Multimodal People ID for a Multimedia Meeting Browser

Jie Yang, Xiaojin Zhu, Ralph Gross, John Kominek, Yue Pan, Alex Waibel

Interactive Systems Laboratories
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{yang+, zhuxj, rgross, jkominek, ypan, waibel}@cs.cmu.edu

ABSTRACT

A meeting browser is a system that allows users to review a multimedia meeting record from a variety of indexing methods. Identification of meeting participants is essential for creating such a multimedia meeting record. Moreover, knowing who is speaking can enhance the performance of speech recognition and indexing meeting transcription. In this paper, we present an approach that identifies meeting participants by fusing multimodal inputs. We use face ID, speaker ID, color appearance ID, and sound source directional ID to identify and track meeting. After describing the different modules in detail, we will discuss a framework for combining the information sources. Integration of the multimodal people ID into the multimedia meeting browser is in its preliminary stage.

Keywords

Multimodal, multimedia, people identification, data fusion, meeting browser.

1. INTRODUCTION

Whether held one-to-one or in groups, meetings are one of the most common, important, and universally disliked events in life. In North American business alone, about 17 million meetings are held daily. Most people find it impossible to attend all relevant meetings or to retain all the salient points raised in meetings they do attend. Meeting records are intended to overcome such problems of attention and memory.

Hand recorded notes suffer several difficulties. Note-taking is time consuming, requires focus, and thus reduces one's attention to and participation in the ensuing discussions. For this reason notes tend to be fragmentary and partially summarized, leaving one unsure exactly as to what was resolved, and why.

A Multimedia meeting record is a digital recording that includes a transcript of what was said, by whom, and may potentially include additional information streams such as video. Since the multimedia production is labor intensive (e.g. when employing AV support staff), it is preferable to have the record generated automatically, even if its result lacks the smoothness quality produced professionally. At the Interactive Systems Lab of Carnegie Mellon University we are developing a multimedia meeting recorder and browser to track and summarize discussions held in a specially equipped conference room [17] [21] [13].

Tracking people is a common problem for developing an intelligent space [18] [12]. To operate effectively, it is essential that an automatic meeting browser have a solution to what we call the "assignment problem." That is, it must know who said what. It must know the names of the participants and correctly assign the appropriate name to the current speaker. This can be achieved by affixing a lapel microphone to each person, assigning the resulting waveform to that person and typing his or her name into the system before proceeding.

For practical and everyday deployment, this simple solution has several drawbacks. First, there is the annoyance of preparation: entering participant's names and wiring everyone up before the meeting begins. Naturally, people would prefer to "just walk in and talk". Secondly, unless the lapel microphones are expensive wireless devices, they restrict movement. As in a normal meeting, participants should be free to get up, move around, distribute papers, walk to the whiteboard, etc. Therefore, we arrive at two requirements for a natural and unencumbered meeting room. It must a) operate automatically, and b) leave people unrestrained. As a part of the first requirement (and in conflict with the second), the system should solve the assignment problem without manual intervention.

In a meeting application, we need to identify multiple people and track their identities throughout the entire meeting. If people are allowed complete freedom of movement, then any given identification technique will falter in some situations. For example, a face ID module will fail if a person's head is turned away from the camera. A speaker ID module will not work well when a person is far removed from the nearest microphone, or when, for example, their voice if muffled by an obstructing hand. To compensate, visual characteristics such as color distribution and patterns are helpful for "locking on" to a person once they have already been identified. Many researchers have demonstrated that color histograms work well for identifying objects [14]. Our own experience in real-time face tracking has shown that color distributions are robust and effective in real-time tracking [19]. With a name assignment determined from speech or face ID, vision-based people tracking enables a system to maintain

continuous awareness of a meeting participant, and hence the location of the acoustic source that is being fed to the speech recognition engine. This is the justification for multimodal people ID in a multimedia meeting browser.

Multimodality has been used to enhance the efficiency and the robustness of person identification algorithms [5][4]. However, most of the multimodal authentication schemes currently developed tend to only combine face ID and speaker ID together to identify an individual person in a restricted application such as the ATM machine. In this paper, we present our approach to identifying meeting participants on the basis of multimodal inputs. After a detailed description of the components color appearance ID, speaker ID and face ID we entertain a possible framework for fusing information from different sources into a unified system. This leads to a concluding discussion of some future work.

2. COLOR APPEARANCE ID

If freedom of movement is granted, occlusion and poor quality of signals are major challenges for identifying people over time. For example, people might turn their faces away from the camera, other objects might occlude the faces or the faces might be too small for a face recognition system. Similarly the speaker ID module will not work well when a person is not close enough to the nearest microphone. Color appearances of people, however, are more robust in a complex scene. In this research, we use color appearances of people as a way to identify them. The procedure for obtaining the color appearance ID is as follows:

- segment people from the background
- compute histogram for each person
- compute model probability for each person

2.1 People Segmentation

As a precursor to building color models for identification and tracking, it is necessary to first separate people from the background. Figure 1 shows 16 examples of foreground-background segmentation of people walking towards the camera.

The segmentation and extraction algorithm exists in multiple configurations, trading speed for accuracy according to the demands of real-time application. In essence, the people in Figure 1 are extracted by performing background subtraction. Our approach is comprised of four sequential stages.

- Background subtraction
- Noise removal
- Region growing
- Background update

Once the algorithm has been initialized with a sample background image, the simplest operation is to compute the difference $|\Delta R|+|\Delta G|+|\Delta B| > threshold$ between the background and current image. More accurate, but slower, is to maintain variance statistics for each pixel in the image and compute the Mahalanobis distance,

$$d(M_{I(i,j)}, x) = \frac{1}{(2\pi)^{3/2}|\Sigma|} e^{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)},$$

where $x = (r,g,b)$ is a pixel, u is the average value at the same coordinates, Σ is the covariance matrix of the point-wise color model M of the image sequence $I(x,y,t)$. Because noise is typically additive white Gaussian, most of the variance will be found in luminance rather than in hue. Consequently, transformation from RGB to YUV color space emulates principal component analysis and Σ can be approximated by a diagonal covariance matrix,

$$\Sigma \cong \begin{bmatrix} \sigma_Y & 0 & 0 \\ 0 & \sigma_U & 0 \\ 0 & 0 & \sigma_V \end{bmatrix}.$$

This can be conveniently computed to determine

$$d(M_{I(x,y)}, x) > threshold \quad {}_{3\sigma}.$$

Nonetheless, point-wise classification can yield spurious results. These results have two forms. First, there may be holes, inlets, and eroded outlines caused by the foreground object having colors similar to the background. Secondly, many small regions

Figure 1. Segmented image of various people

may arise due to noise or object fragmentation. Small-scale noise is removed by grouping foreground pixels together into a set of distinct 8-way connected regions $\{R_i\}$. Only those regions with a count above a threshold (e.g. 10-20 pixels) are considered "interesting objects."

Noise removal has a downside. It also erodes people boundaries. Background subtraction seldom yields a complete region. Instead, the image of a person is likely to be fragmented. (Notice the detached hand of person #11 in Figure 1.) To compensate, noise removal is followed by a stage of region growing. Using the pixels in each region R_i, a corresponding multi-Gaussian color model $M_{R_{i,j}} = \bigcup_j N(\sigma_j, \Sigma_j)$ is constructed in *RGB* space. Typically, a person's representation will be tri-modal – one color cluster each for his/her shirt, pants, and skin tones. Regions are grown through morphological opening, favoring vertical extension over horizontal. A prospective pixel is added to a region if it is connected and satisfies $\min d(x, M_{R_{i,j}}) < d(x, M_{I(x,y)})$. That is, if the color matches one of the Gaussian components $R_{i,j}$ better than the background pixel at that coordinate.

None of the techniques so far mentioned account for large changes in the scene. Yet, to be useful in a meeting room – running continuously for days – the system must adapt to a dynamic environment. Because it depends critically on an accurate representation of the background, it is important to keep it current. Pixels in the background image are updated (depending on their state) according to a temporal recursive filter.

$$I_{back}(x,y,t) = \begin{cases} \alpha_1 I_{back}(x,y,t-1) + (1-\alpha_1) I_{curt}(x,y,t), background \\ \alpha_2 I_{back}(x,y,t-1) + (1-\alpha_2) I_{curt}(x,y,t), I_R < threshold \\ I_{back}(x,y,t-1), I_R < threshold \end{cases}$$

The α's are decay rates that mix the current pixel value in I_{curr} with the accumulated history of I_{back}. The filter is non-linear because it operates on a region-by-region basis, depending on the current image contents. Background pixels behind regions that satisfy a "liveness" measure are left unaltered. Liveness is a combination of motion and size. It can be measured by the following equation:

$$L_{R_i} = w_1 |R_i| + w_2 \sum_{pixels} |R_{i,t} - R_{i,t-1}| .$$

This has the effect of keeping large objects alive. In contrast, small stationary objects (such as books) will gradually be absorbed into the background, thus better supporting the segmentation of people. In the current implementation, the absorption filter counts frames rather than seconds, and the frame rate varies according to scene complexity. The rate of absorption also depends on the object's size. Typical times range from 5-30 seconds. For a body part such as a motionless hand to disappear from the scene, two conditions must be satisfied. One, there must be no connecting pixels from the hand to the body, i.e. it forms a separate small blob. However the algorithm is capable of grouping disconnected blobs together into a single object based on a proximity heuristic. Thus the second condition: the horizontal gap between the hand and body must be large enough to fool the system in "believing" that the two blobs belong to different objects. If that's the case, then the hand will fade into the background.

In choosing the various parameters it is better to err on the conservative side and trade off recall for precision. Underestimating the probability of a pixel belonging to a person reduces the risk of including excessive neighboring background. This is one of main reasons why a 3D color model is preferred for segmentation; it is less likely to be confused by background pixels of similar hue but different intensity. However, once a person is extracted, the desire to track that person under varying lighting conditions favors the use of luminance normalized color spaces.

2.2 Luminance Normalized Color Space

Color is the perception of light in the visible region of the spectrum, having wavelengths ranging from 400 *nm* to 700 *nm*, incident upon the retina. Physical power (or radiance) is expressed in a spectral power distribution. A variety of spectral distributions of light can produce perceptions of colors that are indistinguishable from one another. The human retina has three different types of color photoreceptor cone cells, which respond to incident radiation with somewhat different spectral response curves. Based on the human color perceptual system, three numerical components are necessary and sufficient to describe a color, provided that appropriate spectral weighting functions are used. Theoretically, color coordinates can be defined as product integrals of the stimulus spectrum *U(n)* with three linearly independent color matching functions.

Most video cameras use an RGB color space, where a triple [*R, G, B*] represents not only color but also brightness. In order to reduce sensitivity of the color model to illumination, we use the normalized *r-g* color space and the tint-saturation color space.

The normalized r-g value is obtained by:

$$r = \frac{R}{R+G+B}, \; g = \frac{G}{R+G+B}$$

The tint-saturation space is perceptually more straightforward, where t is the tint (red, yellow, green, blue etc.) and s is the saturation [15]:

$$t = \begin{cases} 0, & r = 1/3, g \le 1/3 \\ 1/2, & r = 1/3, g > 1/3 \\ 3/4, & g = 1/3, r < 1/3 \\ 1/4, & g = 1/3, r > 1/3 \\ \frac{1}{2\pi}\arctan(\frac{g-1/3}{r-1/3}) + \frac{1}{4}, & r > 1/3, g \ne 1/3 \\ \frac{1}{2\pi}\arctan(\frac{g-1/3}{r-1/3}) + \frac{3}{4}, & r < 1/3, g \ne 1/3 \end{cases}$$

$$s = \begin{cases} 3d, & t = 0.25 \\ \frac{3d\,|\tan(t)-1|}{\sqrt{1+\tan^2(t)}}, & 0.1762 \le t < 0.5738 \\ 3d\cos(t-0.75), & 0.5738 \le t \le 0.875 \\ 3d\cos t, & \text{otherwise} \end{cases}$$

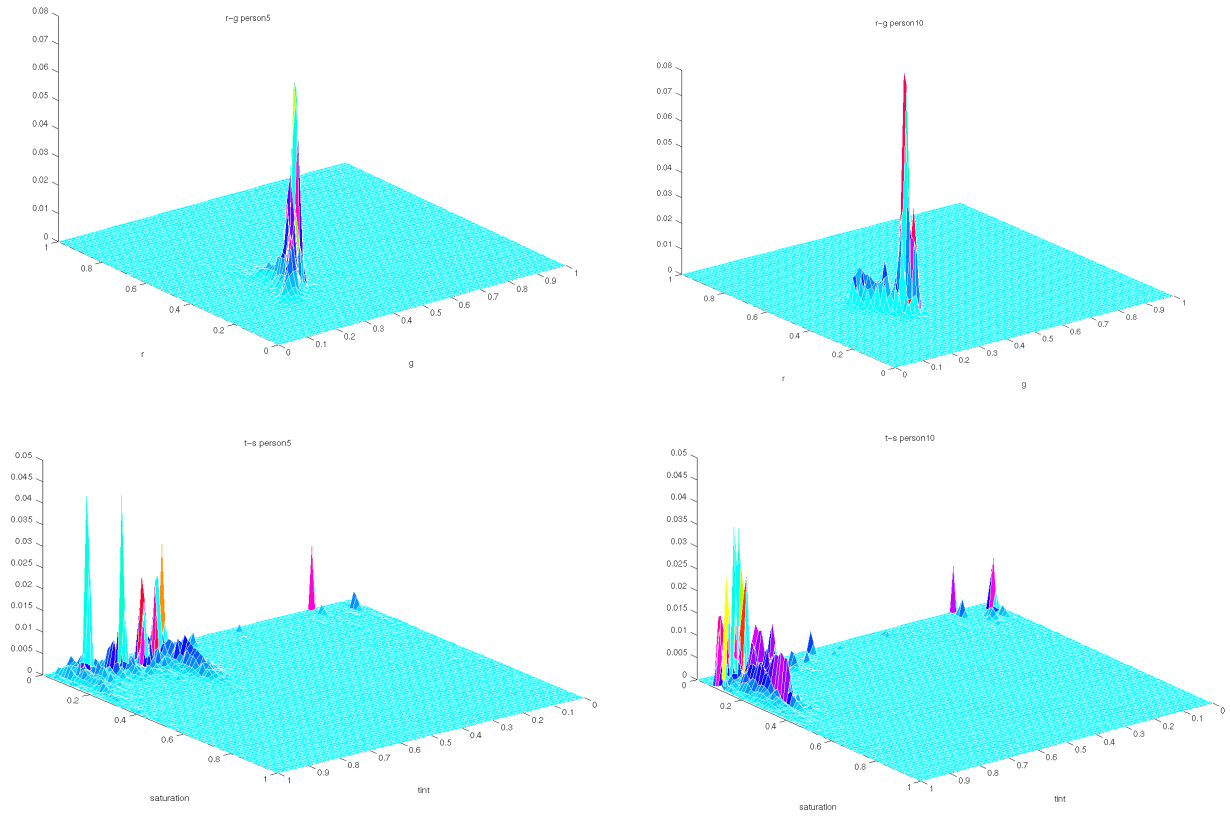

Figure 2. Person 5 and Person 10's color histograms in *r-g* and *t-s* space

where

$$d = \sqrt{(r-1/3)^2 + (g-1/3)^2}$$

The *t* and *s* values are normalized within [*0, 1*].

A color histogram is a distribution of colors in the color space and has long been used by the computer vision community in image understanding. For example, analysis of color histograms has been a key tool in applying physics-based models to computer vision. It has been shown that color histograms are stable object representations largely unaffected by occlusion and changes in view, and that they can be used to differentiate among a large number of objects [14]. It has since been shown that the colors do not fall randomly in a plane, but form clusters at specific points. We have successfully developed an adaptive skin color model for tracking human faces in real time [19].

For each segmented person, we collect (*r, g*) or (*t, s*) histogram counts with an M-bin histogram. Based on the counts, we compute a smoothed probability distribution function (p.d.f.). For a *t-s* space, the p.d.f. is computed as follows:

$$P(t,s) = \begin{cases} \dfrac{count(t,s) - \varepsilon}{\sum_{t,s} count(t,s)}, & \text{count(t,s)} > 0 \\ \dfrac{n \cdot \varepsilon}{(M-n)\sum_{t,s} count(t,s)}, & \text{count(t,s)} = 0 \end{cases}$$

The same formula can be used for *r-g* space by substituting *r* and *g* to *t* and *s*. An absolution discounting smoothing scheme is used here to avoid zero probability in the p.d.f., which is require for the p.d.f. to be a generative model. ε is a small discounting value, e.g., $\varepsilon = 0.001$. *n* is the number of non-empty bins and *M* is the total number of bins. Intuitively we discount histogram bins with non-zero counts by ε, and evenly distribute the discounted probability mass to bins with zero count.

Figure 2 shows color histogram p.d.f.'s of person 5 and person 10 in Figure 1. Both r-g color space histograms and t-s color space histograms are shown. Note how the blue shirt and red pants are represented.

To determine which color space has better classification capability, we performed a test on 5000 images of the 16 people in Figure 1. During training, we generate color histogram p.d.f models *PMi(r, g)* and *PMi(t, s)* for each person *i*. During recognition, an input image's color histogram p.d.f *Po(r, g)* and *Po(t, s)* are compared with the corresponding model histograms in terms of the Kullback-Leibler divergence [6]:

$$D(M_i \,||\, o) = \sum_{x \in S} P_{Mi}(x) \log \frac{P_{Mi}(x)}{P_o(x)}$$

where S stands for all histogram bins in *r-g* or *t-s* color space.

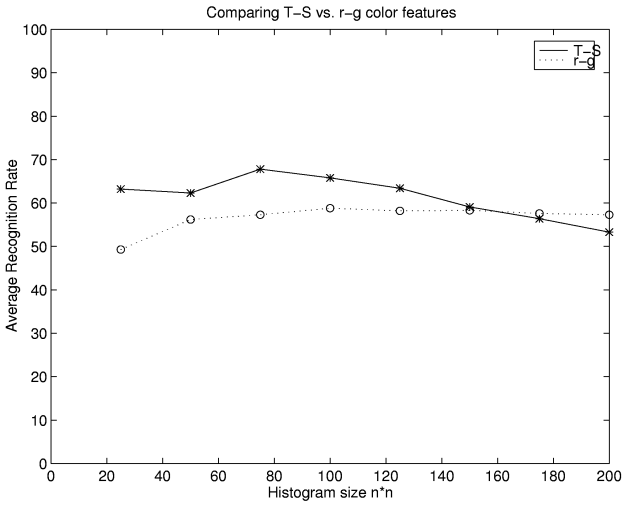

Figure 3. Comparison of r-g and t-s color space

The person whose model has the smallest divergence is regarded as the recognition result. Figure 3 shows the result of comparing *r-g* and *t-s* color spaces with different histogram sizes. Both results are largely insensitive to histogram size. The *t-s* color space has higher recognition rate in general. Therefore, we use t-s space for color appearance.

2.3 A Probabilistic Color Model

In a multimodal fusion framework, we would like to get the probability *P(D | Mi)* from the color appearance ID module, where D is an input image. This probability will be combined later with other modalities.

If we assume the color histogram p.d.f.'s to be Generative Models, i.e. each pixel of the input image is independently sampled from the p.d.f., then the probability of an input image *D* being generated by the model is

$$P(D|M) = \prod_{x \in D} P_{Mi}(x) = \prod_{x \in S} P_{Mi}(x)^{CD(x)},$$

where *CD(x)* is the number of x in the image *D*.

Therefore

$$\log P(D \mid Mi) = \sum_{x \in S} CD(x) \log PMi(x)$$

$$= \frac{1}{N} \sum_{x \in S} PD(x) \log PMi(x) = -\frac{1}{N} H(PD, PMi)$$

That is, the *logProb* is proportional to negative Cross-Entropy of the image and the model, where *N* is the size (number of pixels) of the image *D*.

Nevertheless this particular generative model might not be the optimal one, as revealed by the experiment. In the experiment with 16 people, each person has roughly the same number of images, i.e. the priors *P(Mi)* are equal. Thus the optimal Bayesian classification is equivalent to choosing a model *Mi* for an image *D* such that *P(D|Mi)* is at maximum.

$$M^* = \arg\max_{Mi} P(D \mid Mi) = \arg\max_{Mi} \log P(D \mid Mi)$$

$$= \arg\max_{Mi} -\frac{1}{N} H(P_D, P_{Mi})$$

$$= \arg\min_{Mi} H(P_D, P_{Mi})$$

$$= \arg\min_{Mi} H(P_D, P_{Mi}) - H(P_D)$$

$$= \arg\min_{Mi} D(P_D \,||\, P_{Mi})$$

This indicates that if we use the generative model, the best model is the one such that the Kullback-Leibler distance between the image p.d.f and the model p.d.f is minimized.

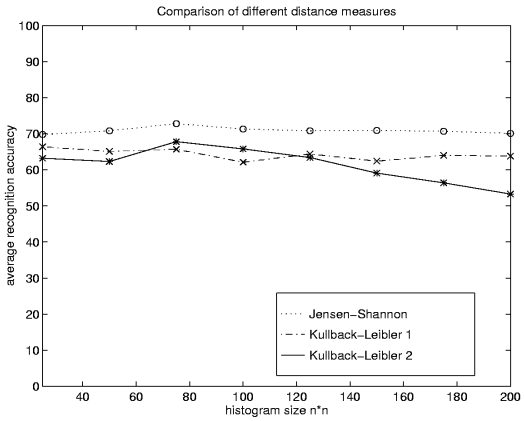

Figure 4. Comparison of three distance measures

Figure 4 shows the recognition accuracy of this generative model (the dashed line, "Kullback-Leibler 1") together with a similar measure (the solid line, "Kullback-Leibler 2") where

$$M^* = \arg\min_{Mi} D(P_{Mi} \,||\, P_D)$$

and the Jensen-Shannon divergence measure [7] (the dotted line), where

$$M^* = \arg\min_{Mi} \left[D(P_{Mi} \,||\, \frac{P_{Mi} + P_D}{2}) + D(P_D \,||\, \frac{P_{Mi} + P_D}{2}) \right]$$

The Jensen-Shannon divergence can be computed with only the non-zero part of both p.d.f.'s. This eliminates the need for probability smoothing. Since many bins have zero count, as shown in Figure 2, using Jensen-Shannon divergence may reduce the side-effect brought forth by smoothing. In Figure 4, the Jensen-Shannon divergence looks better. How to incorporate it into the probabilistic model is still an open question. For the time being we are using the generative model described above

3. SPEAKER IDENTIFICATION

We have considered speaker ID in a meeting room. In our current setup, both audio and video signals are available. The identity of everyone in the room is known. The problem is to identify speaker at any given time. This is a text-independent close-set speaker identification task. Both convolution and additive noise can be considered consistent, except for occasional events such as phone

ringing and door clapping. The limited training set and test set collected in the same noise environment [1] have been used to evaluate the system. Our experiments showed that if training and testing are performed in the same noise conditions, the performance is comparable to the performance achieved on clean speech. A difficulty in this task is how to achieve high performance in real-time with a relatively small amount of training data. We will describe our baseline identification system first. In order to improve performance, we need to combine acoustic information with vision approaches. There will also be a mapping method between acoustic source and person position. The meeting room includes a microphone array or several fixed microphones. Using several microphones will also provide better speech quality for both speaker recognition and optional automatic speech transcription.

3.1 Segmentation

Input speech first goes through a segmentation stage. The module roughly detects a possible acoustic event (utterance) and splits the continuous audio data into shorter segments. We use a simple approach, based on the energy and zero-crossing rate. A finite state machine is built, and threshold and elapsing time are used for state transition. Situations in which the utterance of one speaker is split into several parts or two speakers are merged into a segment, can be handled by the model.

3.2 Modeling

The speech spectrum reflects a person's vocal tract structure and is used both in speech recognition and speaker identification. We use Mel Frequency Coefficients (MFCs) as feature vectors by applying Mel-scaled filter banks on the FFT spectrum [9]. The sampling rate of the speech signal is 16KHz with high-pass pre-emphasis. Frame size is 32 ms with a frame shift of 16 ms. The training is done offline. A few utterances of each speaker (roughly 30 seconds) at the beginning of a meeting are used to build a Gaussian Mixture Model (GMM)[10].

$$P(\vec{x} \mid \lambda_k) = \sum_{i=1}^{M} p_i \frac{1}{(2\pi)^{D/2} \mid \Sigma_i \mid^{1/2}} e^{-\frac{1}{2}(\vec{x}-\vec{u}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{u}_i)}$$

where x is the D-dimensional MFC vector, pi are the mixture weights of the M Gaussian densities where Σ pi = 1. The speaker k is represented by model λk.

The parameters of model λk were estimated by using the expectation-maximization (EM) algorithm. We randomly select M vectors from a speaker's training set and use them as starting means as suggested by [10]. An identity matrix is chosen as starting covariance matrix and our experiment shows that this is sufficient for both iteration convergence and speaker identification performance. Based on evaluations of the system using between 8 and 32 Gaussians we choose 16 since this configuration achieved the best performance. Table 1 shows the performance of the system for 30 speakers.

Correct rates	Test length	
Recording	3 sec.	6 sec.
Clear	97.8%	100.00%
Noisy	96.6%	100.00%

Table 1. Identification Performance on 30 Speakers

The input speech flow is segmented using silence detection. For each segment, we use a fully connected HMM. Each speaker is a state of the HMM. The transition probability is fixed by uniform distribution. To achieve real-time performance, we can assume only one speaker in each segment. This also means that the delay to output the speaker ID is short. The speech vectors are inputted frame by frame and for each frame, a decision is made based on the probability at each stat. In the beginning of an utterance, the decision is more liable to be wrong, since there is not enough speech for the classifier. This shortcoming could be corrected somewhat through the vision approach described next.

3.3 Microphone Array and Source Position

In order to get accurate positions of sound sources, a microphone array is required. In our initial experiment only two microphones are used. We obtained two channels of speech whose energy pair (e_1, e_2) is used to give a coarse estimation of the position. This information is sent to the fusion module for final decision. The feedback position information is useful for distance-adaptation in acoustic signal, e.g. spectral variability compensation, though not yet implemented. We estimated *P(e|A)* of different positions by GMMs as in the section above, but the model order M is much smaller, namely 2 - 4. The energy feature is liable to the influence of environment noise and it is incapable of distinguishing some symmetric positions. It is only a initial work for mapping between acoustic source and visual source. More work needs to be done in the future.

4. FACE ID

While people identification based on color appearance works reasonably well in most situations, it fails when meeting participants are dressed similarly. To overcome this problem we introduced face identification into the system. Just as for all components described so far, the meeting room scenario proves to be a challenging task for a face recognition module. A typical setting includes multiple faces in multiple poses at different sizes in a complex environment. As the subjects communicate with each other we can also expect a full range of facial expressions to be visible.

4.1 Locating and Tracking Faces

Locating and tracking human faces is a prerequisite for face recognition. Facial features, such as the eyes, nose, and mouth, are natural candidates for locating human faces. These features, however, may change from time to time. Occlusion and non-rigidity are basic problems with these features. Four basic techniques are commonly used for dealing with feature variations: correlation templates, deformable templates, spatial image invariants, and neural networks. These methods are computationally expensive and hardly achieve real-time performance. A different approach for locating and tracking faces is to use skin-colors. We have developed an adaptive skin-color model [20]. By combining the adaptive skin color model with the motion model and the camera model, we have developed a real-time face tracker [19]. The system has achieved a rate of 30+ frames/second on both Unix and PC platforms. The system can track a person's face while the person walks, jumps, sits and rises. The face tracker can catch the faces of participants in real-time for face recognition.

4.2 Face Recognition

Face recognition has been actively studied in the computer vision community [3]. The research effort has been towards recognizing frontal faces with limited variance in illumination and facial expression. A basic approach is to develop some powerful low dimensional representations for faces. The techniques based on Principal Components Analysis (PCA), namely "eigenfaces" [11] [16], have demonstrated excellent performance. While the eigenfaces are useful for extracting information from a facial image, it has some disadvantages. The basic problem with the eigenface is that it is a global and linear approach. This causes problems in robustness and generality. Research has showed that eigenfaces are sensitive to variability due to expression, pose, and lighting condition.

A view-based modular eigenspace has been proposed to incorporate salient facial features such as the eyes, nose and mouth, in an eigenfeature layer [8]. Higher recognition rates have been reported for this representation. The problem with this view-based approach is the lack of an optimal way to fuse the information. We propose to use a dynamic space warping (DSW) method to solve the problem. The idea is to compare a set of points instead of one point in the eigen-space. The DSW finds the closest match between two sets of eigen-points if they were indeed the same face. This is achieved by distorting the positions of the eigen-points of the unknown face to match the template.

In the eigenface approach, a face image defines a point in a high dimensional space. Different face images share a number of similarities with each other, so that the points representing these images are not randomly distributed in the image space. They all fall into a lower dimensional subspace. The key idea of the recognition process is to map the face images into an appropriately chosen subspace and perform classification by distance computation. If we restrict ourselves to a linear dimensionality reduction, the optimal solution is provided by the principal component analysis, also called Karhunen-Loeve transformation. The basis of the lower dimensional eigenspace is formed by the eigenvectors of the covariance matrix of the set of training images corresponding to the largest eigenvalues. Instead of transforming a face image into one point in the eigenspace, we break down a face image into sub-images using a moving window. When the square window covers the whole image by moving a half of window size each time, we get a sequence of sub-images. Each sub-image can be transformed to a point in the eigen-space. We then get a set of eigen-points for each face image. During the recognition process, the template set of points is compared to the unknown set of points. The procedure is similar to the dynamic time warping (DTW) in speech recognition [9].

# of people	DSW	Eigenface
14 (with background)	100%	78.5%
40 (without background)	97.5%	87.5%

Table 2. Face recognition using DSW

We have tested the proposed approach on a limited database. The initial results are encouraging. Table 2 shows results from two different test sets. The first set of data is smaller but with some background. The second set of data contains only face images. The results indicate that the proposed approach is much better and more robust than the original eigenface method, especially when face segmentation is not perfect. We currently perform an evaluation on a larger database and will integrate face ID into the people ID system.

5. MULTIMODAL INPUT FUSION

5.1 Framework

Assume that the observation from multimodal inputs is a triple (Φ, Ω, θ), where Φ is an input image, Ω is an utterance and θ is the direction from which the utterance is detected. There are N people $p1,\ldots, pN$. A configuration ($Ap1, \ldots, ApN, S$) is a description of the scene about who is where and who is speaking, with Api being the area in Φ that is occupied by person pi, and S being the person who is currently speaking. The goal of fusion is to find a configuration that could best interpret the observation. Formally, the optimal configuration is defined as the configuration with the highest posterior probability given a multimodal observation:

$$\begin{aligned} & A_{p1}^{*}, A_{p2}^{*}, \ldots, A_{pN}^{*}, S^{*} \\ &= \arg\max_{Ap1 \ldots ApN, S} p(A_{p1}, \ldots, A_{pN}, S \mid \Phi, \Omega, \theta) \\ &= \arg\max_{Ap1 \ldots ApN, S} \frac{p(\Phi, \Omega, \theta \mid A_{p1}, \ldots, A_{pN}, S)\, p(A_{p1}, \ldots, A_{pN}, S)}{p(\Phi, \Omega, \theta)} \end{aligned}$$

The term $p(\Phi,\Omega,\theta)$ doesn't change with different configurations and can be omitted. Then

$$A_{p1}^{*}, A_{p2}^{*}, \ldots, A_{pN}^{*}, S = \arg\max_{Ap1 \ldots ApN, S} p(\Phi, \Omega, \theta \mid A_{p1}, \ldots, A_{pN}, S)\, p(A_{p1}, \ldots, A_{pN}, S)$$

The term $p(\Phi,\Omega,\theta \mid Ap1, \ldots, ApN, S)$ can be further decomposed by repetitively applying conditional independence assumptions,

$$\begin{aligned} & p(\Phi, \Omega, \theta \mid A_{p1}, \ldots, A_{pN}, S) \\ &= p(\Phi \mid A_{p1}, \ldots, A_{pN})\, p(\Omega, \theta \mid A_s, S) \\ &= \left(\prod_{i=1}^{N} p(\Phi_{Api} \mid p_i) \right) p(\Omega \mid S)\, p(\theta \mid A_s) \end{aligned}$$

where ΦApi stands for the sub-image in area Api. The various terms above can be computed with individual multimodal models. $p(\Phi Api \mid pi)$ is the visual model for person pi, which gives the probability of a sub-image generated by the person. We can use an interpolation of a face identification model and a color identification model to compute this probability:

$$p(\Phi_{Api} \mid p_i) = \lambda p_{face}(\Phi_{Api} \mid p_i) + (1 - \lambda)\, p_{color}(\Phi_{Api} \mid p_i)$$

where λ is the interpolation weight. λ may depend on the confidence level of the face identification model and the confidence level of the color identification model. λ can be trained with the EM algorithm. $p(\Omega|S)$ gives the probability of an utterance being generated by speaker S. We use a speaker identification model for this probability. $p(\theta|As)$ is the utterance direction model. Given the actual position (implied from the area As in Φ) of the speaker, it computes the probability that the utterance is detected as from direction θ. This model depends on

the settings of both the cameras and microphones, and is a bridge between video and audio inputs.

The term *p(Ap1, …, ApN, S)* describes the a priori distribution of configurations. It reflects knowledge about where people are likely to be in the scene, and who would talk more. For example, *p(Ap1, …, ApN, S)* might have a high probability when *Ap1, …, ApN* correspond to the position of seats and *S* corresponds to the speaker if we know them in advance. Furthermore, the probability can be conditioned on previous optimal (recognized) configurations over time,

$$p(A_{p1},...,A_{pN},S)^t = p(A_{p1},...,A_{pN},S \mid (A_{p1}^*,...,A_{pN}^*,S^*)^{t-1},(A_{p1}^*,...,A_{pN}^*,S^*)^{t-2},...)$$

In this way, we can incorporate motion estimation, utterance duration and speech turn-taking prediction into the people ID system. Nevertheless, the model has a severe sparse training data problem similar to language models' in speech recognition. We need to make certain independence assumption, parameter tying and proper smoothing to train this model. Alternatively an adaptive model, starting from uniform distribution and adjusting itself to new observations, may be desirable.

In searching for the optimal configuration *(Ap1*, …, ApN*, S*)*, one would have to perform *argmax* over the entire configuration space, i.e. assuming any part of the image may contain a person. Even if we make the simplification that each person is bounded in the image by a rectangle of fixed size, there are still $O(S^N \cdot N)$ combinations to search, where *S* is the size of the image, typically several hundred; *N* is the number of people. It is computationally very expensive. Instead we would only search through a sub-space of the configuration space to reduce computational cost. The sub-space is introduced by the people tracker module, which attempts to segment people from background image. The output of the people tracker module is a set of area *a1,…,ak* occupied by people in an image *Φ*. The sub-space is then defined as

$$\{A_{p1},...,A_{pN},S \mid A_{pi} \in \{a_1,...,a_k,\phi\}, \forall 1 \le i \le N \wedge S \in \{p_1,...,p_N,\phi\}\}$$

That is, we assume people appear and only appear within these areas, one area for each person. Hence we only need to assign people to the areas, and the computational complexity reduces to $O(\frac{N!}{(N-k)!} \cdot N)$. Although sub-space may lead to sub-optimal configuration result, we hope the sub-optimal configuration is reasonably close to the optimal one.

5.2 Experiment

To demonstrate the feasibility of the framework, we set up a simple meeting as shown in Figure 5. Three participants sit around a table. Two video cameras (not shown in the figure) each take part of the scene, and the images are merged to create a wide-angle input video image. Two microphones record the conversation. The microphones can also provide rough directional information of each utterance by the difference in input energies. The wide-angle video image *Φ*, the conversation *Ω* and the microphone energy difference *ΔE* are the input observations to the fusion module. In this experiment, we try to combine the color appearance ID and speaker ID.

Figure 5. A simple meeting setting

Segmentation is performed on the image *Φ* to find three areas *a1, a2, a3* that contain the participants. We are interested in finding who is in which area, and who is the current speaker. Formally, there are 3 people *p1, p2, p3*. A configuration *(Ap1, …, Ap3, S)* is a description of the scene about who is where and who is speaking, with *Api ∈{ a1, a2, a3}* being the area in *Φ* that is occupied by person pi, and *S ∈{ p1, p2, p3}* being the person who is currently speaking. We further assume in this simple setting, the configuration priors are uniform and therefore:

$$\begin{aligned} & A_{p1}^*, A_{p2}^*, A_{p3}^*, S^* \\ &= \arg\max_{Ap1,Ap2,Ap3,S} p(\Phi,\Omega,\Delta E \mid A_{p1},A_{p2},A_{p3},S) \\ &= \arg\max_{Ap1,Ap2,Ap3,S} \prod_{i=1}^{3} p(\Phi_{Api} \mid p_i) p(\Omega \mid S) p(\Delta E \mid A_S) \end{aligned}$$

where *ΦApi* stands for the sub-image in area *Api*, and As is the area occupied by the current speaker. The various terms above can be computed with the modules discussed previously. In this experiment, we take *p(ΦApi | pi)* from the color appearance model; *p(Ω | S)* from the speaker identification model. *p(ΔE | As)* is the utterance direction model: Given the actual position (implied from the position of area *As* in *Φ*) of the speaker, it computes the probability that the utterance is recorded by the two microphones with energy difference ΔE. The *argmax* searching is performed on all possible combinations of *Api ∈{ a1, a2, a3}* and *S ∈{ p1, p2, p3}*.

In our preliminary experiment, we collected one meeting session with 3 people. The meeting lasted 220 seconds, with 2990 audio and video inputs. For both inputs, we find the optimal configuration with information fusion. We also compute the optimal configuration without fusion, i.e. using the models individually:

$$A_{p1}^*, A_{p2}^*, A_{p3}^* = \arg\max_{A_{p1}...A_{p3}} \prod_{i=1}^{3} p(\Phi_{A_{pi}} \mid p_i)$$

$$S^* = \arg\max_{S} p(\Omega \mid S)$$

The result is given in Table 3. We consider a configuration to be erroneous if any of the components *Ap1, Ap2, Ap3, S* is wrong. In this experiment, the configuration error rate drops by 2% absolute after information fusion. Therefore it looks promising to apply fusion to people identification. We plan to perform more experiments and detailed error analysis.

	Number of Configuration errors	Error rate
Without fusion	374	12.51%
With fusion	319	10.67%

Table 3. Configuration errors without/with information fusion

The whole system runs at 2 frame per second. This is sufficient for our meeting application.

6. CONCLUSION

We have presented an approach to identifying people using multimodal input. The people ID is essential for developing a multimedia meeting record. The challenge of the problem is to continuously identify multiple people in a dynamic environment. We have developed systems of color appearance ID, speaker ID and face ID. We have also introduced a framework for combining results from different modalities. We will improve the current systems and integrate them into a multimedia meeting browser. The objective of this project is to develop a system that can transcribe and summarize a meeting from both audio and video inputs.

7. ACKNOWLEDGMENTS

We would like to thank Michael Bett, Hua Yu, Robert Malkin, Rainer Stiefelhagen, Uwe Meier, and Weiyi Yang for their support to this project. We would like to thank Frances Ning for proofreading manuscript of this paper. We also would like to thank our colleagues in the Interactive Systems Lab for participating in data collection and experiments. This research was partially supported by the Defense Advanced Research Projects Agency under the Genoa project, subcontracted through the ISX Corporation under contract No. P097047. The second author was also supported in part by National Science Foundation under contact No. 9720374.

8. REFERENCES

[1] F.Bimbot, H. Hutter, C. Jaboulet, J. Koolwaaij, J. Lindberg, and J. Pierrot. Speaker verification in the telephone network: Research activities in the cave project. Technical report, PTT telecom, ENST, IDIAP, KTH, KUN, and Ubilab, 1997

[2] C.M.Bishop, Neural Networks for Pattern Recognition, Oxford Press, 1995

[3] R. Chellappa, C.L. Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. Proceedings of the IEEE, 83(5), pages 705-41, 1995.

[4] T. Choudhury, B. Clarkson, T. Jebara and A.Pentland. Multimodal person recognition using unconstrained audio and video. In Proceedings of AVBPA'99

[5] B. Duc, E.S. Bigun, J. Bigun, G. Maire, and S. Fischer. Fusion of audio and video information for multi-modal person authentication. Pattern Recognition Letters, 18 (9), pages 835-843, 1998.

[6] S. Kullback, Information theory and statistics, New York: John Wiley and Sons, 1959

[7] J. Lin. Divergence measures based on the Shannon Entropy. IEEE Transactions on Information Theory, 37(1), 145-151

[8] A.Pentland, B.Moghaddam, and T.Starner. View-based and modular eigenspaces for face recognition, Proc. CVPR'94, pp.84-91, June 1994

[9] L.R. Rabiner and B-H. Juang. Fundamentals of speech recognition. Englewood Cliffs, N.J. : PTR Prentice Hall, 1993.

[10] D. A. Reynolds and R. C. Rose, Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models, IEEE Trans. Speech. Audio Processing , vol.3, pp.72-83, Jan, 1995

[11] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. Journal of Opt. Soc. Am., 4(3), pages 519-524, 1987.

[12] S. Shafer, J. Krumm, B. Brumitt, B. Meyers, M. Czerwinski, D. Robbins. The new EasyLiving project at Microsoft Research. Joint DARPA/NIST Smart Spaces Workshop, July 30-31, 1998.

[13] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel. From gaze to focus of attention. In Proceedings of the Workshop on Perceptual User Interfaces (PUI98).

[14] M. J. Swain and D. H. Ballard. Color indexing. International Journal of Computer Vision, 7(1) pages 11-32, 1991.

[15] J. C. Terrillon, M. David, S. Akamatsu. Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments. In Proceedings of the Third international conference on automatic face and gesture recognition. Nara Japan, 112-117, 1998.

[16] M.A. Turk and A. Pentland. Face recognition using eigenfaces. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pages 586-591, 1991.

[17] A. Waibel, M. Bett, M. Finke, and R. Stiefelhagen. Meeting browser: Tracking and summarizing meetings. In Proceedings of the DARPA Broadcast News Workshop 1998.

[18] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland. Pfinder: real-Time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), pp. 780-785,1997.

[19] J. Yang and A. Waibel. A Real-time face tracker. In Proceedings of WACV'96, pages 142-147, 1996.

[20] J. Yang, R. Stiefelhagen, U. Meier, and A. Waibel. Visual tracking for multimodal human computer interaction. In *Proceedings of CHI 98* , pp. 140-147

[21] H. Yu, C. Clark, R. Malkin, and A. Waibel. Experiments in automatic meeting transcription using JRTK. In Proceedings of ICASSP'98.