m=-1 (c) l=1, m=0 (d) l=1, m=1

(e) l=2, m=-2 (f) l=2, m=-1 (g) l=2, m=0 (h) l=2, m=1 (i) l=2, m=2

Fig. 1. Visualization of the first three degree of Spherical Harmonics.

As with Fourier expansion, the SH expansion coefficients f_l^m can be computed as:

$$f_l^m = \int_{\theta} \int_{\phi} f(\theta,\phi) Y_l^m(\theta,\phi) d\theta d\phi \tag{5}$$

The expansion coefficients have a very important property which is directly related to our "pose free" face recognition application.

Proposition: If two functions $f(\theta,\phi)$ and $g(\theta,\phi)$, defined on $\mathbb{S}^2$, are related by a rotation $R \in SO(3)$, i.e. $g(\theta,\phi) = f(R(\theta,\phi))$, and their SH expansion coefficients are f_l^m and g_l^m, respectively, the following relationship exists:

$$g_l^m = \sum_{m'=-l}^{l} D_{mm'}^l f_l^{m'} \tag{6}$$

and the $D_{mm'}^l$s satisfy:

$$\sum_{m'=-l}^{l} (D_{mm'}^l)^2 = 1 \tag{7}$$

In other words, (6) suggests that after rotation, the SH expansion coefficients at a certain degree l are linear combinations of those before the rotation, and coefficients at different degrees do not affect each other. This can also be represented in a matrix form:

$$\begin{pmatrix} f_0^0 \\ f_1^{-1} \\ f_1^0 \\ f_1^1 \\ f_2^{-2} \\ \vdots \\ \vdots \\ f_2^2 \\ \vdots \end{pmatrix} = \left[\begin{array}{c|ccc|ccccc|c} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & . \\ \hline 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ 0 & x & x & x & 0 & 0 & 0 & 0 & 0 & . \\ \hline 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ 0 & 0 & 0 & 0 & x & x & x & x & x & . \\ \hline . & . & . & . & . & . & . & . & . & . \end{array}\right] \begin{pmatrix} g_0^0 \\ g_1^{-1} \\ g_1^0 \\ g_1^1 \\ g_2^{-2} \\ \vdots \\ \vdots \\ g_2^2 \\ \vdots \end{pmatrix}, \tag{8}$$

where the x denotes non-zero entries corresponding to appropriate $D_{mm'}^l$ values.

This proposition is a direct result of the following Lemma [51] [52].

Lemma: Denote by E_l the subspace spanned by $Y_l^m(\theta,\phi)$, $m = \{-l,\ldots,l\}$, then E_l is an irreducible representation for the rotation group SO(3).

(A complete proof of the proposition can be found in the appendix.)

We further look into a energy vector associated with a $f(\theta,\phi)$ defined on $\mathbb{S}^2$ as:

$$e_f = (\|\mathbf{f}_0\|_2, \|\mathbf{f}_1\|_2, \|\mathbf{f}_l\|_2, \ldots), \tag{9}$$

where $\|\cdot\|_2$ denotes the ℓ_2-norm, and $\mathbf{f}_l$ consists of all the SH decomposition coefficients of $f(\theta,\phi)$ at degree l:

$$\mathbf{f}_l = \{f_l^m, m = -l,\ldots,l\}. \tag{10}$$

Equation (7) guarantees that e_f is invariant when $f(\theta,\phi)$ is rotated. In practice, we find that subsequent normalization of e_f with respect to total energy increases reliability. This results in a feature which describes the spectrum of the SH coefficients. We refer to it as the SH spectrum feature.

The specific form of the function $f(\theta,\phi)$ varies with applications and is often numerically defined for sampled points on the surface of a sphere. In our multi-view face recognition scenario, $f(\theta,\phi)$ is the face appearance as represented by a texture map/template. To be more specific, we use a sphere to approximate the human head and the relevant image regions in multi-view data are mapped onto the surface of the sphere according to projective geometry. This procedure will be described in detail in Section IV. Note that the spherical model is different from the 3D face model in a general sense as one does not have to estimate the surface normals. Using a simple spherical model is often sufficient when we deal with low-resolution images and hence, is suitable for camera networks. Constructing a reasonable 3D face model usually requires much higher image resolution and computations. More importantly, this model enables us to set up a connection between multi-view face image and SH representation. Indeed, even when the face undergoes extreme pose variations, the SH spectrum feature extracted from the texture maps remains stable, leading to pose-robust face recognition. Note that the normalization step in feature extraction is equivalent to assuming that all the texture maps have the same total energy, and in a loose sense functions as an illumination normalization step. Although this means that skin color information is not used for recognition, experimental results are good. Fig. 2 shows an example. One can see that features extracted from the same subject's texture map are very close even when large pose variations are present, and they are much closer than those extracted from different subjects but under the same pose.

Another advantage of the SH spectrum feature is its ease of use. There is only one parameter to be determined, namely the number of degrees in the SH expansion. Apparently, a trade-off exists for different choices of parameter values: A higher degree number means better approximation, but it also comes with a price of more expensive computational cost. In Fig. 3, we visualize a 3D head texture map as a function defined on $\mathbb{S}^2$, and its reconstruction resulting from 20, 30 and 40 degree SH transform respectively. The ratio of computation time for the 3 cases is roughly 1:5:21. (On a PC with Xeon 2.13GHz CPU, it takes roughly 1.2 seconds to do a 20 degree SH transform for 18050 points.) We have empirically observed that the 30-degree transform usually achieves a reasonable balance between approximation error and computational cost.

IV. Multi-Camera Tracking and Texture Mapping

In this section, we describe a robust multi-view tracking algorithm based on Sequential Importance Resampling (SIR) (particle filtering) [53]. Tracking is an essential stage in camera-network-based video processing. It automates the localization of the face and has direct impact on the performance of the recognition algorithm. Recall that the proposed SH spectrum feature is extracted from the texture map of the face under a spherical head model. The tracking module, together with a texture mapping step, describes the entire feature extraction process (see Fig. 4).

A. Multi-View Tracking

To fully describe the position and pose of a rigid 3D object, we usually need a 6-D representation ($\mathbb{R}^3 \times SO(3)$), where the

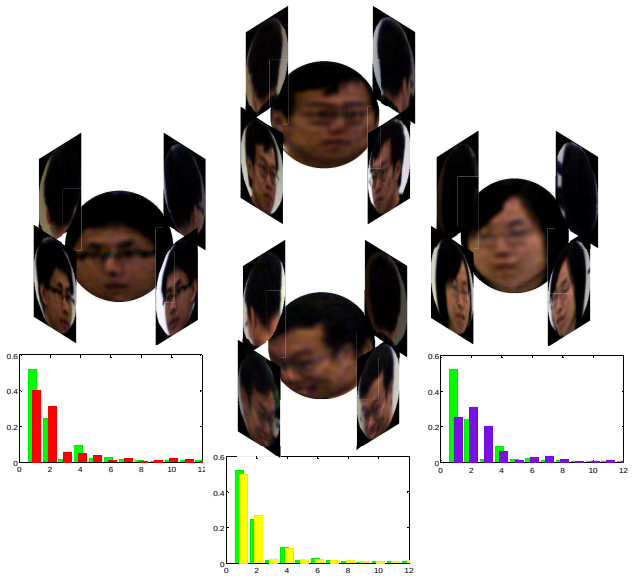


Fig. 2. **Robust features based on Spherical Harmonics.** The texture of each model is constructed from multi-view images captured by four synchronized cameras. The top and bottom models correspond to the same subject, but the capture time of the two sets of images are separated by a time span of more than 6 months. Note that we intentionally rotate the bottom model by 180° so that readers can see that it is the same subject as in the top one. Therefore their actual pose difference is even larger than the one shown. The green bars in the three bar graphs are the same feature vector extracted from the top model. For visualization considerations, only the first 12 elements of the feature vector are plotted here.

Fig. 3. **Comparison of the Reconstruction Qualities with SH Coefficients** The images from left to right are: the original 3D head texture map, the texture map reconstructed from 40-degree, 30-degree and 20-degree SH coefficients, respectively. Note that we interpolated the surface points for a better visualization quality.

3-D real vector space is used to represent the object's location, and $SO(3)$ is used to represent the object's rotation. It is well known that higher the dimensionality of the state space is, the harder the tracking problem becomes. This is especially true for search-algorithms like SIR since the number of particles typically grows dramatically for high-dimensional state spaces.

However, given that our eventual recognition framework is built on the robust feature derived using SH representation under the diffuse lighting assumption, it suffices that we track only the location of the head in 3D. Hence, the state space for tracking $\mathbf{s} = (x, y, z)$ represents only the position of a sphere's center, disregarding any orientation information. Initialization of the tracker can be solved through face detection (For example, the cascaded Haar-feature detector in [54]) applied to the first frame and followed by multi-view triangulation.

The state transition model $P(\mathbf{s}_t|\mathbf{s}_{t-1})$ is modeled as a Gaussian distribution $\mathcal{N}(\mathbf{s}_t|\mathbf{s}_{t-1}, \sigma^2\mathbf{I})$. We found that the tracking result is relatively insensitive to the specific value of σ and have fixed it in all of our experiments.

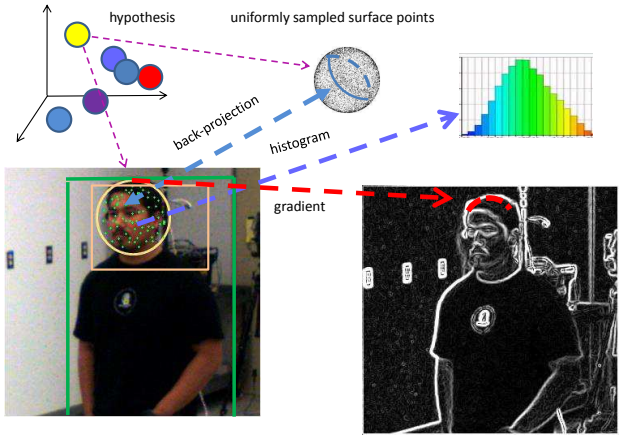


Fig. 4. **The Multi-Cue Tracking Algorithm and Back-Projection.** The yellow circle is the boundary of the head's image for a certain hypothesis state vector. The green and orange rectangles mark the human body detection result and the estimated range of head center's projection, respectively. Green dots are the projections of model's surface points. The navy-blue curve on the sphere highlights the boundary of the visible hemisphere. Note that we draw tracking and back-projection together just for illustration. In actual case, only the MAP estimate of the state vector will be back-projected to construct the texture map.

The observation model $P(O_t|\mathbf{s}_t)$ of the tracker is based on multiple cues such as a histogram, the gradient map and a geometric constraint.

Histogram: To evaluate the image likelihood for a hypothesized state vector $\mathbf{s}_t^i$, we assume a weak-perspective camera model and calculate the image of the spherical model on the jth camera's image plane, which is a disk-like region E_j^i (We shall use the subscript j to indicate the jth view). A normalized 3D histogram in RGB space is built from this image region. Its difference with the template, which is set up at the first frame through the same procedure and subject to adaptive update thereafter, is measured by the Bhattacharyya distance. This defines the first cue matching function $\phi(O_t, \mathbf{s}_t^i)$.

Gradient map: On the circular perimeter of the model's image, we select the 90° arc segment on the top, superimposing it on the horizontal and vertical gradient map of $I_{t,j}$. Despite various shapes of human heads, this part of the boundary turns out to reliably coincide with an arc. Therefore, if the state vector is a good match to the ground truth, we expect the magnitude of the image gradient response along this arc segment to be strong and its direction to be perpendicular to the tangent directions [55]. Consequently, we formulate the second cue matching score as:

$$\varphi(O_t, \mathbf{s}_t^i) = \frac{1}{r_j^i} \sum_{m=1}^{M} |\mathbf{n}_m \cdot \bigtriangledown \mathbf{I}_m|, \tag{11}$$

where r_j^i is the radius of E_j^i measured in number of pixels, $\mathbf{n}_m$ is the normal vector of the m-th pixel on the arc, and $\bigtriangledown \mathbf{I}_m$

is the image gradient at this pixel.

Geometric constraint: We impose geometric constraints to the state vector by applying the part-based human body detector as proposed in [56]. The detector is based on the histogram of gradients (HOG) feature. We further apply body size constraints to filter out potential background human subjects, and then pick the detection result with highest confidence value among the remaining ones. A reliable head region R_j^i with respect to the detected human body area is then selected. Note this cue forms a hard constraint for the state vector:

$$\psi(O_t, \mathbf{s}_t^i) = \begin{cases} 0 & \text{if } E_j^i \subset R_j^i = \emptyset \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

The overall image likelihood can be calculated as:

$$P(O_t|\mathbf{s}_t^i) \propto \ln \psi(O_t, \mathbf{s}_t^i) + \lambda_1 \ln \phi(O_t, \mathbf{s}_t^i) + \lambda_2 \ln \varphi(O_t, \mathbf{s}_t^i), \tag{13}$$

where λ_1 and λ_2 are determined by applying a logistic regression-like algorithm to independent data. We determine the location of the head in 3D space as:

$$\begin{aligned} \mathbf{s}_t &= \underset{\mathbf{s}_t^i}{\text{argmax}} P(\mathbf{s}_t^i|O_t) \\ &= \underset{\mathbf{s}_t^i}{\text{argmax}} P(O_t|\mathbf{s}_t^i) P(\mathbf{s}_t^i|\mathbf{s}_{t-1}^i) \end{aligned} \tag{14}$$

Fig. 5 shows the result of our multi-view tracking algorithm. The tracker is able to track all the 500 frames without failure. Note that the video contains significant head motions in terms of rotation, translation and scaling. It is also subject to interruptions when the head moves out of the field of view. The second video example shown in Fig. 6 was captured when Baratunde Thurston, a technology-loving humorist and host of the Science Channel, visited the Biomotion laboratory at University of Maryland. Our multi-view tracking algorithm accurately locates the subject's head in spite of his dramatic motion. (Both videos are provided as supplementary materials.) Though in real-world surveillance videos subjects usually do not perform such extreme motions as in the example videos, the results clearly illustrate the robustness of our algorithm. The tracker also successfully handles all the videos in our database.

B. Texture Mapping

Once the MAP estimate of the head center is obtained, we are ready to obtain the surface texture map for the model. First, we uniformly sample the sphere's surface according to the following procedure:

1) Uniformly sample within the range $[-R, R]$, where R is the radius of the sphere, to get z_n, $n = 1, 2, ..., N$.
2) Uniformly sample α_n within the range $[0, 2\pi]$, and independent of z_n.
3) $x_n = \sqrt{R^2 - z_n^2} \cos \alpha_n$, $y_n = \sqrt{R^2 - z_n^2} \sin \alpha_n$.

Then, we perform a coordinate transformation for these sample points. Assume that their original world coordinates are $\{(x_n, y_n, z_n), n = 1, 2, ..., N\}$. After the transformation, we obtain $\{(x'_{n,j}, y'_{n,j}, z'_{n,j})\}$, which are their coordinates in the jth camera coordinate reference frame. We determine their visibility to camera j by examining $(x'_{n,j}, y'_{n,j}, z'_{n,j})$. Only an un-occluded point, i.e. which satisfies $z'_{n,j} \leq z'_{0,j}$, can contribute to an image on the jth camera's image plane. Here, $z'_{0,j}$ is the distance from the head center to the jth camera center. It is said that a back-projection link is created between a sample point on the model's surface and a pixel in a frame I_j if the former's world coordinates (x_n, y_n, z_n) and the latter's image coordinates $(x''_{n,j}, y''_{n,j})$ can be related under the weak-perspective projection assumption.

We denote the texture map for the jth camera view obtained by using such a back-projection approach as T^j. Note that when we iterate the procedure over all the cameras in the network, some model points will correspond to pixels from multiple views, because these cameras have overlapped field of views. For sample points in the overlapped region, we adopted a weighted fusion strategy, i.e., we assign weight $w_{n,j}$ to a pixel with image coordinate $\mathbf{p}_{n,j}$:

$$w_{n,j} = \exp(-\|\mathbf{p}_{n,j} - \mathbf{p}_{0,j}\|/r_j^i), \tag{15}$$

where $\mathbf{p}_{0,j}$ is the image coordinates of the pixel back-projected by the head model, and thus roughly the center of all the projections for camera j. Intuitively, the closer a pixel is to this center, the larger its contribution to the texture map should be. On the rim of a sphere a large number of sample points tend to project to the same pixel, and hence are not suitable for back-projection. The texture of the model point with world coordinates (x_n, y_n, z_n) is determined by:

$$T(x_n, y_n, z_n) = T^{j_0}(x_n, y_n, z_n), \tag{16}$$

where

$$j_0 = \arg\max_j \ w_{n,j}, \ j = 1, 2, ..., K. \tag{17}$$

This weighting scheme is illustrated in Fig. 7. Note that in our multi-view face recognition algorithm, T is in fact the function $f(\theta, \phi)$ that is subject to decomposition, as described in Section III.

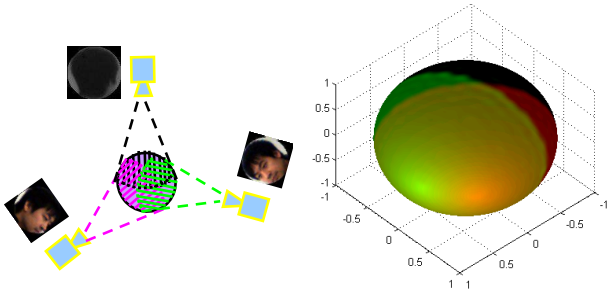


Fig. 7. **Weighted Texture Mapping.** In multi-view texture mapping, the field of views of different cameras in a network often have overlap. The red (green) region on the sphere model represents the targeting back-projection area for the first (second) camera. The redness (greenness) at a certain point is proportional to its texture mapping weight with regard to the first (second) camera. In their overlapping region, whether a point is more red or more green determines which camera's image the texture map at that point should be based on.

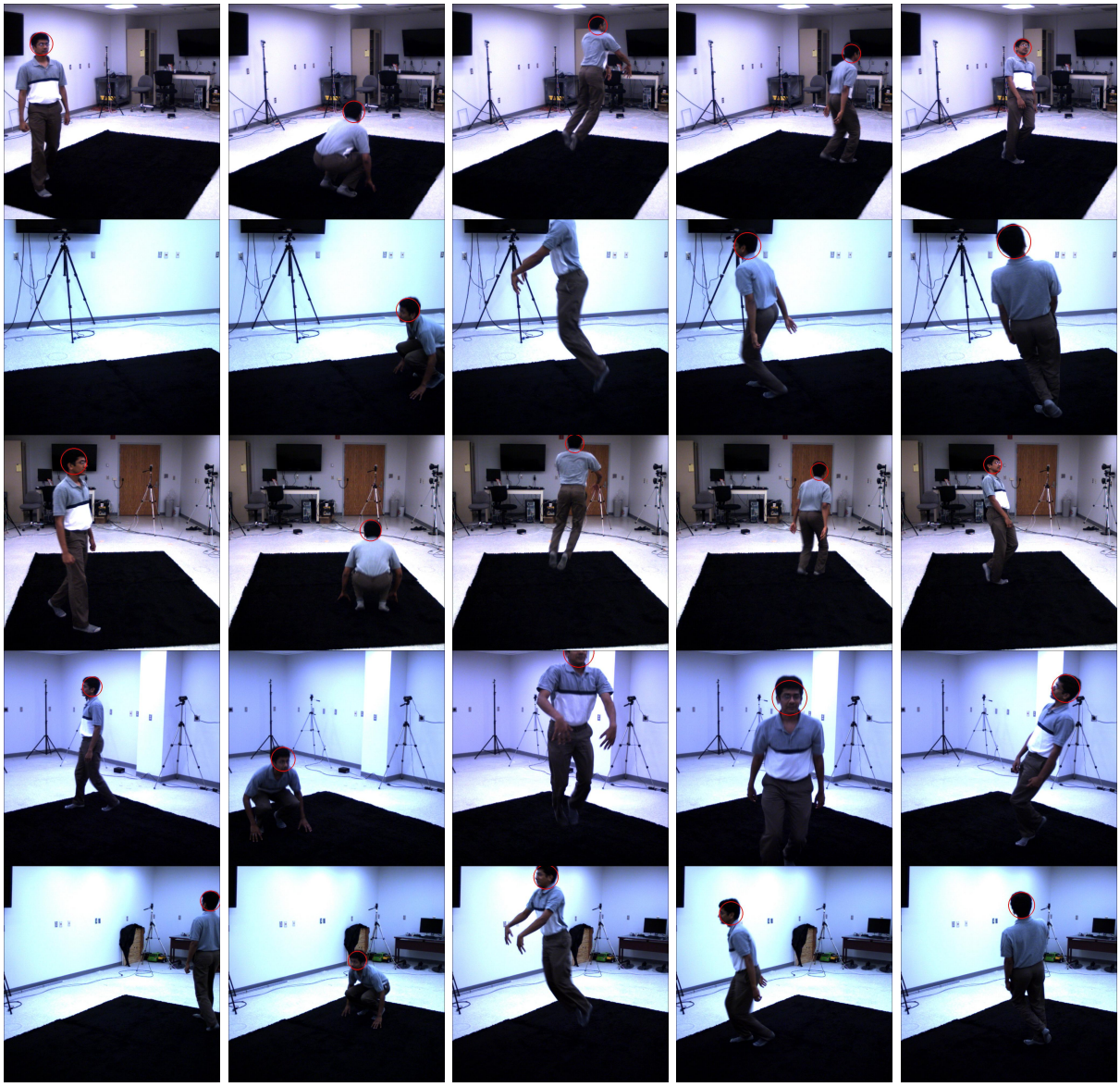


Fig. 5. **Sample Tracking Results** Tracking results for a 500-frame multi-view video sequence. 5 views are shown here. Each row of images is captured by the same camera. Each column of images is captured at the same time.

V. Video-Based Recognition

Video-based face recognition has some advantages. First, video offers data redundancy, which can be exploited to improve the robustness of a recognition algorithm. It has been reported in the literature that video-based algorithms in general achieve better performance than image-based ones. Second, by performing video tracking we can automate feature acquisition. Although it is always possible to extend the frame-based recognition result to a video-based one via simple fusion rules such as majority voting, a principled approach that exploits data's underlying structure is often more desirable for performance reason.

Given two multi-view video sequences with m and n (note that in general $m \neq n$) multi-view frames (a multi-view frame refers to the group of K frames synchronously captured by K cameras), respectively, two sets of feature vectors can be extracted. We look into their projections in the reproducing kernel Hilbert space (RKHS). The projection is indirectly performed via an Radial Basis Function (RBF) kernel. It is known that the kernel trick induces nonlinear feature mapping, which often leads to easier separation in RKHS. We treat each instance of feature vector as a sample from its class-conditional probability distribution. Therefore, the similarity of the two ensembles of features can be measured using the distance between the two class-conditional probability distributions in RKHS. By assuming that these distributions are Gaussian, analytical form of several different distance measures are derived in [57]. We follow [57] to calculate the limiting Bhattacharyya distance. To this end, the rank-deficient covariance matrix (since the dimensionality of RKHS is much higher than the number of data samples) involved in calculating the Bhattacharyya distance is replaced by an invertible approximation $\mathbf{C}$, which preserves the dominant eigenvalues and eigenvectors. The limiting Bhattacharyya distance in this

Fig. 6. **Sample Tracking Results** Tracking results for a 200-frame multi-view video sequence. The subject performs dramatic dancing motions. Five views are shown here. Each row of images is captured by the same camera. Each column of images is captured at the same time.

case is:

$$D = \frac{1}{8}(\alpha_{11} + \alpha_{22} - 2\alpha_{12}), \tag{18}$$

where

$$\alpha_{ij} = \mu_i^T \left(\frac{1}{2}\mathbf{C}_i + \frac{1}{2}\mathbf{C}_j\right)^{-1} \mu_j^T. \tag{19}$$

We now show the steps to calculate (19) from the Gram matrix. Denote the Gram matrix as $\mathbf{K}_{ij}$, where $i, j \in \{1, 2\}$ are the indices of ensembles. The $\mathbf{K}_{11}$ and $\mathbf{K}_{22}$ are then centered:

$$\mathbf{K}'_{ii} = \mathbf{J}_i^T \mathbf{K}_{ii} \mathbf{J}_i, \ \mathbf{J}_i = N_i^{-1/2}(I_N - \mathbf{s}\mathbf{1}^T) \tag{20}$$

where $\mathbf{s} = N_i^{-1}\mathbf{1}$, $\mathbf{1}$ is a $N_i \times 1$ vector of 1s and N_i is the number of vectors in ensemble i. Let $\mathbf{V}_i$ be the matrix which stores the first r eigenvectors of $\mathbf{K}'_{ii}$ (i.e. corresponding to the r largest eigenvalues). Define:

$$\mathbf{P} = \begin{pmatrix} \sqrt{\frac{1}{2}}\mathbf{J}_1\mathbf{V}_1 & 0 \\ 0 & \sqrt{\frac{1}{2}}\mathbf{J}_2\mathbf{V}_2 \end{pmatrix}, \tag{21}$$

then it can be verified that

$$\left(\frac{1}{2}\mathbf{C}_i + \frac{1}{2}\mathbf{C}_j\right)^{-1} = \mathbf{I}_f - \begin{pmatrix} \mathbf{\Phi}_1 & \mathbf{\Phi}_2 \end{pmatrix} \mathbf{B} \begin{pmatrix} \mathbf{\Phi}_1^T \\ \mathbf{\Phi}_2^T \end{pmatrix}. \tag{22}$$

$\mathbf{I}_f$ is $f \times f$ identity matrix, where f is the dimensionality of the RKHS. And $\mathbf{\Phi}$ is the matrix of nonlinearly-mapped data in RKHS, which is not explicitly available to us. Matrix $\mathbf{B}$ can be computed from the Gram matrix:

$$\mathbf{B} = \mathbf{P}\mathbf{L}^{-1}\mathbf{P}^T, \ \mathbf{L} = \mathbf{P}^T \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix} \mathbf{P}. \tag{23}$$

By combining (19) and (22), we have:

$$\alpha_{ij} = \mathbf{s}_i^T \mathbf{K}_{ij} \mathbf{s}_j - \mathbf{s}_i^T \begin{pmatrix} \mathbf{K}_{i1} & \mathbf{K}_{i2} \end{pmatrix} \mathbf{B} \begin{pmatrix} \mathbf{K}_{1j} \\ \mathbf{K}_{2j} \end{pmatrix} \mathbf{s}_j. \tag{24}$$

VI. Experiments

A. Database

As mentioned in Section II, there are very few works addressing the multi-view face recognition problem. We exhaustively searched for a public multi-view video-based face database. It seems that a database which contains videos captured by multiple **synchronized** cameras is not available yet. Therefore, we collected a multi-view video database using an indoor camera network. The database has 40 subjects. The videos were collected at four different sessions and are 100 to 200 frames in length. Most of the subjects have 3 videos and some of them have 2 or 4 videos. We use one as gallery and the rest as probes. This database is double the size of its previous

versions [58] [59] in terms of the number of videos. To test the robustness of our recognition system, we have arranged the time span that separated the sessions to be up to 6 months. The appearance of many subjects has changed significantly between the sessions. Such a dataset well serves our purpose of simulating a practical surveillance environment and poses great challenges to multi-view face recognition algorithms. Fig. 8 shows some example frames from gallery and probe video sequences.

Fig. 8. **Example of Gallery and Probe Video Frames.** Shown in the first row are examples of gallery frames and the second row are examples of probe frames.

B. Feature Comparison

As the proposed feature can work for a single multi-view frame as well as video sequences, we first associate four different kinds of features with different classifiers to compare their performance in image-based face recognition settings. By "image-based face recognition" we mean that each frame (a multi-view frame for the SH spectrum feature and a single-view frame for other features.) is treated as a gallery or probe individually without concerning which video it comes from. We use one multi-view video of each subject as the gallery and the remaining videos as probe. We pick every 10th frame in this experiment. The four features are: Locality Preserving Projection (LPP) and LDA in the original image space, SH raw coefficients with PCA, and the proposed SH spectrum feature. For the first two features, we use the face image that is automatically cropped by a circular mask as a result of tracking, and normalize it to the size 50×50. For LDA, we first train a PCA projection matrix from all the gallery images to reduce the dimension of the original image feature, in order to avoid the intra-class scatter matrix's rank deficiency issue. As in the conventional LDA formulation, the criterion we choose to optimize is $\det(WS_bW^T)/\det(WS_wW^T)$, where W is the projection matrix, and S_b and S_w are the between-class/within-class scatter matrices, respectively. For LPP, we utilize label information in the gallery by setting the weights between inter-class samples to be 0. We also use cross-validation to determine the optimal scale constant which is defined in the weight matrix of LPP. The experiment runs in a single-view vs. single-view mode for the LPP and LDA case, and in a multi-view vs. multi-view mode for the SH+PCA and SH spectrum feature case. The results are shown in Table I. Due to the incompatibility of the nature of single-view features with the special structure of multi-view image data, the performance of the proposed feature exceeds them by a large margin in all cases.

TABLE I
COMPARISON OF RECOGNITION PERFORMANCE

Feature	NN	KDE	SVM-Linear	SVM-RBF
LPP	56.1%	42.7%	58.8%	65.9%
LDA	51.3%	34.8%	40.6%	47.4%
SH PCA	40.7%	36.4%	39.3%	52.2%
Proposed	**65.3%**	**65.1%**	**79.0%**	**87.3%**

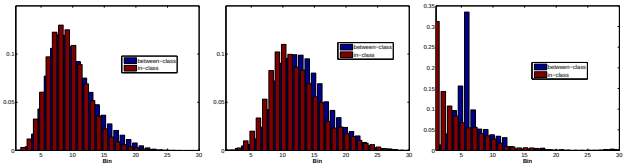


Fig. 9. **Comparison of the Discriminant Power** Histograms of between-class distance distribution (blue) and in-class distance distribution (red) of the LDA feature (left), LPP feature (center) and the SH spectrum feature (right) are presented above. Number of bins is 30.

To quantitatively verify the proposed feature's discriminant power, we then conducted the following experiment. We calculate distances for each unordered pair of feature vectors $\{\mathbf{x}_i, \mathbf{x}_j\}$ in the gallery. If $\{\mathbf{x}_i, \mathbf{x}_j\}$ belongs to the same subject, then the distance is categorized as being *in-class*. Otherwise, the distance is categorized as being *between-class*. We approximate the distribution of the two kinds of distances as histograms. Intuitively, if a feature has good discrimination power, then the in-class distances evaluated using that feature tends to be smaller compared to the between-class distances, and hence the distributions of the two distances should exhibit large divergence. We use the symmetric KL divergence $(KL(p\|q) + KL(q\|p))$ to evaluate the difference between the two distributions. We summarize the results for the four features in Table II and plot three of them in Fig. 9. The in-class distances for the SH spectrum feature are concentrated in the low value bins, while its between-class distance tends to have higher values. Their modes are obviously separated. For the other features, the between-class distance tend to mix with the in-class distance. The symmetric KL-divergence also suggests the same phenomenon.

C. Video-Based Recognition

The algorithm we use for video-level recognition is the one as described in Section V. We compare the performance of our video recognition algorithm with five other ones: (1) Ensemble-similarity-based algorithm directly applied to the raw image. Inputs are the head images which are tracked in a video and scaled to size 50 by 50. The kernel is RBF. (2) View-selection-based algorithm. We use a Viola-Jones frontal face detector [54] to select frontal-view face images from both gallery and probe multi-view videos. The chosen frames from a subject's gallery video are then used to construct the personal

TABLE II
KL DIVERGENCE OF IN-CLASS AND BETWEEN-CLASS DISTANCES FOR DIFFERENT FEATURES

LPP	LDA	SH+PCA	SH Spectrum
0.3511	0.2709	0.2866	**1.3141**

frontal-view face PCA subspace. The frontal-view frames from the probe videos are fitted to the personal PCA subspaces for recognition. Video-level decision is made through majority voting. (3) The probabilistic appearance manifold algorithm proposed in [28]. We use 8 planes for the local manifold model and set the probability of remaining the same pose to be 0.7 in the pose transition probability matrix. We first use this algorithm to process each camera view of a probe video. To fuse results of different camera views we use majority voting. If there is a tie in views' voting, we pick the one with smaller Hausdorff distance. (4) Image-based recognition with SH spectrum feature and majority voting for video-level fusion. We use SVM with RBF kernel for every multi-view frame recognition. Note however that the recognition accuracies in this case should not be compared to the previous experiment's result to draw misleading conclusions[2]. (5) The Manifold-Manifold Distance (MMD) algorithm presented in [30]. We use the author's code and parameter settings. When comparing two multi-view videos, we first calculate the MMD between the sequence pairs of the same view, and then use the minimum MMD across views as the distance measure. We also tried with average MMD across views, which yielded similar results.

We plot the cumulative recognition rate curve in Fig. 10. The view-selection method heavily relies on the availability of frontal-view face images, however, in the camera network case, the frontal pose may not appear in any view of the cameras. As a result, it does not perform well in this experiment. The manifold-based algorithm, the MMD-based algorithm and the image-ensemble-based algorithm use more principled strategies than voting to combine classification results of individual frames. Moreover, they both have certain ability to handle pose variations, especially the two algorithms based on manifold. However, because they are designed to work with a single camera, they are single-view in nature. Repeating these algorithms for each view does not fully utilize the multi-view information. For example, we found in our experiments that mismatches made by the MMD algorithm often happens when the minimum MMD is produced between the back-of-head clusters, which have similar appearance representations even for different subjects. In contrast, the proposed method based on a robust feature performs noticeably better in this experiment. An additional advantage of the algorithm is that it requires no pose estimation or model registration. Comparison between the ensemble matching algorithm and the majority voting method which both use the proposed feature demonstrates the superiority of a systematic fusion strategy to an ad-hoc one.

[2]The numbers in the two cases are not convertible to each other, as in the previous image-based recognition experiment we did not fuse results with respect to video. Think of two extreme situations: (1) For each video of the probe set, 51% frames are individually correctly recognized. (2) For half of the probe videos, 100% frames are individually correctly recognized and for the remaining half only 49% frames are correctly recognized. The overall image recognition rate and majority-voting-based video recognition rate are respectively 51% and 100% in the former case, and 74.5% and 50% in the latter one.

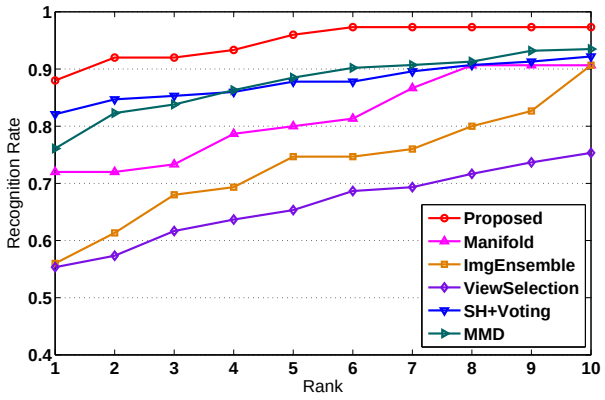


Fig. 10. **Video Face Recognition Results** Cumulative recognition rate of the video-based face recognition algorithms.

VII. CONCLUSION

In this paper, we proposed a multi-view face recognition algorithm. The most noteworthy feature of the algorithm is that it does not require any pose estimation or model registration step. Under the normal diffuse lighting condition, we present a robust feature by exploring the fact that the subspace spanned by Spherical Harmonics is an irreducible representations for the SO(3) group. We also proposed a multi-view video tracking algorithm to automate the feature acquisition in a camera network setting. We modeled the video-based recognition problem as one of measuring ensemble similarities in RKHS. We demonstrated the performance of our method on a relatively uncontrolled multi-view video database.

Limitations One limitation of our method is that the pose insensitivity property of the SH representation relies on the assumption that the spherical function remains unchanged other than a rotation, i.e.: $f(\theta, \phi, t_1) = f(R(\theta, \phi), t_2)$. In practice, this could always be affected by real-world lighting conditions. Under normal lighting conditions, this assumption is reasonable, and as we mentioned, even global illumination variation can be partially compensated for by the energy normalization step in feature extraction. However, extreme lighting conditions can render the assumption invalid. This could happen when, for example, there are anisotropic illumination variations, or strong directional light is casted onto the face from the side. Such situations will result in large fluctuation in the features and cause the recognition performance to degrade. There are some possible solutions to this problem. For example, we could use the self-quotient method to preprocess video frames, or we could figure out a way to integrate the algorithm in [47] for a uniform modeling of both lighting conditions and face appearance. This will be one of our future research directions. Our algorithm also relies on the assumption that human head can be approximated by a sphere. While this approximation may be reasonable, model fitting errors due to the non-spherical nature of human heads

do exist and can become evident in certain cases. Moreover, because we treat the texture map as a spherical function, unavoidably there will be quantization error caused by the discrete pixel value. Finally, calibration of camera network could be a source of error, too. We also plan to extend our work to more complicated conditions, such as outdoor environments, less stringent calibration requirements etc.

APPENDIX
PROOF OF THE PROPOSITION

Proposition If two functions defined on $\mathbb{S}^2$: $f(\theta,\phi)$ and $g(\theta,\phi)$ are related by a rotation $R \in SO(3)$, i.e. $g(\theta,\phi) = R(f(\theta,\phi))$, and their SH expansion coefficients are f_l^m and g_l^m ($l = 0, 1, ...$ and $m = -l, ..l$), respectively, the following relationship exists:

$$g_l^m = \sum_{m'=-l}^{l} D_{mm'}^l f_l^{m'} \tag{25}$$

and the $D_{mm'}^l$s satisfy:

$$\sum_{m'=-l}^{l} (D_{mm'}^l)^2 = 1 \tag{26}$$

Proof Let us denote the lth degree frequency component as $f_l(\theta,\phi)$:

$$f_l(\theta,\phi) = \sum_{m=-l}^{l} f_l^m Y_l^m(\theta,\phi) \tag{27}$$

, then $f_l(\theta,\phi) \in E_l$. According to the Lemma:

$$\begin{aligned} g_l(\theta,\phi) &= R(f_l(\theta,\phi)) \\ &= R(\sum_{m=-l}^{l} f_l^m Y_l^m(\theta,\phi)) \\ &= \sum_{m=-l}^{l} f_l^m R(Y_l^m(\theta,\phi)) \\ &= \sum_{m=-l}^{l} f_l^m \sum_{m'=-l}^{l} D_{mm'}^l Y_l^{m'}(\theta,\phi) \\ &= \sum_{m'=-l}^{l} \sum_{m=-l}^{l} f_l^m D_{mm'}^l Y_l^{m'}(\theta,\phi) \end{aligned} \tag{28}$$

Equation (25) follows by comparing (28) with

$$g_l(\theta,\phi) = \sum_{m'=-l}^{l} g_l^{m'} Y_l^{m'}(\theta,\phi) \tag{29}$$

As for Equation (26), notice that Y_l^ms and $Y_l^{m'}$ are both orthonormal basis:

$$\begin{aligned} RHS &= 1 \\ &= \int_{\theta=0}^{\pi}\int_{\phi=0}^{2\pi} Y_l^m Y_l^m d\phi d\theta \\ &= \sum_{m'=-l}^{l} (D_{mm'}^l)^2 \int_{\theta=0}^{\pi}\int_{\phi=0}^{2\pi} Y_l^{m'} Y_l^{m'} d\phi d\theta \\ &= \sum_{m'=-l}^{l} (D_{mm'}^l)^2 \\ &= LHS \end{aligned} \tag{30}$$

REFERENCES

[1] S. Ba and J. M. Odobez, "Probabilistic head pose tracking evaluation in single and multiple camera setups," in *Multimodal Technologies for Perception of Humans*, 2008, pp. 276–286.

[2] Q. Cai, A. C. Sankaranarayanan, Q. Zhang, Z. Zhang, and Z. Liu, "Real time head pose tracking from multiple cameras with a generic model," in *CVPR Workshops*, June 2010, pp. 25–32.

[3] D. Beymer and T. Poggio, "Face recognition from one example view," in *IEEE International Conference on Computer Vision*, June 1995, pp. 500–507.

[4] X. Chai, S. Shan, X. Chen, and W. Gao, "Locally linear regression for pose-invariant face recognition," *IEEE Trans. on Image Processing*, vol. 16, pp. 1716–1725, July 2007.

[5] H. S. Lee and D. Kim, "Generating frontal view face image for pose invariant face recognition," *Pattern Recognition Letters*, vol. 27, pp. 747–754, May 2006.

[6] V. Blanz, T. Grother, P. J. Phillips, and T. Vetter, "Face recognition based on frontal views generated from non-frontal images," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 454–461.

[7] S. Li, X. Liu, X. Chai, H. Zhang, S. Lao, and S. Shan, "Morphable displacement field based image matching for face recognition across pose," in *European Conference on Computer Vision*, October 2012, pp. 102–115.

[8] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999, pp. 187–194.

[9] ——, "Face recognition based on fitting a 3d morphable model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.

[10] P. Breuer, K.-I. Kim, W. Kienzle, B. Scholkopf, and V. Blanz, "Automatic 3d face reconstruction from single images or video," in *IEEE International Conference on Automatic Face and Gesture Recognition*, September 2008, pp. 1 –8.

[11] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 1994, pp. 84–91.

[12] ——, "Multi-view face recognition by nonlinear tensor," in *International Conference on Pattern Recognition*, December 2008, pp. 1–4.

[13] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *European Conference on Computer Vision*, October 2012, pp. 808–821.

[14] A. Li, S. Shan, and W. Gao, "Coupled bias-variance tradeoff for cross-pose face recognition," *IEEE Trans. on Image Processing*, vol. 21, no. 1, pp. 305–315, 2012.

[15] C. D. Castillo and D. W. Jacobs, "Using stereo matching for 2-D face recognition across pose," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[16] ——, "Wide-baseline stereo for face recognition with large pose variation," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 537–544.

[17] T. Kanade and A. Yamada, "Multi-subregion based probabilistic approach towards pose-invariant face recognition," in *IEEE International Symposium on Computational Intelligence in Robotics Automation*, vol. 2, July 2003, pp. 954–959.

[18] S. Lucey and T. Chen, "Learning patch dependencies for improved pose mismatched face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 909–915.

[19] J. J. Yokono and T. Poggio, "A multiview face identification model with no geometric constraints," in *IEEE International Conference on Automatic Face and Gesture Recognition*, April 2006, pp. 493–498.

[20] A. B. Ashraf, S. Lucey, and T. Chen, "Learning patch correspondences for improved viewpoint invariant face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2008, pp. 1–8.

[21] S. J. D. Prince, J. H. Elder, J. Warrell, and F. M. Felisberti, "Tied factor analysis for face recognition across large pose differences," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 970–984, June 2008.

[22] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 2707–2714.

[23] Q. Yin, X. Tang, and J. Sun, "An associate-predict model for face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 497–504.

[24] F. Schroff, T. Treibitz, D. Kriegman, and S. Belongie, "Pose, illumination and expression invariant pairwise face-similarity measure via Doppelgänger list comparison," in *IEEE International Conference on Computer Vision*, November 2011, pp. 2494–2501.

[25] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 1–8.

[26] Y. Li, S. Gong, J. Sherrah, and H. Liddell, "Support vector machine based multi-view face detection and recognition," *Image and Vision Computing*, vol. 22, pp. 413–427, 2004.

[27] I. Kotsia, N. Nikolaidis, and I. Pitas, "Frontal view recognition in multiview video sequences," in *International Conference on Multimedia and Expo*, June 2009, pp. 702 –705.

[28] K. C. Lee, J. Ho, M. H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003, pp. 313–320.

[29] O. Arandjelovic and R. Cipolla, "A pose-wise linear illumination manifold model for face recognition using video," *Computer Vision and Image Understanding*, vol. 113, no. 1, pp. 113–125, 2009.

[30] R. Wang, S. Shan, X. Chen, and G. Wen, "Manifold-manifold distance with application to face recognition based on image set," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

[31] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell, "Face recognition with image sets using manifold density divergence," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 581–588.

[32] X. Liu and T. Chen, "Video-based face recognition using adaptive hidden Markov models," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003.

[33] G. Aggarwal, A. K. Roy-Chowdhury, and R. Chellappa, "A system identification approach for video-based face recognition," in *International Conference on Pattern Recognition*, August 2004, pp. 175–178.

[34] A. K. Roy-Chowdhury and R. Chellappa, "Face reconstruction from monocular video using uncertainty analysis and a generic model," *Computer Vision and Image Understanding*, vol. 91, pp. 188–213, 2003.

[35] Z. Zhang, Z. Liu, D. Adler, M. F. Cohen, E. Hanson, and Y. Shan, "Robust and rapid generation of animated faces from video images: a model-based modeling approach," *International Journal of Computer Vision*, vol. 58, no. 2, pp. 93–119, 2004.

[36] D. Jiang, Y. Hu, S. Yan, L. Zhang, H. Zhang, and W. Gao, "Efficient 3D reconstruction for face recognition," *Pattern Recognition*, vol. 38, pp. 787–798, June 2005.

[37] A. Pnevmatikakis and L. Polymenakos, *Far-field, multi-camera, video-to-video face recognition*. InTech, 2007, pp. 468–486.

[38] M. Nishiyama, M. Yuasa, T. Shibata, T. Wakasugi, T. Kawahara, and O. Yamaguchi, "Recognizing faces of moving people by hierarchical image-set matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[39] K. Ramnath, S. Koterba, J. Xiao, C. Hu, I. Matthews, S. Baker, J. Cohn, and T. Kanade, "Multi-view AAM fitting and construction," *International Journal of Computer Vision*, vol. 76, pp. 183–204, February 2008.

[40] X. Liu and T. Chen, "Pose-robust face recognition using geometry assisted probabilistic modeling," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 502–509.

[41] ——, "Face mosaicing for pose robust video-based recognition." in *Asian Conference on Computer Vision*, vol. 2, November 2007, pp. 662–671.

[42] A. C. Sankaranarayanan, A. Veeraraghavan, and R. Chellappa, "Object detection, tracking and recognition for multiple smart cameras," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1606–1624, 2008.

[43] J. Yoder, H. Medeiros, J. Park, and A. C. Kak, "Cluster-based distributed face tracking in camera networks," *IEEE Trans. on Image Processing*, vol. 19, pp. 2551–2563, October 2010.

[44] B. Song and A. K. Roy-Chowdhury, "Robust tracking in a camera network: a multi-objective optimization framework," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 4, pp. 582–596, August 2008.

[45] B. Song, A. T. Kamal, C. Soto, C. Ding, J. A. Farrell, and A. K. Roy-Chowdhury, "Tracking and activity recognition through consensus in distributed camera networks," *IEEE Trans. on Image Processing*, vol. 19, no. 10, pp. 2564–2579, October 2010.

[46] A. K. Roy-Chowdhury and B. Song, *Camera Networks: The Acquisition and Analysis of Videos over Wide Areas*, ser. Synthesis Lectures on Computer Vision. Morgan & Claypool Publishers, 2012.

[47] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, 2003.

[48] R. Ramamoorthi, "Analytic PCA construction for theoretical analysis of lighting variability in images of a convex lambertian object," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 10, pp. 1322–1333, 2002.

[49] 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*, vol. 28, no. 3, pp. 351 –363, March 2006.

[50] Z. Yue, W. Zhao, and R. Chellappa, "Pose-encoded spherical harmonics for face recognition and synthesis using a single image," *EURASIP Journal on Advances in Signal Process*, vol. 2008, pp. 1–18, January 2008.

[51] T. Brocker and T. Dieck, *Representations of Compact Lie Groups*. Springer, 2003.

[52] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, June 2003, pp. 156–164.

[53] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002.

[54] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, pp. 137–154, May 2004.

[55] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 1998, pp. 232–237.

[56] P. Felzenszwalb, D. McAllester, and D. Ramaman, "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2008, pp. 1–8.

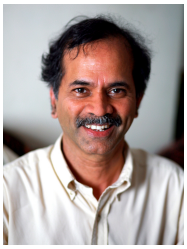
[57] S. K. Zhou and R. Chellappa, "From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel Hilbert space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 917–929, 2006.

[58] M. Du, A. C. Sankaranarayanan, and R. Chellappa, "Face tracking and recognition in a camera network," in *Multibiometrics for Human Identification*, B. Bhanu and V. Govindaraju, Eds. Cambridge University Press, 2011, pp. 235–257.

[59] R. Chellappa, M. Du, P. K. Turaga, and S. K. Zhou, "Face tracking and recognition in video," in *Handbook of Face Recognition, 2nd Edition*, S. Z. Li and A. K. Jain, Eds. Springer, 2011, pp. 323–351.

Ming Du received the BS degree in Electrical Engineering from Beijing Institute of Technology in 2002, and the MS degree from the Department of Electrical and Computer Engineering at Ryerson University, Canada in 2005. He is currently a PhD candidate in the Department of Electrical and Computer Engineering at University of Maryland, College Park. His research interests are in the field of computer vision and statistical pattern recognition, especially video-based face detection, tracking and recognition. He is also working on video analysis based on machine learning algorithms.

Aswin C. Sankaranarayanan is an Assistant Professor in the ECE Department at Carnegie Mellon University, Pittsburgh, PA. His research interests lie in the areas of computer vision, signal processing, and image and video acquisition. Prof. Sankaranarayanan received his B.Tech in Electrical Engineering from the Indian Institute of Technology, Madras in 2003 and MSc and PhD degrees from the Department of Electrical and Computer Engineering at the University of Maryland, College Park in 2007 and 2009, respectively. He was awarded the Distinguished Dissertation Fellowship by the Dept. of Electrical and Computer Engineering at the University of Maryland in 2009. He was a post-doctoral researcher at Rice University from October 2009 to December 2012.



Rama Chellappa received the B.E. (Hons.) degree in Electronics and Communication Engineering from the University of Madras, India in 1975 and the M.E. (with Distinction) degree from the Indian Institute of Science, Bangalore, India in 1977. He received the M.S.E.E. and Ph.D. Degrees in Electrical Engineering from Purdue University, West Lafayette, IN in 1978 and 1981, respectively. During 1981-1991, he was an assistant and associate professor in the department of EE-Systems at University of Southern California (USC). Since 1991, he has been a Professor of Electrical and Computer Engineering (ECE) and an affiliate Professor of Computer Science at University of Maryland (UMD), College Park. He is also affiliated with the Center for Automation Research and the Institute for Advanced Computer Studies (Permanent Member) and is serving as the Chair of the ECE department. In 2005, he was named a Minta Martin Professor of Engineering. His current research interests span many areas in image processing, computer vision and pattern recognition. Prof. Chellappa is a recipient of an NSF Presidential Young Investigator Award and four IBM Faculty Development Awards. He received two paper awards and the K.S. Fu Prize from the International Association of Pattern Recognition (IAPR). He is a recipient of the Society, Technical Achievement and Meritorious Service Awards from the IEEE Signal Processing Society. He also received the Technical Achievement and Meritorious Service Awards from the IEEE Computer Society. He is a recipient of Excellence in teaching award from the School of Engineering at USC. At UMD, he received college and university level recognitions for research, teaching, innovation and mentoring undergraduate students. In 2010, he was recognized as an Outstanding ECE by Purdue University. Prof. Chellappa served as the Editor-in-Chief of IEEE Transactions on Pattern Analysis and Machine Intelligence and as the General and Technical Program Chair/Co-Chair for several IEEE international and national conferences and workshops. He is a Golden Core Member of the IEEE Computer Society, served as a Distinguished Lecturer of the IEEE Signal Processing Society and as the President of IEEE Biometrics Council. He is a Fellow of IEEE, IAPR, OSA, AAAS and ACM and holds four patents