Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2008, Article ID 374528, 19 pages
doi:10.1155/2008/374528

Research Article

Integrated Detection, Tracking, and Recognition of Faces with Omnivideo Array in Intelligent Environments

Kohsia S. Huang and Mohan M. Trivedi

Computer Vision and Robotics Research (CVRR) Laboratory, University of California, San Diego, 9500 Gilman Drive MC 0434, La Jolla, CA 92093, USA

Correspondence should be addressed to Kohsia S. Huang, kshuang@alumni.ucsd.edu

Received 1 February 2007; Revised 11 August 2007; Accepted 25 November 2007

Recommended by Maja Pantic

We present a multilevel system architecture for intelligent environments equipped with omnivideo arrays. In order to gain unobtrusive human awareness, real-time 3D human tracking as well as robust video-based face detection and tracking and face recognition algorithms are needed. We first propose a multiprimitive face detection and tracking loop to crop face videos as the front end of our face recognition algorithm. Both skin-tone and elliptical detections are used for robust face searching, and view-based face classification is applied to the candidates before updating the Kalman filters for face tracking. For video-based face recognition, we propose three decision rules on the facial video segments. The majority rule and discrete HMM (DHMM) rule accumulate single-frame face recognition results, while continuous density HMM (CDHMM) works directly with the PCA facial features of the video segment for accumulated maximum likelihood (ML) decision. The experiments demonstrate the robustness of the proposed face detection and tracking scheme and the three streaming face recognition schemes with 99% accuracy of the CDHMM rule. We then experiment on the system interactions with single person and group people by the integrated layers of activity awareness. We also discuss the speech-aided incremental learning of new faces.

1. INTRODUCTION

Intelligent environment is a very attractive and active research domain due to both the exciting research challenges and the importance and breadth of possible applications. The central task of the intelligent environment research is to design systems that automatically capture and develop awareness of the events and activities taking place in these spaces through sensor networks [1–5]. The awareness may include where a person is, what the person is doing, when the event happens, and who the person is. Such spaces can be indoor, outdoor, or mobile, and can be physically contiguous or otherwise. An important requirement of them is to let the humans do their activities *naturally*. In other words, we do not require humans to adapt to the environments but would like the environments to adapt to the humans. This design guideline places some challenging requirements on the computer vision algorithms, especially for face detection and face recognition algorithms.

In this paper, we work toward the realization of such an intelligent environment using vision and audiosensors. To develop such a system, we propose the architecture for the networked omnivideo array (NOVA) system as shown in Figure 1 [6]. This architecture demonstrates a detailed and modularized processing of the general multilevel intelligent system using omnidirectional camera arrays. As in Figure 2, the omnidirectional cameras are composed of a hyperboloidal mirror in front of a regular camera for a full 360-degree panoramic field of view [7]; thus a large area of coverage can be provided by a relatively small number of cameras. Perspective views can also be generated from the omnidirectional videos for area-of-interest purposes. With these two types of coverage, the system can obtain a coarse-to-fine awareness of human activities. The processing modules of the NOVA system include

(1) full 3D person real-time tracking on omnivideo array [8],
(2) face analysis: detection and recognition [9–11],
(3) event detection for active visual context capture [1, 3],
(4) speech-aided incremental face learning interface.

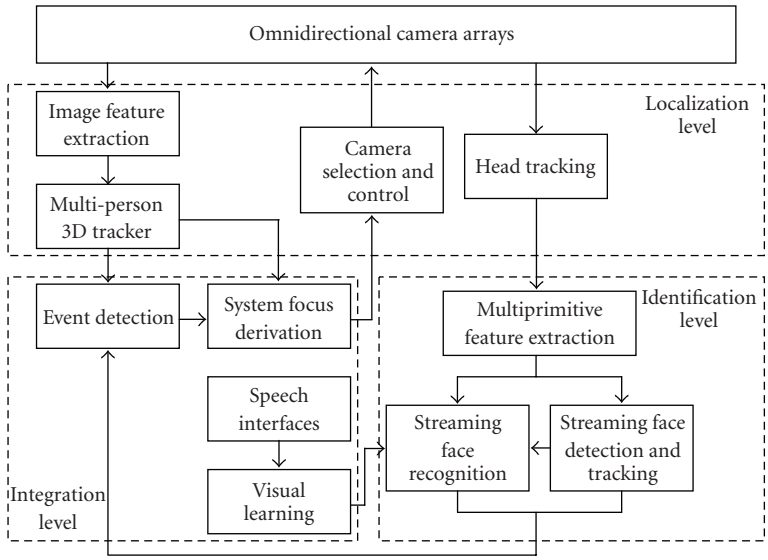
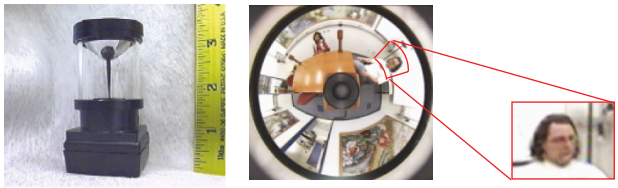


Figure 1: System architecture of the multilevel NOVA system.



(a) (b)

Figure 2: An omnidirectional camera, an omnidirectional video, and a perspective unwarping on a face.

In this NOVA architecture, camera videos are first captured and processed for signal-level visual cues such as histograms, colors, edges, and segmented objects by separate processors. The challenges at this level include robustness to illumination, background, and perspective variations.

At the next localization level, 3D tracking plays an important role in event analysis [8, 12]. It monitors the environment constantly at low resolution and derives the current position and height of a person as well as the histories and predictions of the person's trajectory. With prior knowledge of the environment, events can be detected from the tracking information; for example, one person enters the room and goes beside a table. The challenges at this level include the speed, accuracy, and robustness of the tracker, as well as the scalability of the semantic database which allows for incremental updating when new events are detected.

Motion-related events then trigger the system to capture human details to derive higher semantic information. Given the immense human-related visual contexts that can be derived, we include facial contexts of face detection and face recognition. These contexts will give the system awareness about what the subjects are doing and who they are within the environment. A suitable camera can be chosen to generate a perspective that covers the event at a better resolution, for example, perspective on a person around the head area for face capture and person identification. These face analysis modules are very active research topics since extensive visual learning is involved. We note that the perspectives generated electronically from omnicameras have higher pivot dynamics than mechanical pan-tilt-zoom (PTZ) cameras; yet PTZ cameras have higher resolution. Therefore, at situations where speed is critical, omnicameras are preferable. The challenges at this level include speed, accuracy, and robustness of the view generation and recognition modules.

Finally, the results of multiple levels of visual context analysis need to be integrated to develop an awareness of the human activities. The detected events of the lower levels are spatial-temporally sorted to derive interested spots in the space. It is noted that while the system focuses on the interested events, other activities are still being monitored by the lower levels. If something alters the priority, the system shifts its focus of interest.

The primary objective of this paper is to design such an end-to-end integrated system which takes video array inputs and provides face-based person identification. The proposed NOVA architecture for real-world environments is actually quite ambitious compared to other intelligent systems. As discussed in the survey in [6], the majority of researches emphasize on the individual components, but very few have covered high-level integrated activity awareness. In this paper, our main contributions include.

(1) A multilevel semantic visual analysis architecture for person localization, facial tracking and identification, and integrated event capture and activity awareness from the networked omnivideo array.

(2) The face analysis algorithms utilize the temporal continuity of faces in the videos in order to enhance the robustness to real-world situations and allow for natural human

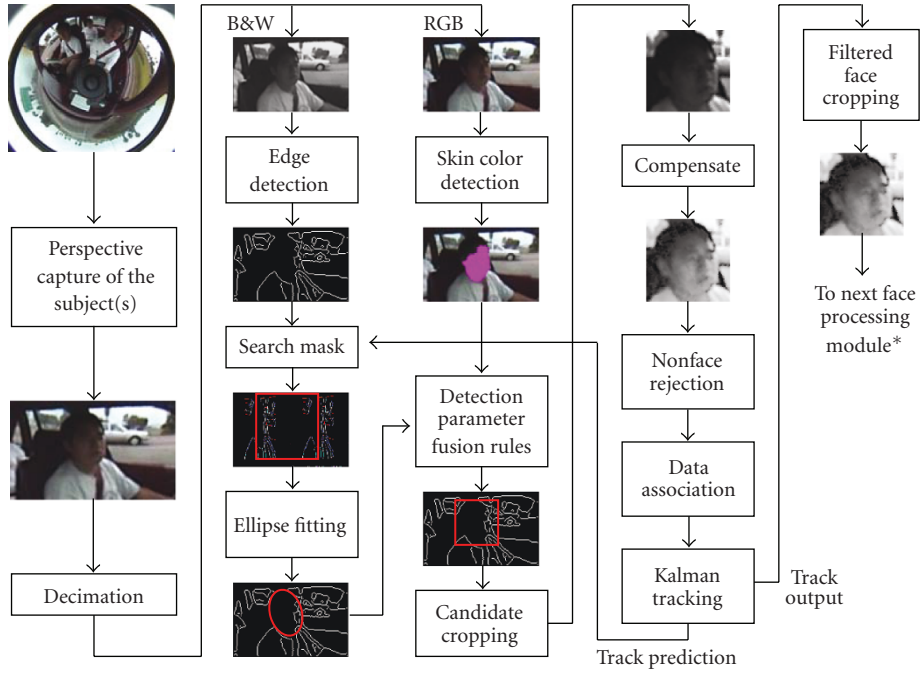


Figure 3: The integrated "closed-loop" face detection and tracking on an omnivideo.

activities; multiple image feature detection and closed-loop tracking enable our face detection and tracking to work in extreme lighting changes; accumulation of matching scores along the video boosts our face recognition accuracy.

(3) Integrated system experiments demonstrate the semantic activity awareness of single and multiple people events as well as multimodal face learning in real-world environments.

For person localization in the NOVA system, we have extensively studied real-time 3D tracking on omnivideo arrays in [8]; so it will not be discussed again in this paper. In the following sections, we will present our video-based face detection, face tracking, and face recognition algorithms in detail. Finally, integrated event detection and speech-aided incremental face learning will be demonstrated as the examples of the integrated system capability.

2. ROBUST MULTIPRIMITIVE FACE DETECTION AND TRACKING

In intelligent systems, human-computer interaction including person identification and activity analysis has been an active research field, within which face analysis is the central focus [9–11, 13]. However, it is known that without an accurate, robust, and efficient face detection as the front-end module, successful face analysis, including face orientation estimation and face recognition, cannot be achieved [14]. Robust and fast face searching is a crucial primer for face detection. It seeks all the possible faces in the captured image regardless of poses, scales, and appearances of the faces. Once a face candidate is found, it can be verified as a face or nonface by a face classifier [15, 16]. In the last decade, there has been a lot of face detection research conducted [9, 10]. Among these methods, face candidates in the image are searched by view-based methods [15–20] and feature-based methods [21–24]. In view-based methods, generally component analysis [15, 17–19], wavelet [16], and statistical approaches [20] are used. In feature-based methods, various types of features are used such as edge [22], motion [25, 26], color [27], gray-level [28], shape [21, 23], and combination of these features [24]. In addition, video-based face detection methods also utilize temporal continuity of faces in a video sequence to enhance accuracy [27, 29, 30]. Note that many single-frame algorithms use multiscale window scanning to locate the faces, especially for view-based methods [9, 16, 20]. As the window searches across the image step by step and scale by scale, a face classifier is applied to the size-equalized candidate at each location. This approach is time-consuming and is not plausible for high frame-rate cases. In this section, we propose a multiprimitive video-based closed-loop face detection and tracking algorithm [31]. Unlike single feature-based methods, our multiprimitive method combines the advantages of each primitive to enhance robustness and speed of face searching under various challenging conditions such as occlusion, illumination change, cluttered background, and so forth. The face candidates found by multiprimitive face searching are then verified by a view-based face classifier. Then, video-based face tracking interpolates the single-frame detections across frames to mitigate fluctuations and enhance accuracy. Therefore, this is a two-fold enhanced face detection algorithm by the combination of multiprimitive face searching in image domain and temporal interpolation across the video.

The process of the proposed closed-loop face detection and tracking algorithm is illustrated in Figure 3. For face searching, we chose skin-color and elliptical edge features in

this algorithm to quickly find possible face locations. Using these two primitives, time-consuming window scanning can be avoided and face candidates can be quickly located. Skin color allows for rapid face candidate searching, yet it can be affected by other skin-tone objects and is sensitive to the lighting spectrum and intensity changes. Elliptical edge detection is more robust in these cases, yet it needs more computation and is vulnerable to highly cluttered backgrounds. These two primitives tend to complement each other [24]. The subject video is first subsampled in image resolution to speed up the processing. On the skin-color track, skin-tone blobs are detected [27] if their area is above a threshold. The parameters of the face cropping window are then evaluated from the geometric moments of the blob [1, 12]. On the edge track, face is detected by matching an ellipse to the face contour. We note that direct ellipse fitting from edge detections by randomized Hough transform [32] or least-squares [33] is not feasible here since the aspect ratio and pose of the detected ellipse are not constrained and improper ellipse detections for faces have to be discarded; thus they waste much computation resources on these improper detections and are inefficient for real-time purpose. Other approaches match a set of predefined ellipses to the edge pixels [22, 24, 30]. Our method is a combination of their advantages. First, the horizontal edge pixels are linked to a horizontal line segment if their distance is below a threshold and the image intensity gradient is nearly vertical. These line segments represent possible locations of the top of head. Then, a head-resembling ellipse template is attached along the horizontal edge links at the top pixel of the ellipse. The aspect ratios, rolls, and sizes of the ellipse templates are chosen to be within usual head pose ranges. Then, the matching of ellipse templates to image edges is done by finding the maximum ratio

$$R = \frac{(1 + I_i)}{(1 + I_e)} \tag{1}$$

for all the ellipse-edge attachments, where

$$I_i = \frac{1}{N_i} \sum_{k=1}^{N_i} w_k \cdot p_k \tag{2}$$

is a weighted average of p_k over a ring zone just inside the ellipse with higher weights w_k at the top portion of the zone so that the ellipse tends to fit the top of the head, N_i is the number of edge pixels within the ellipse interior ring zone, and

$$I_e = \frac{1}{N_e} \sum_{k=1}^{N_e} p_k \tag{3}$$

is the averaged p_k over a ring zone just outside the ellipse, N_e is the number of edge pixels within the ellipse exterior ring zone. In (2) and (3), the value

$$p_k = \left| n_k \cdot g_k \right| \tag{4}$$

is the absolute inner product of the normal vector on the ellipse n_k with the image intensity gradient vector g_k at the edge pixel k. This inner product forces the image intensity gradients at the edge pixels to be parallel to the normal vectors on the ellipse template, thus reducing the false detections of using gradient magnitude alone as in [22]. This method also includes a measure which speeds up the ellipse search. It only searches along the edges at the top of human heads instead of every edge pixel in the image as in [24]. This scheme enables the full-frame ellipse search to run in real time.

Figure 4: Illumination compensation of the face video. The left is the originally extracted face image, the middle is the intensity plane fitted to the intensity grade of the original face image, and the right is the compensated and equalized face image.

After the skin blobs and face contour ellipses are detected, their parameters are fused to produce the face candidate cropping window. The square cropping window is parameterized by the upper-left corner coordinates and the size. For each skin-tone blob window, we find a nearby ellipse window of similar size and average the upper-left corner coordinates and window sizes of the two windows for the face candidate cropping window. The weighting between the skin-tone blob and the ellipse is adjusted to yield the best detection accuracy experimentally. If there is no ellipse detected, only the skin-tone blobs are used, and vice versa for the ellipses.

The detected face windows then crop the face candidates from the perspective image and scale them to 64 × 64 size. These face candidates are then compensated for uneven illumination [34]. As shown in Figure 4, illumination compensation is done by fitting a plane $z = ax + by + c$ to the image intensity by least squares, where z is the pixel intensity value and (x, y) is the corresponding image coordinate:

$$\underbrace{\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}}_{Z} = \underbrace{\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix}}_{P} \Longrightarrow P = \left(A^T A\right)^{-1} A^T \cdot Z. \tag{5}$$

Then, we verify these compensated images by distance from feature space (DFFS) [9, 15] to reject nonface candidates. We first construct the facial feature subspace by principal component analysis (PCA) on a large set of training face images of different persons, poses, illuminations, and backgrounds. The facial feature subspace is spanned by the eigenvectors of the correlation matrix of the training face image vectors which are stretched row by row from the compensated training face images as in Figure 4. Illumination compensation is needed since PCA method is sensitive to illumination variations. Then, given a face image vector, the DFFS value is computed as the Euclidean distance between the face image vector and its projection vector in the facial feature subspace.

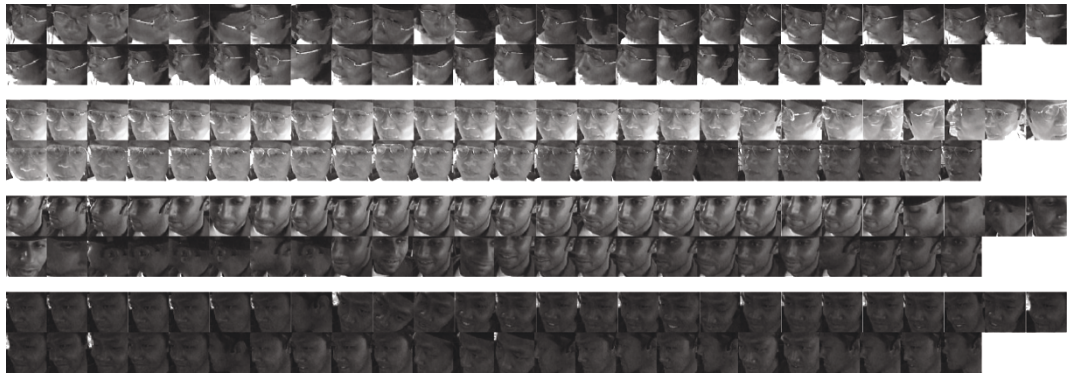


Figure 5: Some driving face videos from the face detection and tracking showing different identities, different head and facial motion dynamics, and uneven varying illuminations. These frames are continuously taken every 10 frames from the face videos.

The face candidate is rejected to be a valid face image if this distance is larger than a preset DFFS bound.

After nonface rejection, the upper-left corner coordinates and the size of the justified face cropping window are associated with the existing tracks by nearest neighborhood then used to update a constant velocity Kalman filter [35] for face tracking as

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \overset{\Delta}{\mathbf{x}}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{I} & T\cdot\mathbf{I} \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \overset{\Delta}{\mathbf{x}}(k) \end{bmatrix} + \begin{bmatrix} T^2\cdot\dfrac{\mathbf{I}}{2} \\ T\cdot\mathbf{I} \end{bmatrix} \nu(k), \qquad \mathbf{y}(k) = \begin{bmatrix} \mathbf{I} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \overset{\Delta}{\mathbf{x}}(k) \end{bmatrix} + \omega(k), \tag{6}$$

where the state $\mathbf{x}$ and measurement $\mathbf{y}$ are 3×1, and $\mathbf{I}$ is a 3×3 identity matrix. T is the sampling interval or frame duration that is updated on the fly. The covariance of measurement noise $\omega(k)$ and the covariance of random maneuver $\nu(k)$ are empirically chosen for a smooth but agile tracking. The states are used to interpolate detection gaps and predict the face location in the next frame. For each track, an elliptical search mask is derived from the prediction and fed back to the ellipse detection for the next frame as shown in Figure 3. This search mask speeds up the ellipse detection by minimizing the ellipse search area. It also helps to reduce false positives.

A face track is initialized when a single-frame face is detected for several consecutive frames. Once the face is found and under tracking, the ellipse search window can be narrowed down from full-frame search. The track is terminated when the predicted face location is classified as nonface for some consecutive frames. Track initialization helps to filter sporadic false positive detections, and track termination helps to interpolate discontinuous true positive detections. Usually we set the termination period longer to keep the track continuity.

3. STREAMING FACE RECOGNITION SCHEMES

In the intelligent room applications, single-frame-based face recognition algorithms are hardly robust enough under unconstrained situations such as free human motion, head pose, facial expression, uneven and changing illumination, different backgrounds, sensor noise, and many other human and physical factors as illustrated in Figure 5 [11, 15, 18]. For single-frame methods, some efforts have been devoted to loose the environmental constraints [11, 36], yet they only cope with limited situations and may consume much computation power. On the other hand, since it is very easy to obtain real-time face videos with face detection and tracking on video cameras, fully utilizing the spatial/temporal image information in the video-by-video-based face recognition methods would enhance performance by integrating visual information over frames. Some existing methods are based on mutual subspace method [37] and incremental decision tree [38, 39]. The mutual subspace method finds the subspace principal axes of the face images in each video sequence and compares the principal axes to those of the known classes by inner products. Another method models the distribution of face sequences in the facial feature space and classifies distributions of identities by Kullback-Leibler divergence [40, 41]. Among these few methods, facial distributions of the identities are modeled and the unknown density is matched to the identified ones in order to recognize the face.

In this paper, we propose another approach [42] of combining principle component analysis (PCA) subspace feature analysis [15] and hidden Markov models (HMMs) time sequence modeling [43, 44] because it is straightforward to regard a video as a time series like a speech stream. Observing Figure 5, we can see that the identity information of each person's face video is blended with different face turning dynamics as well as different fluctuations of illumination and face cropping alignments. In terms of the subspace features, the facial feature distribution of a certain pose would be scattered by perturbations including illumination changes, misalignments, and noises, yet the distribution would be shifted along some trajectory as the face turns [45]. These dynamics and scattering can be captured by an HMM with Gaussian mixture observation models, and the HMM states would represent mainly different face poses with some perturbations. Thus, by monitoring how the recognition performance changes with the model settings, we wish to investigate how the identity information is related to these

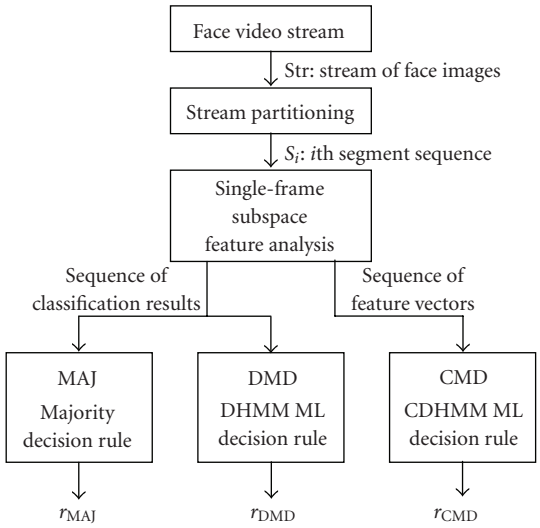


Figure 6: The streaming face recognition (SFR) architecture of the NOVA system.

factors so we can best work out the identification. In order to recognize people, we propose the video-based decision rules to classify the single-frame recognition results or visual features of the face frames in a video segment either by the majority voting rule or by maximum likelihood (ML) rules for the HMMs of each registered person. The performance of the proposed schemes is then evaluated on our intelligent room system as a testbed.

Suppose we have a face image stream $\text{Str} = \{f_1, f_2, f_3, \ldots\}$ available from the NOVA system. Similar to speech recognition [43, 44], the face image stream is then partitioned into overlapping or nonoverlapping segment sequences of fixed length L, $S_i = \{f_{K_i+1}, f_{K_i+2}, \ldots, f_{K_i+L}\}$, $S_i \subset \text{Str}$, $K_i = (i-1)K$, $i = 1, 2, 3, \ldots$, where $0 < K \leq L$ is a fixed advance length. The segments are overlapping if $K < L$. Also suppose we have M individuals in the set $I = \{1, 2, \ldots, M\}$ who are the subjects of the face image sequences. The streaming face recognition (SFR) schemes we propose here are shown in Figure 6.

3.1. Single-frame subspace feature analysis

The single-frame subspace feature analysis we have applied is an alteration to the standard eigenface PCA method [11, 15]. The major differences are as follows. (a) The eigenvector basis is generated by the *correlation* matrix of training faces instead of the covariance matrix, and (b) the projection vector of a test face image on the eigenvector basis is *normalized* as in [37]. In this manner, the single-frame face recognition would be less subject to illumination changes, because by (a) the norm of a projection vector in the eigenvector subspace is proportional to the intensity of the face image [46] and by (b) the intensity change of face images due to illumination change can thus be normalized as will be detailed below.

Suppose we have D training face vectors $t_1, t_2, \ldots, t_D$ of M individuals. For standard eigenface PCA [15], first the mean face $\mu = (1/D)\sum_{k=1}^{D} t_k$ is constructed. Next, the covariance matrix Ψ is computed as $(1/D)\sum_{l=1}^{D} \delta_l \delta_l^T$, where $\delta_l = t_l - \mu$. Then, the orthonormal eigenvectors of Ψ, that is, the eigenfaces, span the facial feature subspace centered at μ. Thus, given a new face image f, its projection in the eigenface subspace is the vector of the inner products of $(f - \mu)$ with the eigenfaces. Now suppose only the illumination intensity is changed and the poses of the person and the camera are not changed, then the intensity of the pixels of f would be proportional to the illumination intensity [46]. Since the nonzero μ does not reflect such illumination change, it would be difficult to compensate for the illumination change with standard eigenface method.

On the other hand, for the correlation-based PCA, since the mean face μ is not computed and is set to zero, the eigenvectors of the training set $\mathbf{T}$ are zero-centered and can be evaluated by singular value decomposition (SVD) as

$$\mathbf{T} = \begin{bmatrix} t_1 & t_2 & \cdots & t_D \end{bmatrix} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T, \tag{7}$$

where $\mathbf{U} = [\mathbf{u_1}\ \mathbf{u_2}\ \cdots\ \mathbf{u}_n]$ are the eigenvectors of the correlation matrix $\mathbf{T}\mathbf{T}^T$ of the training faces, n is the dimension of t_i's, and the singular values in $\boldsymbol{\Sigma}$ are in descending order. Thus, the zero-centered feature subspace can be spanned by the first D orthonormal eigenvectors $\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_D}$. For dimensionality reduction, first $d < D$ eigenvectors are utilized for the feature subspace $\mathfrak{I}$.

For the new face image f, its feature vector in $\mathfrak{I}$ is

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \cdots & x_d \end{bmatrix}^T, \tag{8}$$

where the projections $x_i = \langle f, \mathbf{u}_i \rangle = f^T \mathbf{u}_i$, $i = 1, 2, \ldots, d$. For recognition, we use the normalized feature vector

$$\hat{\mathbf{x}} = \frac{\mathbf{x}}{\|\mathbf{x}\|}. \tag{9}$$

We denote the procedure (8)-(9) as $\hat{\mathbf{x}} = \text{Projn}(f)$. Since the pixel intensity of f is proportional to illumination intensity from zero upward, the norm $\|\mathbf{x}\|$ is also proportional to the illumination intensity. Thus, the proportion of illumination changes in the feature vector $\hat{\mathbf{x}}$ can be compensated by this *correlation normalization*. The original face image can then be approximately reconstructed as $f \approx \sum_{i=1}^{d} x_i \mathbf{u}_i = \|\mathbf{x}\| \sum \hat{x}_i \mathbf{u}_i$.

At this stage, single-frame face recognition result can be drawn by the nearest-neighborhood decision rule as

$$r_{\text{SF}} = \text{ID}\Big(\arg \min_k \|\hat{\mathbf{x}} - \hat{\mathbf{t}}_k\| \Big), \tag{10}$$

where $\hat{\mathbf{t}}_k = \text{Projn}(t_k)$, $k = 1,2,\ldots,d$, and $\text{ID}(k)$ returns r if t_k is a training face image of individual r, $r \in I$. The procedure of (8)–(10) can be denoted as $r_{\text{SF}} = \text{SF}(f)$.

3.2. The majority decision rule

The input to the majority decision rule (MAJ) is a sequence of single-frame recognition results:

$$R_i = \{r_{\text{SF1}}, r_{\text{SF2}}, \ldots, r_{\text{SFL}}\}_i = \text{SF}(S_i), \tag{11}$$

where $r_{SFj} \in I,\ j = 1, 2, \ldots, L$. Then, the MAJ rule decides the streaming face recognition result of S_i as the r_{SF} that occurs most frequently in R_i as

$$r_{MAJ} = \arg\max_{m \in I} p_m, \tag{12}$$

where $p_m = \sum_{j=1}^{L} \text{Ind}\{r_{SFj} = m\}/L$, and $\text{Ind}\{A\}$ is an indicator function which returns 1 if event A is true, otherwise 0 is returned. We denote this majority voting process of (11) and (12) as $r_{MAJ} = \text{MAJ}(S_i)$.

3.3. Discrete HMM (DHMM) ML decision rule

For DHMM ML decision rule (DMD), DHMM [44] is used to model the temporal recognition sequence R_i instead of using a simple maximum occurrence as in majority rule. Suppose the training face sequences S_i, i =1,2,3,..., belong to an individual m, $m \in I$, and $R_i = \text{SF}(S_i)$ as in (11) are sequences of the single-frame recognition results which are discrete values of I. Thus, it is straightforward to train a DHMM $\lambda_m = (\pi, A, B)_m$ of N states and M observation symbols per state for the individual m. $\pi_{1\times N} = [\pi_q]$, $q = 1, 2, \ldots, N$, is the N initial state distributions of the Markov chain, $A_{N\times N} = [a_{pq}]$, $p, q = 1, 2, \ldots, N$, is the state transition probabilities from p to q, and $B_{N\times M} = [b_q(r_{SF})]$, $q = 1, 2, \ldots, N$, $r_{SF} \in I = \{1, 2, \ldots, M\}$, is the discrete observation densities of each state q. Baum-Welch re-estimation is applied on multiple observation sequences R_i, $i = 1, 2, 3, \ldots$, [42, 43] for each individual m, $m \in I$. Then, given a test sequence $R_{test} = \text{SF}(S_{test})$, the DMD rule classifies the sequence by ML as

$$r_{DMD} = \arg\max_{m \in I} P(R_{test} \mid \lambda_m), \tag{13}$$

where

$$P(R \mid \lambda) = \sum_{q_1, \ldots, q_L} \pi_{q_1} b_{q_1}(r_{SF1}) a_{q_1 q_2} b_{q_2}(r_{SF2}) \cdots a_{q_{L-1} q_L} b_{q_L}(r_{SFL}) \tag{14}$$

is computed using forward procedure [43]. We denote the DMD rule of (8)-(9) and (13)-(14) as $r_{DMD} = \text{DMD}(S_{test})$.

3.4. Continuous density HMM (CDHMM) ML decision rule

For CDHMM ML decision rule (CMD), instead of (11), the training sequences for the CDHMM [42, 44] are sequences of normalized feature vectors by (8)-(9) as

$$\hat{X}_i = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_L\}_i = \text{Projn}(S_i), \tag{15}$$

for $i = 1, 2, 3, \ldots$, as shown in Figure 6. Again we assume that $\hat{X}_i$'s belong to an individual m in I. Thus, we train a CDHMM $\lambda_m = (\pi, A, C, \mu, U)_m$ of N states and G Gaussian mixtures per state for each individual m, $m \in I$. $\pi_{1\times N}$ and $A_{N\times N}$ are the same as in DHMM case, while $C_{N\times G}$ represents the Gaussian mixture coefficients for each state. In contrast to DHMM, Gaussian mixture approximates the multidimensional continuous observation density of $\hat{\mathbf{x}}$ for each state q, $1 \le q \le N$, as [42, 47]

$$b_q(\hat{\mathbf{x}}) = \sum_{g=1}^{G} c_{qg} \mathbf{N}(\hat{\mathbf{x}}, \mu_{qg}, U_{qg}), \tag{16}$$

where $\sum_{g=1}^{G} c_{qg} = 1$ are the nonnegative mixture coefficients, $\mathbf{N}(\cdot)$ is Gaussian density function, and μ_{qg} and U_{qg} are mean vector and covariance matrix, respectively. On the D components of $\hat{\mathbf{x}}_k$, $k = 1, 2, \ldots, L$, we pick the first d components, $d \le D$, for the d-dimensional Gaussian mixture densities $b_q(\hat{\mathbf{x}}_k)$, because the first d principal components are more prominent and save computation. Expectation maximization (EM) re-estimation procedure [42, 47] is used to train the CDHMM on multiple training sequences. Then, given a test feature vector sequence $\hat{X}_{test}$, CMD rule classifies it by ML as

$$r_{CMD} = \arg\max_{m \in I} P(\hat{X}_{test} \mid \lambda_m), \tag{17}$$

where

$$P(\hat{X} \mid \lambda) = \sum_{q_1, \ldots, q_L} \pi_{q_1} b_{q_1}(\hat{\mathbf{x}}_1) a_{q_1 q_2} b_{q_2}(\hat{\mathbf{x}}_2) \cdots a_{q_{L-1} q_L} b_{q_L}(\hat{\mathbf{x}}_L) \tag{18}$$

is computed using forward procedure. The CMD rule is a *delayed decision* in that the single-frame classification (10) is skipped and full feature details are retained until the final decision (17). The decision procedure of (15)–(18) is denoted as $r_{CMD} = \text{CMD}(S_{test})$.

4. EXPERIMENTAL EVALUATIONS

In this section, we present the experimental evaluations of the face detection and streaming face recognition algorithms. The two types of algorithms are evaluated separately with natural setups. First, the face detection algorithm is evaluated in Section 4.1. Then, in Section 4.2, the detected face videos of different subjects are collected to train, test, and compare the proposed streaming face recognition algorithms.

4.1. Face detection and tracking

Evaluation of the face detection and tracking is accomplished using an extensive array of experimental data. We collected many video clips of different setups and in different environments, including indoor, outdoor, and mobile, as shown in Figure 7. In order to evaluate the accuracy of face detection and tracking specifically, we ask the human subjects to be at static locations with respect to the omnicamera. Figure 8 shows an indoor example where ten people sitting around a meeting table are all detected from the panorama of an omnivideo. Figure 9 shows the single-person indoor face detection results. Row 1 shows the source images, row 2 shows the overlapped edge gradient strength, the skin-tone area, the detected ellipse, and the square face cropping border before Kalman tracking, and row 3 shows the cropped face images after Kalman tracking. Column 1–column 4 indicate

Figure 7: Sample images of the test video sequences for face detection and tracking on indoor, outdoor, and mobile environments. Columns from left to right show the omnidirectional videos, the unwarped panoramas, and the perspective videos of the subjects.

that the skin-tone and ellipse detections cooperate to detect faces on some difficult situations such as a turned-away face, highly cluttered background, and an invasion of nonface skin-tone objects to the face blob. Column 5 shows an extreme situation where the lights are turned off suddenly, and the face detection and tracking can still keep the face with ellipse detection.

For face tracking performance (cf. Figure 3), we tested the clips with the measurement noise variance of the Kalman filter set to 64-pixel square and the random maneuver variance set to 512-pixel square. The standard deviation of the detected face alignment within the 64×64 face video after tracking is about 7 pixels. For track initialization and termination, we set initialization period to 450 milliseconds to filter sporadic false positive face detections, and set termination period to 1700 milliseconds to interpolate discontinuous true positive face detections. Actual frames for track initialization and termination in Section 2 are converted from these periods according to the current processing frame rate. For the distance from feature space (DFFS) bound in Section 2, currently we set a sufficiently large value of 2500 so that the detector would not miss true positive faces in the image.

For face detection performance evaluation, we recorded multiple human faces in the omnivideo clips on a DV camcorder. Then, with analog NTSC video output of the camcorder and video capture card on the computer, we replay the clips of almost exactly the same starting to ending frames many times to the face detection and tracking module as in Figure 3 with different DFFS bound settings and with/without Kalman tracking. The DFFS bound matters with the true positive and false positive rates, and Kalman tracking interpolates between the single-frame face detections over the video. On each playback, the resultant video with face detection and tracking results (the face cropping window) is recorded by screen shot as shown in Figure 10. Finally, the detection counts and false positive counts are manually counted frame by frame in the resultant videos. Each frame of the test videos contains 2 or 3 faces; so the number of faces would be 2 or 3 times the number of frames in the videos. These counts are summarized in Table 1.

Table 1 lists the averaged detection rates and false positives in terms of the DFFS bound on the indoor and outdoor test sequences. The detection rate increases with the DFFS bound for all cases because increasing DFFS bound would allow more face-plausible images to be included

Figure 8: Face detections on a panorama of an indoor meeting setup.

Figure 9: Some results of the proposed multiprimitive face detection and tracking. Note that in the fifth column, there is a drastic change of illumination. See text for details.

as face images. With single-frame face detection, however, the false positives do not always increase with the DFFS bound monotonically. For outdoor setting, the trend of false positives basically increases with the DFFS bound with some exception, but it is not the case for the indoor setting. This difference between the indoor and outdoor settings would be due to more irregular backgrounds in the outdoor scene. Hence, more ellipses and more skin-tone regions can be detected and thus they increase the chance of false positives. The nonmonotonic performance of indoor single-frame false positives could also be due to noises in the video upon simple backgrounds. For these causes, we have briefly verified another indoor clip which has complex background and the false positives are higher on larger DFFS bounds as in outdoor cases. Therefore, it is desirable for further counting of the detections and false positives on videos of various backgrounds. Note that the perspective unwarping videos in Figure 10 are not of high resolution and pixel noises in the original omnivideo would cause more prominent noises in the perspective videos. With Kalman face tracking, Table 1 also indicates that both the detection rates and false positives are increased. This is due to the fact that with temporal interpolation of the Kalman filters, the durations of the true positives are lengthened. At low DFFS bounds, the false positives increase more significantly because the single-frame detections are more discontinuous and the face tracks are lost easily and go astray, causing more false positives. This effect gets better at higher DFFS bounds and the false positives after tracking reflect more directly the single-frame false positives. In addition, tracking initialization helps to reduce the false positives because it takes some frames to start a track. Therefore, if the single-frame false positives are sporadic, they would be filtered out by face tracking. This is the case for the indoor case with DFFS bounds of 2500 and 4000.

We have used our real-time videos for face detection and tracking evaluations. Note that it is also possible to test the single-frame face searching and verification separately from tracking with some face databases, for example, PIE database [48]. However, the tracking effects on the speedup measure of ellipse search window which affect the detection rate and false positives cannot be evaluated with those databases that are not video-based.

For computation complexity, the most computation-intensive part of the face detection and tracking algorithm is on multiprimitive face searching since it is a pixel-level processing. The next is on face verification because it projects the face image into PCA subspace by inner products between face image vectors. Kalman filter is the fastest module since its data involve only 2 dimensions of image location and 1 dimension of the size of the face cropping window.

Currently, we are using DFFS face classification because the PCA subspace feature analysis is also used in streaming face recognition. To further improve the false positive issues, the cascade type of face classification algorithm such as Viola-Jones could be a good choice [16]. Using boost algorithms on PCA features, we could enhance the DFFS face classifier with optimized cascading of weak DFFS face

Figure 10: Samples of indoor and outdoor test video clips for counting the face detection rates and false positives.

Table 1: Face detection and false positive rates of the indoor and outdoor test sequences on single-frame and tracking-based settings.

	DFFS bound		1500	1700	2000	2100	2500	4000
Indoor	Single-frame	Faces	2649	2652	2646	2646	2646	2649
		Detected	94 (3.6%)	435 (16.4%)	1312 (49.6%)	1501 (56.7%)	2407 (91.0%)	2645 (99.9%)
		F. Pos.	3	2	2	5	4	0
	Tracking	Faces	2661	2658	2652	2649	2652	2649
		Detected	437 (16.4%)	1294 (48.7%)	2050 (77.3%)	2418 (91.3%)	2652 (100%)	2649 (100%)
		F. Pos.	26	78	14	6	0	0
Outdoor	Single-frame	Faces	1766	1766	1766	1772	1766	1766
		Detected	119 (6.7%)	253 (14.3%)	601 (34.0%)	715 (40.4%)	1290 (73.1%)	1748 (99.0%)
		F. Pos.	93	152	179	170	221	524
	Tracking	Faces	1766	1766	1770	1770	1760	1768
		Detected	63 (3.6%)	382 (21.6%)	951 (53.7%)	1081 (61.1%)	1621 (92.1%)	1752 (99.1%)
		F. Pos.	398	439	601	409	681	510
Total	Single-frame	Faces	4415	4418	4412	4418	4412	4415
		Detected	213 (4.8%)	688 (15.6%)	1913 (43.4%)	2216 (50.2%)	3697 (83.8%)	4393 (99.5%)
		F. Pos.	96	154	181	175	225	524
	Tracking	Faces	4427	4424	4422	4419	4412	4417
		Detected	500 (11.3%)	1676 (37.9%)	3001 (67.9%)	3499 (79.2%)	4273 (96.8%)	4401 (99.6%)
		F. Pos.	424	517	615	415	681	510

classifiers which may utilize different sets of eigenvectors. For preliminary verification, we tried the Viola-Jones single-frame face detector of OpenCV library for the combined frontal face and left and right profiles using the same video clips as in Table 1. The indoor detection rate was (1761 faces)/(2661 faces) = 66.18% and the outdoor detection rate was (1149 faces)/(1770 faces) = 64.92%. Faces were not detected well while transiting between frontals and profiles mainly because of different image quality. There was no false positive in both cases. Although the single-frame detection rate is lower as compared to DFFS bound of 2500 in Table 1, it shows that the false positive rate can be much improved with boost type of cascaded face classifiers. Besides, the Schneiderman-Kanade face classifier could be another view-based approach that needs more complex and exhaustive statistical pattern classification [20].

4.2. Streaming face recognition (SFR)

In this section, the three proposed streaming face recognition schemes (MAJ, DMD, and CMD) are compared by numerical experiments on the intelligent room testbed.

Figure 11: An omni-image showing our intelligent room testbed. Perspective video of a human face can be generated from a source omnivideo.

Their accuracies are also compared to the single-frame face recognition accuracy.

4.2.1. Experimental setup

In evaluating the streaming face recognition, we used perspective view driven by 3D tracker on an indoor omnivideo array to capture the human faces, as illustrated in Figure 1. Omnivideo covers the entire room, including people's faces of different distances and with different backgrounds, as shown in Figure 11.

We have collected face video streams of 5 people. People were sitting in the testbed room and turning their faces randomly with various expressions. Single omnivideo clips were recorded on a Digital-8 camcorder and later played back to the NOVA system video input for data collection. Of every person, 9 video clips were recorded. For training session, 5 video clips were recorded for each person at different locations and backgrounds with different omnicameras in the room. The clip duration varied from 1 minute and 10 seconds to 1 minute and 30 seconds. For testing session, 4 video clips were recorded at other 4 different locations and backgrounds with different omnicameras. The clip duration varied from 50 seconds to 1 minute and 15 seconds. Some examples of the face images in the video streams are shown in Figure 12, exhibiting the live situations that NOVA streaming face recognition (SFR) needs to deal with. When playing back the videos, the NOVA perspective view and face video extraction logged data streams of both single-frame recognition results, r_{SF}'s, and single-frame feature vectors, $\hat{\mathbf{x}}$'s, of the face video for both the training and testing sessions. The number of frames logged for each person varied from 4360 to 5890 frames in the training session and from 1880 to 3980 frames in the testing session. This same set of data streams was used to compare the SFR schemes on a common ground. The SFR algorithms processed the data offline.

In the experiments, the training data streams are first used to train the HMMs of the DMD and CMD rules for each individual. Then, the testing data streams are used to compare the performances of single-frame face recognition and the MAJ, DMD, and CMD rules for each person. The performance index to be evaluated is the averaged accuracy of the correct recognition rates for the 5 individuals. Multiple trials are also experimented to find the accuracy statistics of the SFR schemes on various settings.

The purposes of the experimental design are to

(i) investigate how the recognition accuracy changes with the model settings which are related to the modeling of face turning dynamics and face image fluctuations,

(ii) compare the optimum accuracies among the single-frame and streaming face recognition schemes.

For (i), we need to find the relationship of the accuracy with respect to the number of states N, the number of Gaussian mixtures G, and the utilized dimension d of the feature vector $\hat{\mathbf{x}}$ for the DMD and CMD rules. The accuracy of the MAJ rule only depends on the length of the segment sequence L, which we tend to fix for the considerations of real-time implementation. The accuracy of single-frame face recognition does not depend on these settings and is fixed for the given testing streams. Then, for (ii), it will be straightforward to compare the best accuracies of the SFR schemes.

The data streams are partitioned into nonoverlapping segment sequences of $L = 49$ frames. L is chosen to be an odd number to avoid possible tie cases in the MAJ rule. The size of face video is 64×64, and thus the dimension of face image vector n is 4096. The dimension D of PCA feature subspace in single-frame feature analysis is chosen to be 135.

4.2.2. Results

We first compare the MAJ and DMD rules because they use the same streams of single-frame face recognition results. As shown in Figure 13, the experimental results of DMD rule are plotted with the MAJ results. The DMD accuracy depends on the number of states N of the DHMM. Four trials of the DHMM training for each N were exercised, and the mean and standard deviations of the DMD accuracy are plotted as the error bars for each N. From the 7th-order polynomial fitting of the DMD accuracies, the best accuracy is 89.7% when $N = 14$, and the worst accuracy is 86.6% when $N = 6$. The MAJ accuracy is 81.7% regardless of N.

Then, we monitor the performance of the CMD rule, starting from the simplest settings: $N = 1$, $G = 1$, $d = 1$. The dependency of CMD accuracy on d is experimented and plotted in Figure 14. The accuracies are experimented on one trial because with $N = G = 1$, the training of CDHMM parameters $\lambda = (\pi, A, C, \mu, U)$ converges to the same value. The peak accuracy is 99.0% when $d = 8$.

With $d = 8$, we then find the accuracy with respect to G, as shown in Figure 15, and N, as shown in Figure 16. Four trials are exercised for each setting and the means and standard variations are plotted as error bars. From the polynomial fittings, the accuracies decay monotonically as G or N increases.

Thus, the best accuracies of the MAJ, DMD, and CMD rules can be compared to the accuracy of the single-frame

(a) Perspective views for face capture

(b) Sample face video frames of the 5 subjects

Figure 12: Examples of the face images in the training and testing video streams. The left six are perspective views, and the right face images are automatically extracted by face detection from the perspective video. They show different facing angles, face sizes, backgrounds, expressions, and other variations that face recognition needs to cope with.

face recognition, which is the averaged value of the correct recognition rates in all the frames of the testing streams for each individual. These optimum accuracies are summarized in Table 2.

4.2.3. Analysis of the results

In this section, we first propose an interpretation on the experimental results. We also discuss the implementation complexity of the proposed SFR schemes. Analogy of these schemes to automatic speech recognition is also interesting to study. Then, future works are to be discussed.

To explain the experimental results, we start from an insight into the results of the CMD rule. After the trials of different model settings, the optimum CMD accuracy occurs when $N = G = 1$. Out of this point, the accuracy decays monotonically. It is noted that when $N = G = 1$, the likelihood computation in (18) becomes

$$P(\hat{X} \mid \lambda_m) = \sum_{q_1,\ldots,q_L} \pi_{q_1} b_{q_1}(\hat{\mathbf{x}}_1) a_{q_1 q_2} b_{q_2}(\hat{\mathbf{x}}_2) \cdots a_{q_{L-1} q_L} b_{q_L}(\hat{\mathbf{x}}_L) \mid_{\lambda_m} = b(\hat{\mathbf{x}}_1) b(\hat{\mathbf{x}}_2) \cdots b(\hat{\mathbf{x}}_L) \mid_{\lambda_m} \tag{19}$$

since π_i's and a_{ij}'s are all 1 for $N = 1$. For $G = 1$, the Gaussian mixture density in (16) is reduced to single multidimensional Gaussian density function $b(\hat{\mathbf{x}}) = N(\hat{\mathbf{x}}, \mu, U)$. The Baum-Welch training of the CDHMM λ is then degenerated to the fitting of a multidimensional Gaussian density to the training feature points in the feature subspace. For a testing sequence $\hat{X} = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_L\}$, the ML computation of (17) and (19) is actually performing a distribution matching of the points $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_L$ in the feature subspace by product rule or likelihood accumulation, as illustrated in Figure 17. As G increases, the Gaussian mixture density of (16) is broadened and the chance to overlap with other identities is increased. Hence, the accuracy decays in Figure 15. Also as $N > 1$, the CDHMM starts to model the temporal variations of the feature vectors $\hat{\mathbf{x}}$'s in the sequence $\hat{X}$ mainly due to face poses [45]. The temporal dynamics in the sequences are modeled more precisely as N increases. Because of the different temporal patterns between the training and testing sequences, the accuracy drops with N in Figure 16.

The DMD and MAJ rules are built upon single-frame face recognition results. Note that in single-frame face recognition (see (10)), the point $\hat{\mathbf{x}}$ is clustered to a training point $\hat{\mathbf{t}}_k$ by nearest neighborhood in Euclidian distance. Therefore, these decision rules would not model the density well since they approximate ellipsoids by globes in the feature subspace. In addition, as illustrated in Figure 17, some points may be deviated into other identity classes by noise or other issues. Therefore, the accuracy of the single-frame

Table 2: Comparison of the optimum accuracies of the single-frame face recognition (FR), the MAJ rule, the DMD rule, and the CMD rule.

Decision rules		Optimum accuracy	Note
Single-frame FR		75.9 %	
Streaming FR	MAJ	81.7 %	
	DMD	89.7 %	$N = 14$
	CMD	99.0 %	$N = 1, G = 1, d = 8$
Common settings: $D = 135$, $L = 49$, nonoverlapping sequences			

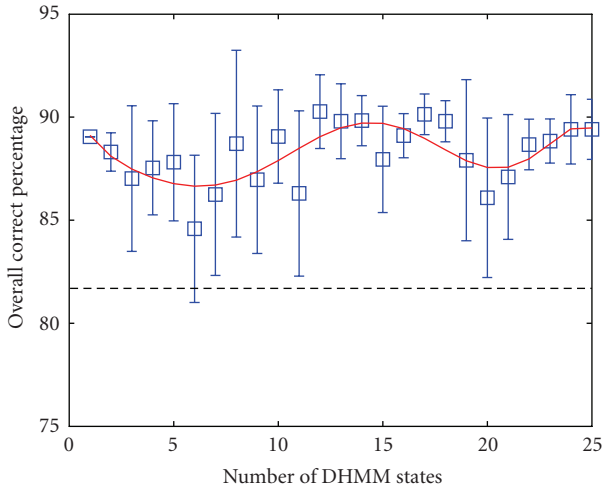


Figure 13: Accuracy of the DMD rule with respect to the number of DHMM states N. The error bars show mean and standard deviations of the experimental accuracy on four trials. Solid curve is a polynomial fitting of the mean values. Dotted line is the accuracy of the MAJ rule.

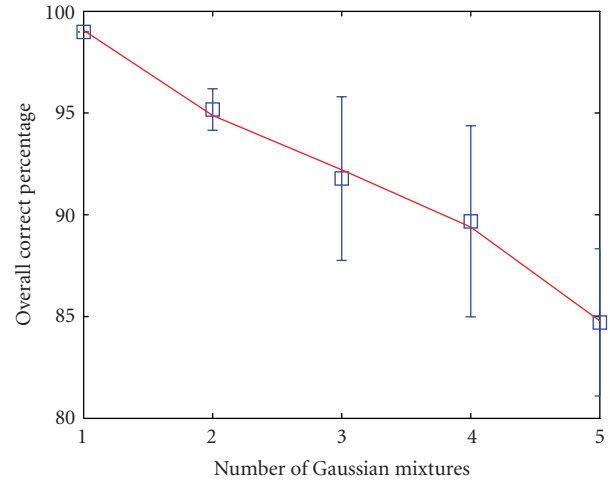


Figure 15: Overall correct percentage of the CMD rule with respect to the number of Gaussian mixtures G. The number of utilized dimensions d is 8 and the number of CDHMM states N is 1. Four trials are exercised. The solid curve is a polynomial fitting of the experimental mean values.

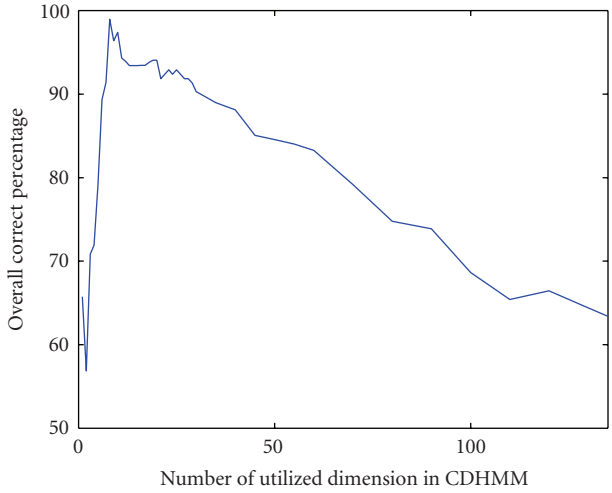


Figure 14: Accuracy of the CMD rule with respect to the number of utilized dimensions d of the feature vectors. The full dimension D of the PCA feature vectors is 135. Peak accuracy of 99% occurs when $d = 8$. Both the numbers of CDHMM states N and of Gaussian mixtures G are 1.

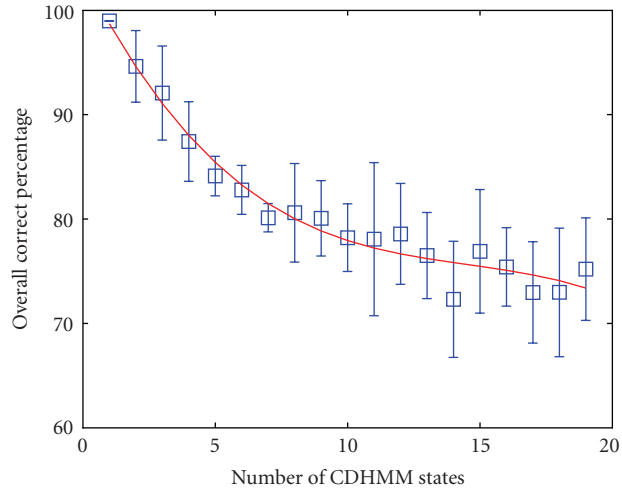


Figure 16: Overall correct percentage of the CMD rule with respect to the number of CDHMM states N. The number of Gaussian mixtures G is kept 1. Four trials are experimented and polynomial fitting is plotted.

face recognition is the lowest among the four rules. On the other hand, if the single points are collected together in a sequence, the distribution is better approximated. Hence, the MAJ accuracy is better than that of single-frame face recognition. In addition to collective single points, the DMD rule also models the temporal sequence of the points by a Markov chain. This explains the waving phenomenon in Figure 13. When $N = 1$, the DHMM is like (19) that models the joint density in a collective way. When N increases, the DHMM correlates with the dynamics of the testing temporal sequences, thus causing a resonance response. We can thus deduce that if the dynamics of the testing sequence, for example, pattern of human motion, change a lot, the resonance pattern in Figure 13 would also change. As a whole, the DMD rule performs better than the MAJ rule by collecting more information from the temporal sequences. Therefore, the accuracy performance of the decision rules is CMD > DMD > MAJ > single-frame face recognition, as in Table 2.

To summarize the above geometric interpretations, the hidden states of the HMM represent mainly the temporal dynamics of face poses, and the Gaussian mixture for each state models the disturbances of illumination changes and other noise factors. Since different persons may have

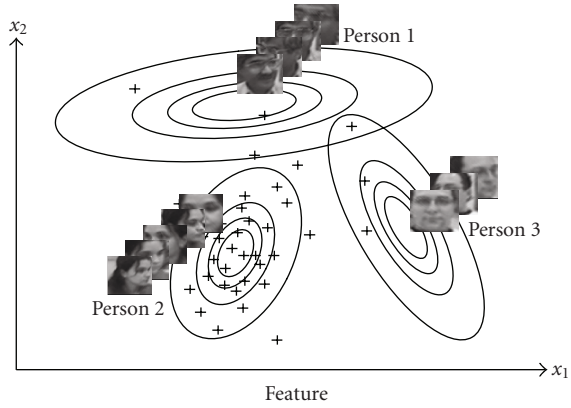


Figure 17: The geometric interpretation of the ML computation as a distribution matching in the feature subspace. The ellipses are the Gaussian density functions of the clusters, and the dots are the feature vectors in a sequence $\hat{X}$. x_1 and x_2 are the components of the feature vector.

different face turning patterns as suggested in Figure 5, using HMM to capture this identity-related information would be desirable in some situations such as driving and needs further experimental verifications with specific datasets. However, in general situations as in this experimental setup, the pattern of face turning dynamics encoded in the HMM may not match the test sequence. Therefore, we generalize the matching with $N = 1$ by fusing all different face poses into one state and simply modeling the omnipose distribution by Gaussian mixture. For the Gaussian mixture, single Gaussian $G = 1$ gives the most crispy modeling of the distribution of an identity without much overlapping to others, thus rendering the highest accuracy. For DHMM, the ball shape of distribution modeling less matches the actual distributions and thus gives lower accuracy.

Concerning the phenomenon in Figure 14, recall that the feature subspace is spanned by the principle components of the training face images ordered from the most significant one to the least significant one. From $d = 1$ to $d = 8$, the representation power grows rapidly in a polynomial fitting flavor. After $d = 8$, higher dimension starts to overfit the feature points in the identity clusters and cause more confusion among the clusters (cf. Figure 17). In other words, the curse of dimensionality starts to take effect, and hence the accuracy starts to drop gradually. It can also be implied that this turning point $d = 8$ as well as the optimum accuracy depends on the number of training samples and clusters, currently 5 people. So, further experiments with more videos and more subjects are needed for these regards.

Also for the sequence length L, although it is to be fixed for real-time implementation, we can still deduce that as L goes up, the accuracy would improve because more points better match the Gaussian density in the feature subspace. It would be worthwhile to perform more experiments to verify this viewpoint.

For implementation complexity, MAJ is the lowest because it simply collects single-frame face recognition results and does maximum finding. DMD is higher by introducing DHMM training and likelihood computations. CMD is the highest because CDHMM further involves the parameters of multidimensional Gaussian density. But it is worth the extra computation because the CMD accuracy is much higher than others.

Compared to speech recognition, the processing procedure of CMD and DMD in Figure 6 is similar to speech recognition. Speech recognition first partitions the speech stream into segment sequences, usually overlapping. It then computes features of the speech signal in the segment by cepstrum and/or other features. Then, the features of the segments are modeled by an HMM to derive the transitions of the states, which represent phonemes. In our case, this procedure is almost the same, yet the identity information is mainly embedded in the individual frames. Only person-related facial motions such as face turning and expression are related to the transitions between the frames, and the HMM states represent intermediate face poses and expressions.

In the future, facial expression tracing can be done by analyzing the transitions of HMM states using Viterbi algorithm [43]. However, PCA-based subspace feature analysis might not be sufficient to represent expression definitely. ICA-based subspace feature analysis [18] would be a powerful tool for this purpose. Learning algorithms [48, 49] can also be applied to the feature subspaces so that the recognition capability of the NOVA system can be scaled up by learning how to recognize new people dynamically from the video streams.

5. INTEGRATION EXPERIMENTS

In this section, we experiment on the interactions of the integrated NOVA system with humans. At the integration level, we combine the results from the previous levels, that is, the information of tracking, face detection, and face recognition, for event detection and responding. We test the intelligent room system on these three kinds of single-person and group activities:

(i) a person entering or exiting the room,

(ii) identity tagging on a person during tracking,

(iii) a group of people interacting in the room.

For activity (i), when a person enters or exits the room, access zones of the room are defined as shown in Figure 18. Our team has developed a context visualization environment (CoVE) interface to visualize the room composition, predefined zones, and human tracks in real time [50]. Track data are sent to a server to monitor the zones over long periods of time and to archive the tracks for later retrieval. By this integration, the passing counts of the zones with the track indices are accumulated.

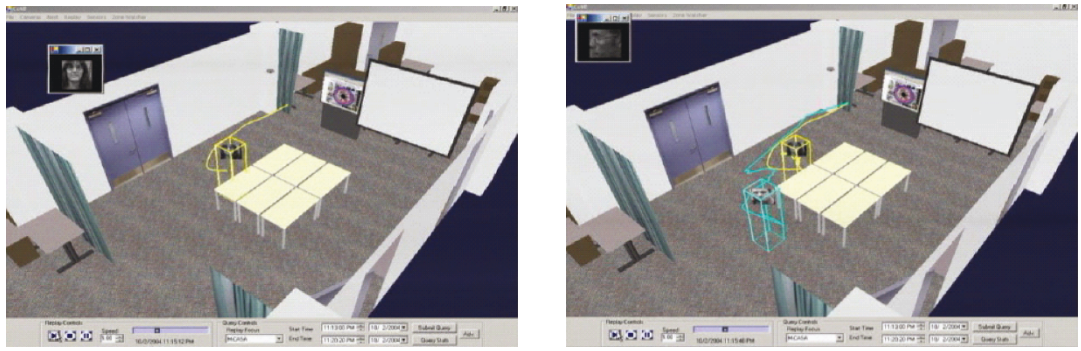
For activity (ii), a pan-tilt-zoom (PTZ) camera is driven by the NOVA tracker to capture the human face upon the person's entrance. The person is recognized by the system from the detected face video, and face image is tagged to the human volume in CoVE as shown in Figure 19.

Long-term face capture of the entering people is shown in Figure 20. This figure shows the captured human entrance

(a) (b)

(c)

Figure 18: Long-term zone watching of the CoVE interface in the intelligent room. A predefined zone is indicated by red if a person passes through it, or it remains blue. Humans are represented as color-coded rectangular cubes. Trajectories of humans have the same color as the rectangular cubes of people, respectively.



(a) (b)

Figure 19: Face video capture when a person enters into the room. The video is displayed in a subwindow and archived for later retrieval.

events over a period of ~50 minutes. The NOVA tracker monitors the room continuously and archives the entering people automatically with a time stamp. Past subjects can be retrieved with the face videos and the entrance times to the accuracy of seconds. It is suitable for automatic surveillance and forensic support applications.

For group activities (iii), the faces are captured sequentially by the system. An example of such a system-attentive scenario is shown in Figure 21, where four people sitting in the room and facing the projector screen are scanned by the closest cameras. When a new person enters, the system changes the scanning order; that is, it is "distracted."

5.1. Speech modality and visual learning

In this section, we experiment on learning a new subject for streaming face recognition in the intelligent environment. Face detection does not need further learning once it is trained with a general face database such as CMU PIE database [48]. However, for streaming face recognition as in (17), if the highest likelihood $\max_{m\in I} P(\hat{X}_{\text{test}} \mid \lambda_m)$ is below a threshold for all currently known identities, $m \in I$, then the sequence of face $\hat{X}_{\text{test}}$ is of an unknown person. It is the same for the majority rule in (12) and the discrete HMM rule in (13) when the outcome indicates an unknown identity. In these cases, an identity name must be given to the face recognizer through some interfaces, and the identity knowledge base I can be increased. For an intelligent room, it is natural and unobtrusive to take the speech modality for this incremental visual learning, as illustrated in Figure 22.

A speech command interface with certain predefined grammars is ideal for this purpose. We used IBM ViaVoice SDK and defined several question and answer grammars for the system to talk to the people in the room. If only one

Figure 20: Examples of automatic face capture and archive while people appear in the room. Video clips were captured during approximately 50-minute duration. In two-people cases (epochs of 15 : 43 : 02 and 15 : 54 : 22), the subjects are captured in turn in the same video clip.

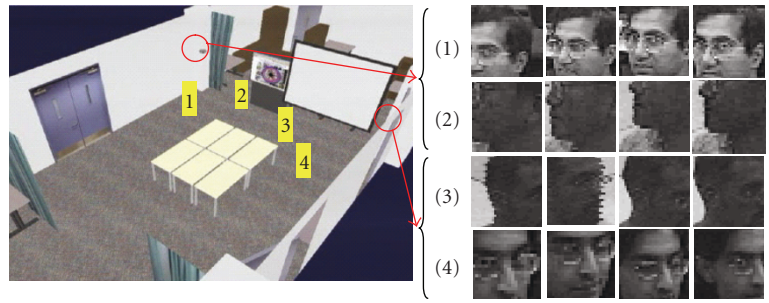


Figure 21: Face subject scanning of four persons in the intelligent room. Persons are facing the projector screen in front of the room. Videos of the persons are taken by two most nearby PTZ cameras at the corners of the room.

person is in the room and is unknown, the system will greet the person and will ask for his name. If one known person is present and another is unknown, as in Figure 22, then the system will greet the known person and ask for the other person's name. Then the identity, along with the streaming face recognition model λ_m derived from $\hat{X}_{\text{test}}$, is added to the face recognition database. Figure 23 shows an example of the scenario in Figure 22, where a known person is recognized and an unknown person is detected. The ViaVoice interface then asks for the unknown person's name from the known one, and the unknown person is learned by the streaming face recognizer.

6. CONCLUDING REMARKS

The primary objective of this research is to design an end-to-end integrated system which takes video array inputs and provides face-based person identification. This system architecture includes multiple analysis levels for person localization, facial identification analysis, and integrated event capture from the networked omnivideo array. The face analysis algorithms utilize the temporal continuity of faces in the videos in order to enhance robustness to environmental variations and allow for natural human activities. For face detection, two types of image primitives are used in cooperation to find face candidates on various challenging conditions. The face candidates are then verified to reject false detections and put them into tracking to filter and interpolate face detections across frames. The extracted face video is then analyzed to recognize the identity of person over the frames. Three types of video-based face recognition schemes collect single-frame face analysis outputs, either single-frame face recognition identities or single-frame feature vectors, in a video segment and compare the accumulated scores to make final decisions. Experimental results support the streaming face recognition scenario by showing significant improvements of recognition accuracy. With these video-based face analysis algorithms, higher-level

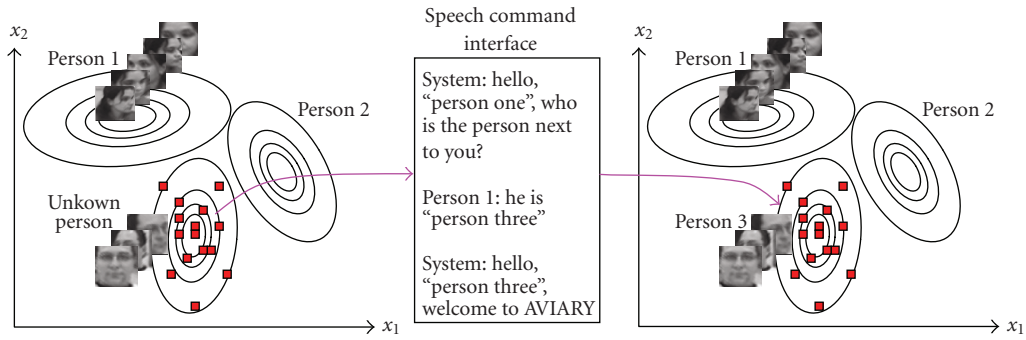
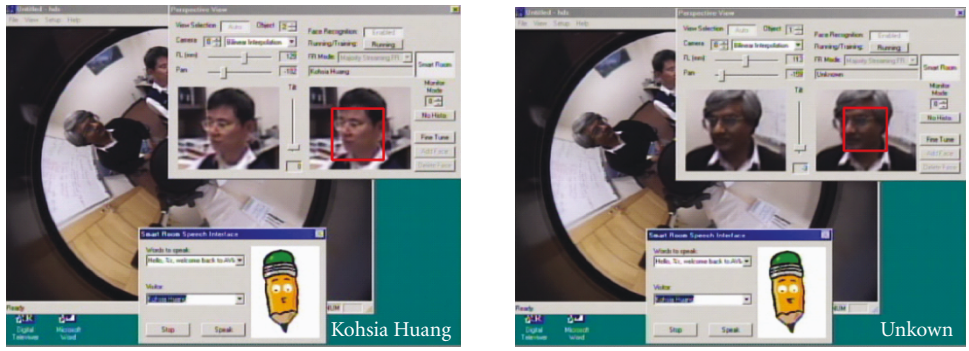


FIGURE 22: Speech command-based incremental visual learning for streaming face recognition.



(a) (b)

FIGURE 23: An experiment of the incremental learning of face recognition identity. (a) The ViaVoice interface (the cartoon pencil) first greets the recognized person. (b) Having detected the unknown person, ViaVioce then asks for his name from the known person.

awareness of human activities can be realized. The integrated analysis derives the awareness for who and where the subjects are as well as when they are there. The system knowledge of people can also be expanded through speech-aided visual learning.

The experimental studies have shown the basic feasibility for such a system in "real-world" situations. Having shown the promises of the integrated framework, further investigations should provide detailed comparative analysis of individual modules as well as the system. One key requirement would be to have well annotated database with ground truth from real-world situations.

ACKNOWLEDGMENTS

Our research is supported by the US DoD Technical Support Working Group (TSWG), Sony Electronics, Compaq Computers, DaimlerChrysler, Caltrans, and UC Discovery Grant. The authors also acknowledge the assistance of their colleagues during the course of this research. They thank Mr. Erik Murphy-Chutorian for his assistance in experiments involving Viola-Jones single-frame face detector. They are also grateful to the Guest Editor, Professor M. Pantic, and the reviewers for their insightful and valuable comments.

REFERENCES

[1] M. M. Trivedi, K. S. Huang, and I. Mikić, "Dynamic context capture and distributed video arrays for intelligent spaces," IEEE Transactions on Systems, Man, and Cybernetics Part A , vol. 35, no. 1, pp. 145–163, 2005.

[2] M. Pantic, A. Pentland, A. Nijholt, and T. Huang, "Human computing and machine understanding of human behavior: a survey," in *Proceedings of the 8th International Conference on Multimodal Interfaces (ICMI '06)*, pp. 239–248, Banff, Canada, November 2006.

[3] M. M. Trivedi, T. L. Gandhi, and K. S. Huang, "Distributed interactive video arrays for event capture and enhanced situational awareness," IEEE Intelligent Systems , vol. 20, no. 5, pp. 58–65, 2005.

[4] I. Mikic, K. Huang, and M. M. Trivedi, "Activity monitoring and summarization for an intelligent meeting room," in *Proceedings of IEEE Workshop on Human Motion*, pp. 107–112, Los Alamitos, Calif, USA, December 2000.

[5] M. M. Trivedi, K. Huang, and I. Mikic, "Intelligent environments and active camera networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 804–809, Nashville, Tenn, USA, October 2000.

[6] K. S. Huang, " *Multilevel analysis of human body, face, and gestures with networked omni video array*, Ph.D. thesis," University of California, San Diego, Calif, USA, March 2005.

[7] S. K. Nayar, "Catadioptric omnidirectional camera," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 482–488, San Juan, Puerto Rico, USA, June 1997.

[8] K. S. Huang and M. M. Trivedi, "Video arrays for real-time tracking of person, head, and face in an intelligent room," Machine Vision and Applications , vol. 14, no. 2, pp. 103–111, 2003.

[9] E. Hjelmås and B. K. Low, "Face detection: a survey," Computer Vision and Image Understanding , vol. 83, no. 3, pp. 236–274, 2001.

[10] M.-H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 24, no. 1, pp. 34–58, 2002.

[11] A. S. Pentland and T. Choudhury, "Face recognition for smart environments," Computer , vol. 33, no. 2, pp. 50–55, 2000.

[12] K. Huang and M. M. Trivedi, "Networked omnivision arrays for intelligent environment," in *Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation IV*, vol. 4479 of *Proceedings of the SPIE*, pp. 129–134, San Diego, Calif, USA, July 2001.

[13] J. Wu and M. M. Trivedi, "An integrated two-stage framework for robust head pose estimation," in *Proceedings of the 2nd International Workshop on Analysis and Modelling of Faces and Gestures (AMFG '05)*, vol. 3723 of *Lecture Notes in Computer Science*, pp. 321–335, Beijing, China, October 2005.

[14] R. Frischholz, "The Face Detection Homepage," http://www.facedetection.com/.

[15] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 586–591, Maui, Hawaii, USA, June 1991.

[16] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. I511–I518, Kauai, Hawaii, USA, December 2001.

[17] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 19, no. 7, pp. 711–720, 1997.

[18] G. Donate, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Classifying facial actions," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 21, no. 10, pp. 974–989, 1999.

[19] F. Fleuret and D. Geman, "Fast face detection with precise pose estimation," in *Proceedings of the 16th International Conference on Pattern Recognition (ICPR '02)*, vol. 1, pp. 235–238, Quebec City, Canada, August 2002.

[20] H. Schneiderman and T. Kanade, "Object detection using the statistics of parts," International Journal of Computer Vision , vol. 56, no. 3, pp. 151–177, 2004.

[21] X. Li and N. Roeder, "Face contour extraction from front-view images," Pattern Recognition , vol. 28, no. 8, pp. 1167–1179, 1995.

[22] A. Jacquin and A. Eleftheriadis, "Automatic location tracking of faces and facial features in video sequences," in *Proceedings of International Conference on Automatic Face and Gesture Recognition (AFGR '95)*, pp. 142–147, Zurich, Switzerland, June 1995.

[23] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," International Journal of Computer Vision , vol. 8, no. 2, pp. 99–111, 1992.

[24] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 232–237, Santa Barbara, Calif, USA, June 1998.

[25] S. McKenna, S. Gong, and H. Liddell, "Real-time tracking for an integrated face recognition system," in *Proceedings of the 2nd Workshop on Parallel Modelling of Neural Operators*, Faro, Portugal, November 1995.

[26] H. Graf, E. Cosatto, D. Gibbon, M. Kocheisen, and E. Petajan, "Multi-modal system for locating heads and faces," in *Proceedings of the 2nd IEEE International Conference on Automatic Face and Gesture Recognition (AFGR '96)*, pp. 88–93, Killington, Vt, USA, October 1996.

[27] J. Yang and A. Waibel, "Real-time face tracker," in *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*, pp. 142–147, Sarasota, Fla, USA, December 1996.

[28] G. Yang and T. S. Huang, "Human face detection in a complex background," Pattern Recognition , vol. 27, no. 1, pp. 53–63, 1994.

[29] K. Mikolajczyk, R. Choudhury, and C. Schmid, "Face detection in a video sequence—a temporal approach," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 96–101, Kauai, Hawaii, USA, December 2001.

[30] S. Birchfield, "An elliptical head tracker," in *Proceedings of the 31st Asilomar Conference on Signals, Systems, and Computers*, vol. 2, pp. 1710–1714, Pacific Grove, Calif, USA, November 1997.

[31] K. S. Huang and M. M. Trivedi, "Robust real-time detection, tracking, and pose estimation of faces in video streams," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04)*, vol. 3, pp. 965–968, Cambridge, UK, August 2004.

[32] R. McLaughlin, "Randomized hough transform: better ellipse detection," in *Proceedings of IEEE International Conference on Digital Signal Processing Applications (TENCON '96)*, vol. 1, pp. 409–414, Perth, Australia, November 1996.

[33] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 21, no. 5, pp. 476–480, 1999.

[34] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 20, no. 1, pp. 23–38, 1998.

[35] Y. Bar-Shalom and T. E. Fortmann, Tracking and Data Association , Academic Press, New York, NY, USA, 1988.

[36] D. Beymer, "Face recognition under varying pose," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 756–761, Seattle, Wash, USA, June 1994.

[37] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition using temporal image sequence," in *Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition (AFGR '00)*, pp. 318–323, Nara, Japan, April 1998.

[38] J. Weng, C. H. Evans, and W.-S. Hwang, "An incremental method for face recognition under continuous video stream," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition (AFGR '00)*, pp. 251–256, Grenoble, France, March 2000.

[39] B. Raytchev and H. Murase, "Unsupervised face recognition from image sequences," in *Proceedings of IEEE International Conference on Image Processing (ICIP '01)*, vol. 1, pp. 1042–1045, Thessaloniki, Greece, October 2001.

[40] G. Shakhnarovich, J. W. Fisher, and T. Darrell, "Face recognition from long-term observations," in *Proceedings of the 7th*

European Conference on Computer Vision (ECCV '02), vol. 3, pp. 851–868, Copenhagen, Denmark, June 2002.

[41] O. Arandjelović, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell, "Face recognition with image sets using manifold density divergence," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 581–588, San Diego, Calif, USA, June 2005.

[42] K. S. Huang and M. M. Trivedi, "Streaming face recognition using multicamera video arrays," in *Proceedings of the International Conference on Pattern Recognition (ICPR '02)*, vol. 4, pp. 213–216, Quebec City, Canada, August 2002.

[43] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," Proceedings of the IEEE , vol. 77, no. 2, pp. 257–286, 1989.

[44] C.-H. Lee and Q. Huo, "On adaptive decision rules and decision parameter adaptation for automatic speech recognition," Proceedings of the IEEE , vol. 88, no. 8, pp. 1241–1268, 2000.

[45] J. Ng and S. Gong, "Multi-view face detection and pose estimation using a composite support vector machine across the view sphere," in *Proceedings of the International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS '99)*, pp. 14–21, Corfu, Greece, 1999.

[46] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 19, no. 7, pp. 711–720, 1997.

[47] B.-H. Juang, "Maximum-likelihood estimation for mixture multivariate stochastic observations of markov chains," AT&T Technical Journal , vol. 64, no. 6, part 1, pp. 1235–1249, 1985.

[48] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," IEEE Transactions on Pattern Analysis and Machine Intelligence , vol. 25, no. 12, pp. 1615–1618, 2003.

[49] S. Chandrasekaran, B. S. Manjunath, Y. F. Wang, J. Winkeler, and H. Zhang, "An eigenspace update algorithm for image analysis," Graphical Models and Image Processing , vol. 59, no. 5, pp. 321–332, 1997.

[50] D. A. Fidaleo, R. E. Schumacher, and M. M. Trivedi, "Visual contextualization and activity monitoring for networked telepresence," in *Proceedings of the ACM SIGMM Workshop on Effective Telepresence (ETP '04)*, pp. 31–39, New York, NY, USA, October 2004.