Journal on Multimodal Interfaces manuscript No.
(will be inserted by the editor)

Multimodal Identification and Localization of Users in a Smart Environment

First Author · Second Author

Received: date / Accepted: date

Abstract Detecting the location and identity of users is a first step in creating context-aware applications for technologically-endowed environments. We propose a system that makes use of motion detection, person tracking, face identification, feature-based identification, audio-based localization, and audio-based identification modules, fusing information with particle filters to obtain robust localization and identification. The data streams are processed with the help of the generic client-server middleware SmartFlow, resulting in a flexible architecture that runs across different platforms.

Keywords Multimodal tracking, multimodal identification, particle filters, smart rooms

1 Introduction

The spatio-temporal localization and recognition of people through various sensors poses problems of great theoretical and practical interest, in particular for home environments and smart rooms. In these scenarios, context-awareness is based on technologies like gesture and motion segmentation, unsupervised learning of human actions, determination of the focus of attention or intelligent allocation of computational resources to different modalities.

In this work we aim at putting together different algorithms for detection, tracking and identification, working in a completely automatic way. Through the observation and subsequent processing of the data captured using a large number of sensors from multiple modalities, we try to determine the identity and the spatial positions of people in the room. As noted in [51], the full implications of real-time human tracking only become concrete within the context of applications. Obtaining this knowledge is the first step in developing more elaborate smart applications like meeting managers that track the speaker, gesture-based interfaces that require the identity of the person performing a certain gesture, systems

F. Author
first address
Tel.: +123-45-678910
Fax: +123-45-678910
E-mail: fauthor@example.com

S. Author
second address

that provide users with customized information (as in MIT's Oxygen project), or automatic activity summarization systems.

We have collected two recordings and fully annotated them for experimental evaluation. The implemented system contains some modules that are not novel in themselves, but are adapted to the experimental setting. These modules are also included in the exposition for completeness' sake. A robust face recognition method is employed to identify persons initially as they enter the smart room. For visual tracking, a recently proposed method based on probabilistic occupancy maps is extended and adapted to the experimental setting. For visual recognition, a novel feature-based identification method is proposed that alleviates the shortcomings of contemporary Bayesian approaches for dealing with models constructed on-the-fly. For audio-based localization and identification, we use recently developed state-of-the-art methods. Initially, the performance of each modality is separately measured. Then several methods for combining different modalities are proposed and assessed, including a particle filter based tracker that incorporates identity into tracking for robust operation.

This paper is organized as follows. In Section 2 we briefly describe our experimental setting including the sensors and the collected data. The setup justifies our methodological choices further on. In Section 3 we detail the methods we have used for processing of visual sensor information, followed by a similar exposition in Section 4 for audio sensors. Literature surveys of related methodologies are delegated to the subsections, as we have numerous issues to deal with, spanning a large area of research. The particle filter-based tracking approach and our experimental results with multimodal information are reported in Section 5, followed by conclusions and future research directions in Section 6.

2 Experimental Setup

2.1 The UPC Smart Room

In this study, audiovisual recordings of interactive small working-group seminars have been used. These recordings were collected at the UPC smart room, in accordance with the "CHIL Room Setup" specifications [13]. Recordings were performed at different dates (several months apart) to ensure proper variability (face, hair, etc.) of the participants. In the recordings, four people enter an empty room, one by one. Once inside, they move around a central table, always in standing position, talk to each other, walk around the room from time to time, and finally leave the room one by one. The length of each recording is approximately five minutes. For some algorithms a relatively large amount of training data needs to be available. One of the recordings was intended for training in those cases, and the second one is used for testing without any further change or parameter adjustment. The second recording is more difficult in terms of tracking, as all subjects wear clothes of similar colours. For additional training we have used a set of similar recordings from the CLEAR evaluation campaign [33].

Fig. 1 depicts a brief description of the UPC smart-room sensors and space conditions. The room has an entrance door, a big window and a table in the middle. The window was closed during recordings to avoid illumination changes. However, small illumination changes can occur when the door is opened.

The room is monitored using six cameras: four fixed cameras at the corners of the room (labeled Cam1 to Cam4 in Fig. 1), one zenithal fish-eye camera at the ceiling (Cam5) and one active camera (PTZ) aimed and zoomed at the entry door to capture the faces of the incoming

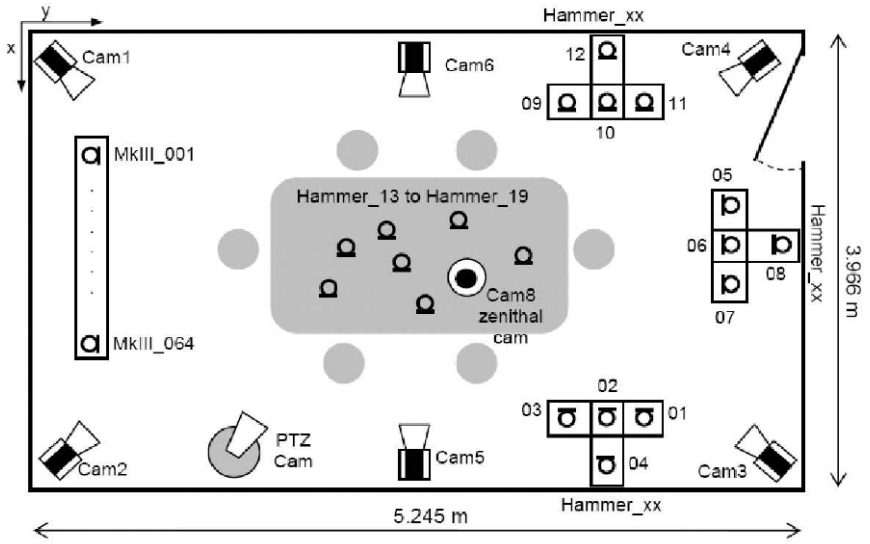


Fig. 1 *The UPC smart room setup.*

people at high resolution. Video is interlaced, recorded in compressed JPEG format, at 25 fps and 768×576 resolution.

The audio sensor setup is composed by one NIST Mark III 64-channel microphone array, three T-shaped four-channel microphone clusters and eight tabletop microphones. Audio is recorded in separate channels in *wav* format, at 44.100 sampling frequency.

Far-field conditions have been used for both audio and video modalities. All data flows are timestamped and the computers used to record the signals are synchronized using the network time protocol (NTP). This makes possible to synchronize audio and video data. There is no manual segmentation of the data. Each technology is supposed to automatically segment the recorded signal. Fig. 2 shows a sample set of recordings from the room setup.

2.2 The Middleware

The problem of interconnecting several algorithms that work on data streams coming from a high number of sensors from different modalities is far from trivial. Synchronization of the different data flows, distributed computing and the interconnection of the algorithms are issues that need to be addressed.

To allow efficient communication of sensor data and distributed computation, it is useful to have a middleware that provides infrastructure services. We propose to use the NIST SmartFlow system that allows the transportation of large amounts of data from sensors to recognition algorithms running on distributed, networked nodes [2, 3]. The working installations of SmartFlow is reportedly able to support hundreds of sensors [42]. In the present version of our system, the integration was not completed, as some modules are implemented with MATLAB, and data exchange of modules was simulated. However, the architecture is set up in a modular fashion to allow complete implementation under SmartFlow.

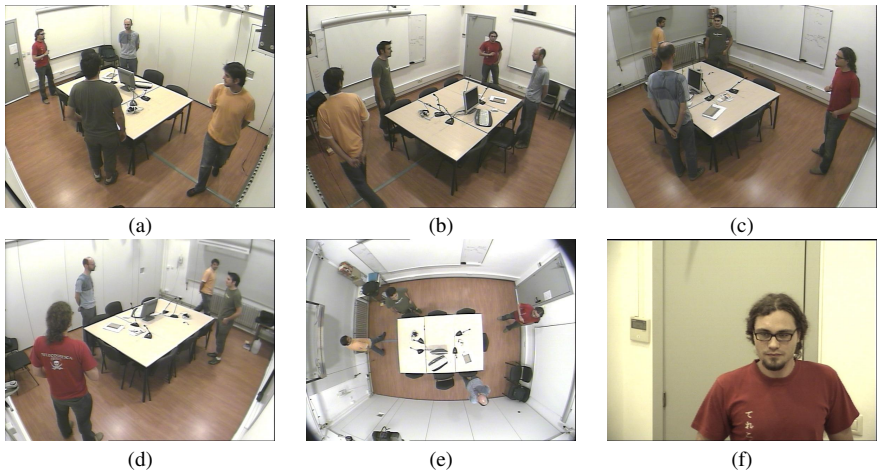


Fig. 2 *Sample camera recordings from the first session. (a)-(d) Four corner ceiling cameras. (e) Center ceiling camera. (f) Door camera.*

Smartflow offers a great deal of data encapsulation for the processing blocks, which are called "clients". Each client can output one or more flows for the benefit of other clients. The communication over TCP/IP sockets is transparent to the user, and handled by the middleware. The design of a working system is realized through a graphical user interface, where clients are depicted as blocks and flows as connections. The user can drag and drop client blocks onto a map, connect the clients via flows, and activate these processing blocks.

The synchronization of the clients is achieved by synchronizing the time for each driving computer, and timestamping the flows. The network time protocol (NTP) is used to synchronize the clients with the server time, and this functionality is provided by SmartFlow. A separate client is used to start the processing clients simultaneously. The video streams are not completely in one-to-one correspondence, as clients sometimes drop frames.

There are several drawbacks of SmartFlow. Handling multiple flows is difficult, and clients cannot selectively subscribe to parts of a data flow. An alternative middleware called *ChilFlow* that alleviates some of these issues is currently in development under the CHIL project [1], but not made available yet [45].

2.3 The Information Flow of the System

The main contribution of the paper is an intuitive way of connecting different tracking and recognition methods to perform multimodal tracking and identification in the smart environment. Fig. 3 depicts the information flow within the system: When people enter the room, the face detection module detects the face on the PTZ camera, and marks the face area, which is then identified by the face identification module. This provides a reliable identification, which is used to trigger the feature-based identification (FBI) module. The FBI receives moving blobs that are detected and tracked by the tracking module, and builds a feature model for the identified person on-the-fly. The FBI module is thus responsible for the continuity of tracked frames, and serves as a weak biometrics system that can identify users in cases where tracking fails, or stronger biometric information is not available. Motion detec-

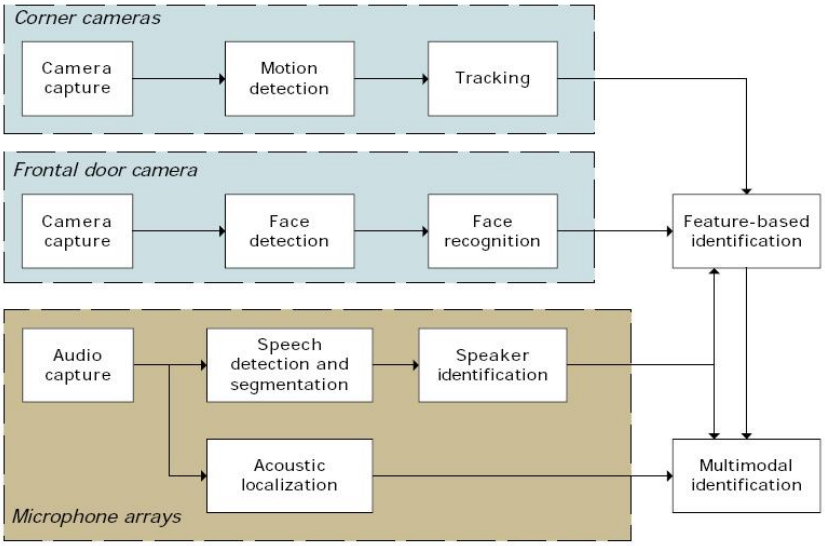


Fig. 3 The flow of information within the smart room architecture.

tion and tracking within the room are performed using the data from the four corner cameras and the ceiling camera.

The audio modules track people in the room via sound localization, and identify them based on their speech characteristics. Since sound and speech are not constantly available, this modality is mainly used for making the decisions of the system more robust. The acoustic identification can help the FBI module to re-assign true IDs to the detected blobs in case of tracking failures due occlusions or colour similarity. Similarly, acoustic localization is used to assign the ID of the speaker to one of the tracked blobs.

3 Visual Processing

The cameras in the system are responsible from motion detection, tracking, face detection and identification, and feature-based identification. State of the art methods in tracking humans can be grouped according to single-camera or multi-camera usage [18,52]. In multi-view approaches several methods are deployed using colour information, blob information or occupancy maps. When the scene is not crowded, simple background-foreground separation in combination with colour features can do the detection and tracking of the humans [31, 24]. In more crowded environments, Haritaoğlu *et al.* use vertical projection of the blob to help segment a big blob onto multiple humans [22]. Blob information and trajectory prediction based on Kalman filtering for occluded objects is used in [11].

3.1 Motion Detection Module

The motion detection module attempts to separate the foreground from the background for its operation. Foreground detection using background modeling is a common computer vi-

sion task, particularly in the field of surveillance [43]. The method we use is based on detecting moving objects under the assumption that images of a scene without moving objects show regularities, which can be modeled using statistical methods. The training set is constructed with a short sequence of offline recording taken from the empty room.

In order to adapt to illumination changes, we update the training set by adding new samples. At any time t, the colour value of pixel i can be written as $X_{i,t} = [R_{i,t}, G_{i,t}, B_{i,t}]$.

The recent history of every pixel within an image is stacked as $[X_{i,1}, X_{i,2}, ..., X_{i,t-1}]$ and modeled as a set of Gaussian distributions. With this approach, the probability of the current observation of a pixel i can be estimated using the model built from previous observations:

$$P(X_{i,t}|X_{i,1}, ..., X_{i,t-1}) = \sum_{j=1}^{t-1} w_{i,j} * \mathcal{N}(X_{i,j}, \mu_{i,j}, \sigma_{i,j}^2), \tag{1}$$

where $\mathcal{N}$ is the Gaussian probability density function. $\mu_{i,j}$ and $\sigma_{i,j}^2$ are mean and covariance matrix of the Gaussian. $w_{i,j}$ is the weight associated with the Gaussian. To make the process on-line, a matching process is carried out; a new pixel is considered to belong to the background if it matches with the current Gaussian component, i.e. if the distance between the pixel and the mean of the Gaussian in question is less than ε. In this study we have chosen $\varepsilon = 2 * \sigma$. If a current pixel does not match the mean of the given distribution, then the parameters of the distribution are updated with a higher weight, otherwise they are updated with a lower weight $w_{i,j}$. The adaptation procedure is as follows:

$$w_{i,t} = (1 - \alpha) w_{i,t-1} + \alpha M_{i,t} \tag{2}$$

where α is learning rate, $\alpha \in [0, 1]$ and $1/\alpha$ determines the speed of the adaptation process. The optimal value of α depends on the illumination conditions and on the background. It is empirically set to 0.1 in this work, and it has been observed that the model is not sensitive to small perturbations of it (because of the stable illumination conditions of the recordings). $M_{i,t} = 1$ if the current pixel matches a model, otherwise it is 0 for rest of the models. In a similar vein μ and σ are updated as follows:

$$\mu_{i,t} = (1 - \lambda) \mu_{i,t-1} + \lambda X_{i,t} \tag{3}$$

$$\sigma_{i,t}^2 = (1 - \lambda) \sigma_{i,t-1}^2 + \lambda (X_{i,t} - \mu_{i,t})^T (X_{i,t} - \mu_{i,t}) \tag{4}$$

where

$$\lambda = \alpha \mathcal{N}(X_t | \mu_{i,t-1}, \sigma_{i,t-1}) \tag{5}$$

We have used one second pixel history for modeling the Gaussian distributions, i.e. 25 frames.

One significant advantage of this technique is, as the new values are allowed to be part of the model, the old model is not completely discarded. If the new values become stabilized over time, the weighting changes accordingly, and new values tend to have more weight as older values become less important. Thus, if there is a movement of furniture within the room, the background model is updated rather quickly; and the same is true for lighting changes. The output of the motion detection module is further processed by a standard connected component analysis to remove motion fragments smaller than 20 pixels.

3.2 Multi-Camera Localization and Tracking

One of the major drawbacks of current tracking systems is the lack of reliable features to keep track of moving humans in unconstrained environments. The most popular visual features in use are colour features and foreground segmentation or movement features [31, 26, 50]. In this study, we use a probabilistic occupancy map (POM) approach, related to the algorithm proposed in [18], but simplified to deal with indoor environments, where motion trajectories are short and bursty when compared to the more consistent motion expected in an outdoor environment.

In the POM approach, the discretized occupancy map is used to back-project the stub image of a person (a simple rectangle) to each camera view. The overlaps between the stubs and the detected motion images across multiple cameras indicate the presence of a person at a given location. Denote the set of blobs detected at time t by B_t. Let L_k^t be a Boolean variable that indicates the presence of an individual at location k at time t. The algorithm starts by setting all L_k^t to zero. Then the best candidate is iteratively selected as

$$\arg\max_k P(L_k^t|B_t^-) \tag{6}$$

where B_t^- denotes the reduced set of blobs obtained by removing blobs accounted for so far, and L_k^t is set to one if a pixel support condition is satisfied (we have motion in the corresponding area from at least two cameras). $P(L_k^t|B_t^-)$ is set proportional to the pixel overlap between the stubs and the blobs. Thus, if the motion blob completely covers the stubs in all four cameras, the probability of occupancy is set to unity. Conversely, if no motion is detected in the stub locations across cameras, the probability of occupancy is zero. The process automatically stops once the support condition is is no longer satisfied.

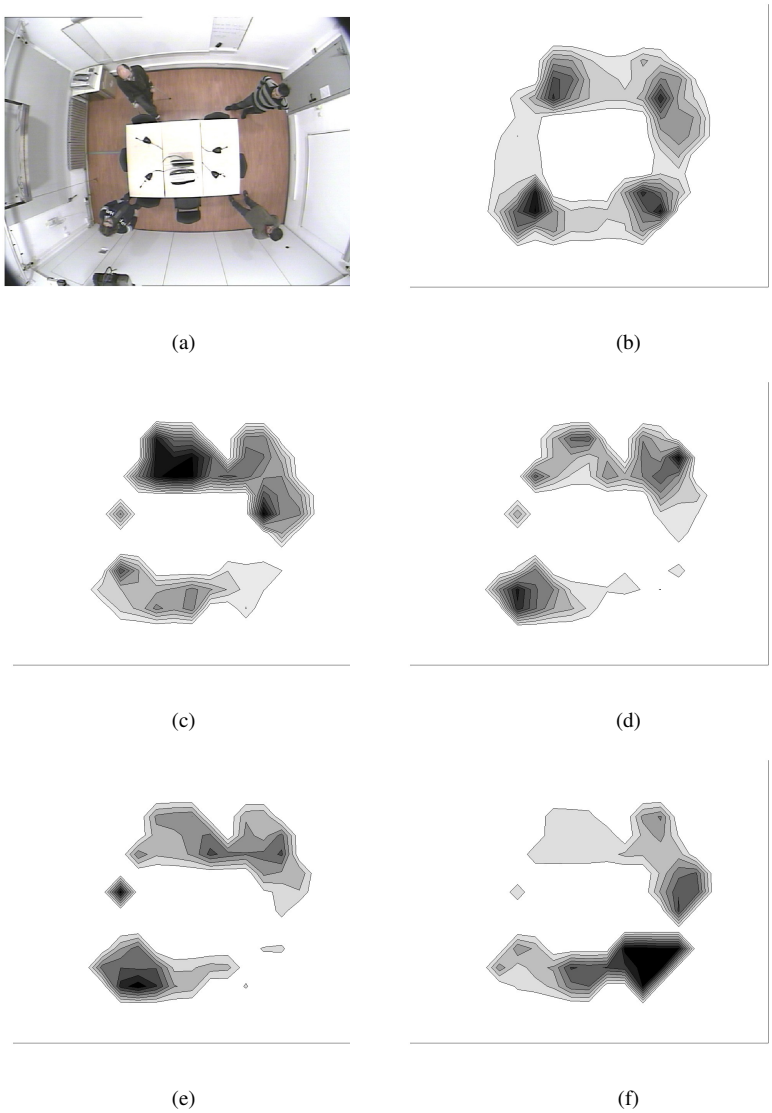
The accuracy of this algorithm in terms of correct occupancy detection on the first session is 96.3 per cent, allowing at most a single grid square deviation from the ground truth. The number of false detections are small, they amount to 5.5 per cent of the total detections. Of the true detections, 58.7 per cent are exact matches with the ground truth. The grid size is set to roughly 40 cm., which is slightly denser than [18] that uses 50 cm. On the more difficult second session, the correct occupancy detection is 91.7 per cent, but there are many more false detections (about 23 per cent of total detections) due to moved furniture.

For continuous detection and identification, the output of this module is combined with other types of information at a later stage. In this mode of operation, it is not necessary to select locations, and the POM is used to output probability values for each grid location. Fig. 4(b) illustrates the occupation probability distribution produced by the algorithm for a sample frame. Only the four corner cameras are used in computing the occupancy, the ceiling camera is just used for ground truth annotation and visualization.

3.3 Face Detection Module

Face detection is needed for face identification and feature based identification modules. In this module, the face of each person present in the scene must be detected roughly (i.e. a bounding box around a face will be the output of this module).

For face detection, we use the OpenCV face detection module that relies on the adaboosted cascade of Haar features, i.e. the Viola-Jones algorithm [49]. The client that performs face detection receives a video flow from a client that in its turn directly receives its input from one of the cameras, and outputs a flow that contains the bounding box of the



(a) (b)

(c) (d)

(e) (f)

Fig. 4 The operation of POM and FBI modules. (a) A sample frame from the second session. (b) The POM is computed through motion detection in four corner cameras. Darker colours indicate higher probability. (c)-(f) The posterior probabilities under models of individuals in the room, computed by the FBI module. The order of maps reflects the relative positions of the persons in the room. While the persons in (c), (e) and (f) are unambiguosly identified, the model for the top-right person produces high posterior probabilities for two persons as shown in (d). The assignment of identities to occupied locations that jointly maximizes the posterior probability is able to identify all persons correctly.

detected face. The face images captured by ceiling cameras are too small for reliable detection or identification. Consequently, only the door camera is used in face detection and recognition.

3.4 Face Identification Module

During the last years, much research has been devoted to video based face recognition (for an extensive review, see Zhou et al. [53]). In addition to recognizing faces from single image captures, the continuous monitoring environments offer the possibility of multiple-still-based face recognition.

In this work, an existing technique for face recognition in smart environments is used [48, 30]. The technique takes advantage of the continuous monitoring of the environment and combines the information of several images to perform the recognition. Models for all individuals in the database are created off-line using sequences of images collected at a different date than the training and testing recordings.

The face recognition module takes motion tracking and face detection for granted. This module therefore subscribes to the face detection flow that indicates face locations, and to the video flow to analyze the visual input coming from the door camera. The combination of these modules results in a completely automatic face detection/identification system.

The algorithm works with groups of face images provided by the face detection module. For each test sequence, face images of the same individual are gathered into a group. Then, for each group, the system compares these images with the gallery images for each person. A PCA based approach [28] has been used for comparison, which has the low computational complexity that is necessary for online applications. Individual decisions are combined into a single decision for a group of images using the algorithm described in in [48].

The face detection module, by way of its construction, forwards only frontal faces for identification, i.e. faces where the two eyes, the nose and the mouth are visible. The gallery images for each person are generated in a similar automatic fashion. During training, candidate faces are normalized to a size of 60×40 pixels and projected onto the subspace spanned by the first 200 eigenvectors of the data covariance matrix, created from the gallery images added so far. The resulting vector is added to the model only if it is sufficiently different from the vectors already present in the model. The reader is referred to [48] for the details. The subspace dimensionality is empirically determined, and the number of eigenvectors is intentionally kept high to ensure reliable identification, as this is the point where we start tracking individuals.

3.4.1 Experimental results

We have evaluated the face identification module combined with the face detection module for the door camera (PTZ), which records high resolution images as people enter the room. Its positioning allows the capture of a head-and-shoulders image with a resolution of 768×576 pixels. As pre-processing, the images are downsampled by a factor of two. This removes artifacts caused by interlacing.

The identification module operates in real-time, giving one identification result each second, when a person is visible. Since the camera operates at 25 fps, up to 25 images acquired in succession are used for each identification result. The camera is positioned in a way to allow only one person to be present in its field of view, as long as the entrance is restricted to one person at a time.

As previously mentioned, the gallery models are created with images of each person taken from a different recording. We have used 20 training images per subject, and the dataset was collected with four people participating in the recordings.

Note that the input to the face ID system is the output of the face detection module, so the results shown here do not suppose perfect face detection. For the first session (TRA), the

face detection module outputs 172 groups of faces. Of these 172 groups, only 111 (65 per cent) correspond to good detections of frontal faces. The rest of the detections correspond to non-frontal faces and false positives (35 per cent).

The identification module is able to give correct results in all the correctly detected groups of faces. In the case of false positives of the face detection module, the face identification module correctly classifies all the cases in the *Unknown/No face* class.

The high accuracy is the result of the fact that the face recognition module was initially developed for handling more difficult problems (i.e. more subjects and lower resolution of the images). The intra-subject and inter-subject variation is illustrated in Fig. 5.

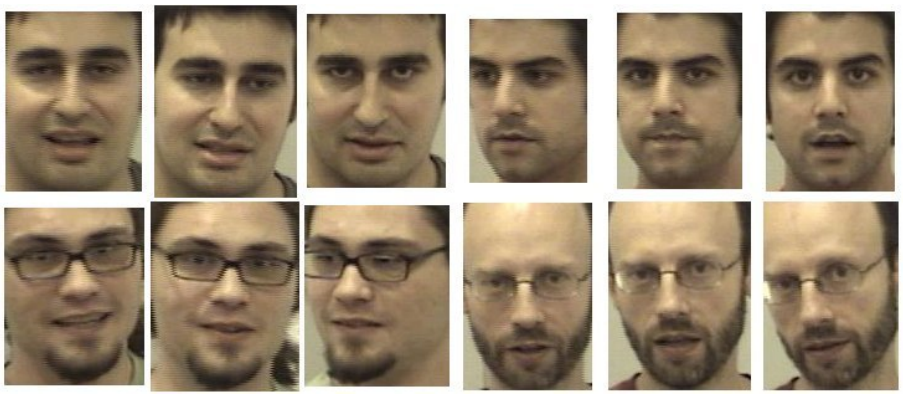


Fig. 5 Examples demonstrating variability in faces, acquired with the door camera. Variations due rotation, expression change and motion blur are apparent.

For the second session (TEST), the face detection module outputs 23 groups of well detected faces at the entrance stage. Of these, 8 (35 per cent) are correctly identified and 15 are classified in the *Unknown/No face* class. For the first session, recorded a few weeks after the collection of the images used to generate the gallery, all the face groups are correctly identified. The second data set was collected several months later. As a result, many of the detected groups of faces are classified in the *Unknown/No face* class, as PCA is not robust against time variability in the face images. But even in that case, the system can be used for reliable identification, because all the participants are correctly identified at least once when they enter, and there are no wrong initializations. Note that the models can be automatically updated during each system operation to avoid excessive time variability. See [48] for further details. For scenarios where the number of users prohibits reliable face recognition at the door, an RFID-based identification can serve as a foolproof initialization method. The rest of the system requires no modification to accomodate such a change.

3.5 Feature Based Identification Module

In order to ensure robust identification in the room, we rely on weak or unanticipated sensor correlations and patterns for sensing. For this purpose, we propose to use features that are easy to capture, present most of the time, and helpful when the strong modalities (e.g. face or speech) are unreliable. This is particularly important in the smart-room scenario, where the

ceiling cameras usually cannot capture discriminative face images. Most proposed systems for this type of application rely on continuous tracking of identified people in the room. Important approaches that are previously proposed and successfully used are Kalman filters [25] and particle filters [34]. However, tracking multiple people for a long period is difficult, there is little possibility of recovering from mistakes.

The feature based identification (FBI) module we propose aims at identifying persons in the room when the tracking or speaker identification results are not available, or not discriminatory. The primary assumption behind the operation of this module is that the variability in a user's appearance for a single camera is relatively low for a single session (this case is termed intra session in [48]), and a user model created on-the-fly can provide us with useful information [46]. We use the following general procedure for this purpose: Whenever a person is reliably identified (i.e. when the face identification or speaker identification modules return a result with high confidence[1]), the tracking cameras forward the detected motion blobs to the FBI module. Then, the pixel intensities within the motion blobs are modeled statistically, and this statistical model is used to produce posterior probabilities for identification purposes. Typically, the system creates one model per person per camera, immediately after the person enters the room, as this is the point where the door camera acquires the face image. Using a separate module for each camera makes a colour-based calibration across cameras unnecessary. In the remainder of this section, we describe the operation of FBI in more detail. We have omitted camera indices for simplicity.

The colour modeling for the subjects is based on the assumption that the pixels are independent and identically distributed, following a distribution on the RGB (or HSV) space. Fleuret *et al.* argue that this approach is sufficient in practice, and a body area based segmentation is not necessary [18]. For a set of pixels X, and a set of classes C, where each class C_i is modeled with a mixture model G_i, the posterior probability of class C_i with evidence X is written as:

$$p(\mathrm{C}_i|X) = \frac{p(\mathrm{C}_i)p(X|\mathrm{C}_i)}{p(X)}, \tag{7}$$

and $p(X)$ is equivalent to $\prod_{j=1}^{n} p(x_j)$, a product over the pixels in the set $X = \{x_1, x_2, \ldots, x_n\}$. We can assume that $p(\mathrm{C}_i)$ are uniformly distributed for simplicity, or we can use it to inject some previous knowledge, e.g. the previous positions and velocities of people can be used to condition $p(\mathrm{C}_i)$.

There are two problems with this formulation. First of all, the model construction is performed on the fly, and might be incomplete. The model indicates the presence of a subset of features belonging to class C_i, but not necessarily the absence of the rest. However, the evidence of each pixel x_j contributes to the probability term in Eq. 7. Consequently, for a very low $p(x_j|\mathrm{C}_i)$, the presence of x_j serves as a strong negative evidence against class C_i. The second problem is the possible presence of small clusters of distinguishing features (e.g. a small colourful badge). Even if we assume that the model has captured this feature in the distribution, the likelihood under the class model will suffer from the small component prior, and this feature will not contribute much to the overall likelihood.

The first problem can be solved by using only 'positive' evidence for each class. We can produce a ranked list $S_i = \{x^j\}$ from X for each class, which sorts the pixels according to

[1] The confidence depends on the modality. For face identification, we set a threshold on the RML of the class with the highest $M(C_j)$. The probabilistic nature of detection precludes hundred per cent confidence in classification, but a high ratio of probability for the highest probability class and its successor is usually indicative of a good decision. Since the posterior is normalized, a threshold of 90 per cent means that the best candidate is about ten times more likely than its successor. In practice, the face identification part of the system is very accurate, and consequently very robust to the actual value of the threshold.

their likelihood values with respect to the model of C_i:

$$p(x^j|C_i) \geq p(x^{j+1}|C_i) \quad \forall j = 1 \ldots n \tag{8}$$

Then, $S_i(\tau)$ is used for computing the posterior instead of X:

$$p(C_i|X) \sim p(C_i|S_i(\tau)) \tag{9}$$

with τ representing the fraction of pixels that are retained. For $\tau = 1$, the whole $S_i = X$ is consulted for evidence.

The second problem involves the presence of small, discriminative features. The distinguishing characteristic of these features is that even though their class-conditional probability is small, it is much smaller under models of the other classes. We can use this fact to select these features by looking at the likelihood ratios instead of the likelihood itself. We retain Eq. 9, but change Eq. 8 accordingly:

$$\frac{p(x^j|C_i)}{\max_m p(x^j|C_m)} \geq \frac{p(x^{j+1}|C_i)}{\max_k p(x^{j+1}|C_k)} \quad \forall j = 1 \ldots n \tag{10}$$

with $i \neq k, i \neq m$.

To build statistical models G_i for the users of the smart room, we contrast several approaches of increasing complexity in the RGB or HSV space. Our simplest approach is a single Gaussian, followed by a mixture of Gaussians with either 2 or 3 components, or a mixture of factor analysers with arbitrary number of components. For the last approach, we use the IMoFA algorithm that tailors the mixture complexity automatically to the dataset [40]. When modeling the colour distribution of a subject, the spherical or diagonal covariance assumptions are usually not justified, and we use full covariance for the Gaussian models. Fig. 6 shows a sample colour distribution, which illustrates this point graphically.

We evaluate the FBI models for about 8.000 blobs extracted from four camera sequences. The ground truth for comparison is obtained from the manual annotation. Fig. 7 shows the accuracy of the different approaches for different values of τ, which is the threshold of discriminativeness.

Our results confirm our prediction that consulting positive evidence is a better approach. The trend for accuracy decrease with the increasing proportion of consulted pixels is apparent from Fig. 7. This decrease is less for mixtures with larger number of components (most notably the IMoFA models) that are able to model the distributions more accurately. The accuracy peaks surprisingly early, suggesting that consulting around twenty per cent of the pixels should be enough. The figure also indicates a slight superiority of the HSV space to the RGB space.

4 Audio Processing

A total of 20 microphones were used to track speakers in the room. Of these, 12 omni-directional microphones were placed on the walls in three T-shaped groups of four microphones each. In addition to these arrays, four directional and four omni-directional microphones were placed on the table. The data were collected at a rate of 44.100 kHz, 2 bytes per sample, and downsampled to 16.000 kHz as a pre-processing step.

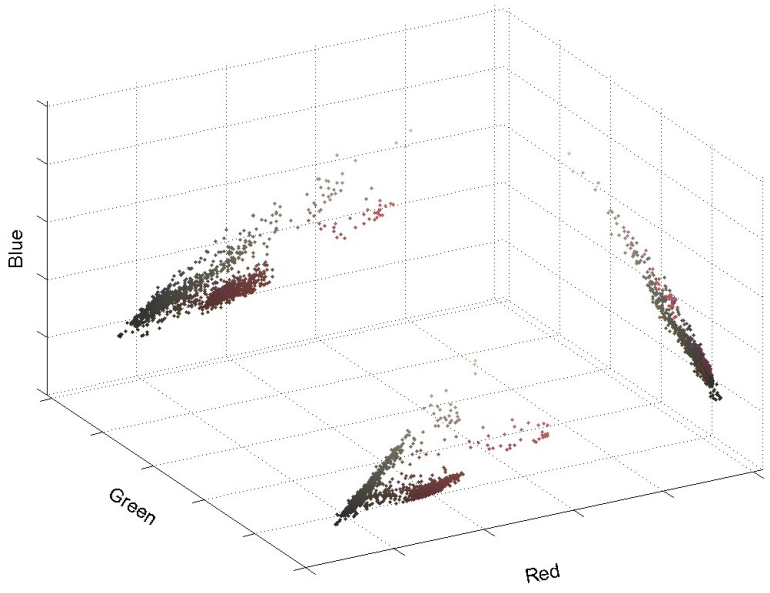


Fig. 6 Colour distribution for a sample subject in the RGB space, where each pixel is projected to each of the 2D axial planes (red=255, green=255, and blue=0, respectively) to show the 3D point distribution from three different views. The distribution calls for non-zero covariances, and multiple components.

4.1 Acoustic Localization Module

Many approaches to the task of acoustic source localization in smart environments have been proposed in the literature. Their main distinguishing characteristic is the way they gather spatial clues from the acoustic signals, and how this information is processed to obtain a reliable 3D position in the room space. Spatial features, like the Time Difference of Arrival (TDOA) between a pair of microphones [38] or the Direction of Arrival (DOA) of sound to a microphone array can be obtained on the basis of cross-correlation techniques [35], High Resolution Spectral Estimation techniques [36] or by source-to-microphone impulse response estimation [15]. Depending on such features, the source position that agrees most with the data streams and with the given geometry is selected. Conventional acoustic localization systems also include tracking stage that smooths the raw position measurements to increase precision according to a motion model. These techniques need several synchronized high-quality microphones.

The acoustic localization system used in this project is based on the SRP-PHAT localization method, which is known to perform robustly in most scenarios. The SRP-PHAT algorithm (also known as Global Coherence Field [17]) tackles the task of acoustic localization in a robust and efficient way. In general, the basic operation of localization techniques based on steered response power (SRP) is to search the room space for a maximum in the power of the received sound source signal using a delay-and-sum or a filter-and-sum beamformer. In the simplest case, the output of the delay-and-sum beamformer is the sum of

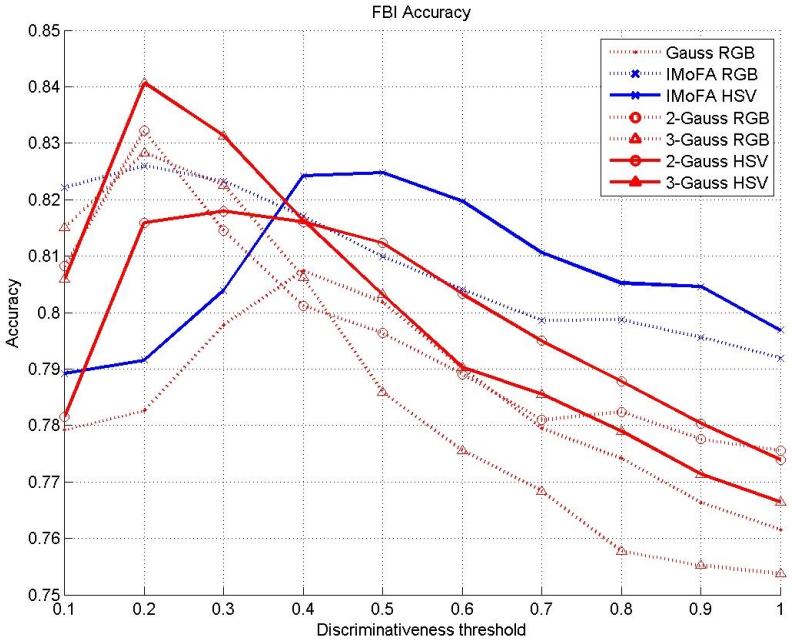


Fig. 7 Accuracy of the FBI module for different statistical models. Consulting only the more discriminative pixels as indicated by the threshold on the *x*-axis is beneficial.

the signals of each microphone with the adequate steering delays for the position that is explored. The SRP-PHAT algorithm consists of exploring the 3D space while searching for the maximum of the contribution of the PHAT-weighted cross-correlations between all the microphone pairs. The SRP-PHAT algorithm performs very robustly due the the PHAT weighting, keeping the simplicity of the steered beamformer approach.

Consider a smart-room provided with a set of N microphones from which we choose M microphone pairs. Let $\mathbf{x}$ denote a $\mathbf{R}^3$ position in space. Then the time delay of arrival $TDOA_{i,j}$ of an hypothetic acoustic source located at $\mathbf{x}$ between two microphones i, j with position $\mathbf{m}_i$ and $\mathbf{m}_j$ is:

$$TDOA_{i,j} = \frac{\| \mathbf{x} - \mathbf{m}_i \| - \| \mathbf{x} - \mathbf{m}_j \|}{s}, \tag{11}$$

where s is the speed of sound.

The 3D room space is then quantized into a grid of positions with typical separations of 5–10 centimeters. The theoretical TDOA $\tau_{\mathbf{x},i,j}$ from grid locations to each microphone pair are pre-calculated and stored. During the operation, PHAT-weighted cross-correlations of each microphone pair are estimated for each frame [35]. These can be expressed in terms of the inverse Fourier transform of the estimated cross-power spectral density $G_{m_1 m_2}(f)$ as follows:

$$R_{m_i m_j}(\tau) = \int_{-\infty}^{\infty} \frac{G_{m_i m_j}(f)}{|G_{m_i m_j}(f)|} e^{j2\pi f \tau} df, \tag{12}$$

The estimated acoustic source is the grid location that maximizes the contribution of the cross-correlation of all microphone pairs:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \sum_{i,j \in \mathbb{S}} R_{m_i m_j}(\tau_{\mathbf{x},i,j}), \tag{13}$$

where $\mathbb{S}$ is the set of microphone pairs. The sum of the contributions of each microphone pair cross-correlation gives a value of confidence for the estimated position, which can be used in conjunction with a threshold to detect acoustic activity and to filter out noise. In our work, we use a threshold of 0.5 for each microphone cluster. It is important to note that in the case of concurrent speakers or acoustic events, this technique will only provide an estimation for the dominant acoustic source at each iteration.

The experimental results obtained with the localization module are given in Section 5, as they are used jointly with the speaker identification module.

4.2 Speaker Segmentation and Identification Module

Tracking of speakers inside the room involves both audio segmentation and speaker identification, two tasks particularly researched in the context of spoken document indexing. Recently, automatic segmentation and clustering of audio segments based on speaker identity received renewed attention, and applied to the problem of detecting cases where a speaker intervenes in a conversation [5, 32, 39].

In speaker segmentation, the identification of speakers is not required and no assumptions are made about the number of speakers or the speaker characteristics [8, 7]. On the other hand, speaker tracking aims at detecting regions uttered by a given speaker, for which a speaker model is trained beforehand.

Most of the proposed applications in the literature analyse the tracking problem in three parts:

1. Detecting the segments that contain speech.
2. Detecting the speaker turn.
3. Identifying the speaker.

Our speaker identification system is based on Hidden Markov Models (HMMs), which allow for a global segmentation of the audio sequence with the maximum a posteriori (MAP) approach that is optimal in the maximum likelihood sense [37]. The HMMs are used to model speaker features and to encode the temporal evolution of the speech segments. The algorithm segments the audio signal into several clusters using a minimum duration of the speaker turn parameter, and assigns the most likely identity from the closed database of target speakers to each segment. We have collected a small database from four speakers, and one minute of speech per speaker was collected in the room to estimate the initial models.

The algorithm starts with several stages of pre-processing. The input signals from each microphone channel are first Wiener-filtered using the implementation of the QIO front-end system [4]. These channels are then fed into the *Beamforming* code implemented by ICSI in order to obtain a single enhanced channel for further processing [6]. This processed output channel is analyzed by the *Speech Activity Detector* (SAD) module of UPC, described in [47] in order to obtain the speech segments. The non-speech segments are not considered any further. The enhanced speech data are parameterized using 19 Mel Frequency Cepstral Coefficients (MFCC) features and fed into an iterative clustering system, based on [29]. However, the system described in [29] is designed with the diarization task in focus, and

therefore oblivious to the identity. Consequently, the algorithm is modified to take into account prior knowledge of speakers, as well as general sound characteristics of a meeting scenario. In the remainder of this section we treat each of the stages briefly, for completeness' sake.

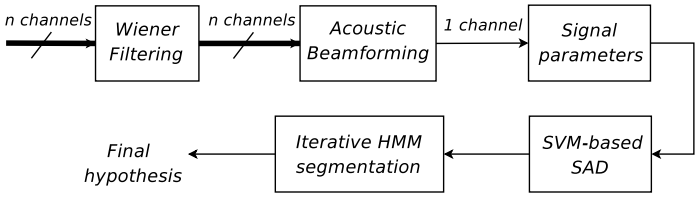


Fig. 8 The schematic outline of the speaker identification and segmentation system.

4.2.1 Speech Activity Detection

The SAD module used in this work is based on a support vector machine (SVM) classifier [41]. The performance of this system was shown to be good in the 2006 Rich Transcription SAD Evaluations (RT06) [47]. A GMM-based system that ranked among the best systems in the RT06 evaluation was selected as a baseline for performance comparison [47].

For classical audio and speech processing techniques that involve GMMs, the training data are in the order of several hundred thousand examples. However, SVM training usually involves far less training samples, as the benefit of additional samples is marginal, and the training becomes infeasible under too large training sets. In this work we use the NIST RT05 and RT06 datasets for samples of Speech and Non-Speech classes and employ proximal SVMs (PSVM) in the same way as proposed in [47] to reduce the amount of data for training without losing accuracy. Unlike conventional SVM, PSVM solves a single square system of linear equations and thus it is very fast to train [19]. We penalize errors from the Speech class more in comparison to errors from the Non-Speech class. The performance of the SAD algorithm is shown in Table 1.

4.2.2 Speaker Modeling

The Person Identification (PID) problem consists of recognizing a particular speaker from a segment of speech spoken by a single speaker. The assumptions under which we build our system are matched training and testing conditions, far-field data acquisition, limited amount of training data, and no a-priori knowledge about the room environment. The speech characteristics of each speaker in the system are modeled with Gaussian Mixture Models (GMMs). Diagonal covariance matrices are assumed, and the number of components per mixture is set to 32. The large number of components assures that the statistical learning is robust, and its use is further justified by the availability of a very large training set.

The speech parameterization is based on a short-term estimation of the spectrum energy in several sub-bands. The beamformed channel is analysed in frames of 30 milliseconds at intervals of 10 milliseconds and 16 kHz of sampling frequency. The algorithm commences by processing each data window by subtracting the mean amplitude, supposing the DC offset is constant throughout the waveform. A Hamming window was applied to each frame

and an FFT is computed. The FFT amplitudes are then averaged in 19 overlapped triangular filters, with central frequencies and bandwidths defined according to the Mel scale. The scheme we present follows the classical procedure used to obtain the Mel-Frequency Cepstral Coefficients (MFCC) [16].

The parameters of the model are estimated from speech samples of the speakers using the Baum-Welch algorithm [37]. Although this procedure is sensitive to initial conditions when the training data are few, under the present conditions, 15 iterations are demonstrably enough for robust parameter convergence. In addition to the speaker models, a Universal Background Model (UBM) is estimated to deal with cases that do not match any of the speakers.

4.2.3 Iterative Segmentation System

Our segmentation algorithm models the acoustic data using an ergodic Hidden Markov Model (HMM) with 5 states, four for the modeled speakers and one for the UBM, respectively. Each state contains a set of S sub-states as shown in Fig. 9 (a), each identical in terms of parameters. These sub-states are used to enforce a minimum speaker duration constraint, and they model the spectral features of speakers via Gaussian mixtures with shared parameters. The transition probabilities between subsequent sub-states are set to unity, except for the last sub-state where the self-transition probability is set to α and transition probability to any other state is set to $\beta = 1/\#states$. Thus, once a segment exceeds the MD, the HMM state transitions no longer influence the turn length; turn length is solely governed by acoustics [7]. Since we do not impose prior assumptions for the average turn length, we adjust the values of α and β so that $\alpha = \sum \beta$.

Using the HMM classifier for segmentation brings a number of advantages. The minimum duration constraint and the extension to additional classes/states can easily be imposed. For instance, it is possible to vary the MD by simply changing the number of sub-states within each class. It is also possible to impose different MD constraints for different acoustic classes.

Fig. 9 (b) depicts the flow of the algorithm. The MD is empirically set to 3 seconds. After the segmentation and the Viterbi decoding, an initial clustering of each class is obtained. These Speaker and Event clusters are then merged with the initial enrollment data, and the class models are re-trained by means of a MAP adaptation of the mean. After the adaptation, a final segmentation is conducted with a lower value of the MD (set to 1.5 seconds) and a final segmentation/identification hypothesis is obtained.

During the segmentation, some verification techniques have been also used. The emission probabilities of each state are normalized with the frame score computed from the UBM model by using the Log-Likelihood Ratio (LLR) method at score-level [10]:

$$\hat{L}_{\lambda}(X) = L_{\lambda}(X) - L_{UBM}(X) \tag{14}$$

where X denotes the segmented sequence, $L_{\lambda}(X)$ denotes the likelihood under class model λ and $L_{UBM}(X)$ is the likelihood under the UBM, trained with the complete training sequence.

4.2.4 Assessment protocol and results

We briefly recall here the content of the corpus, as well as the assessment rules for the speaker tracking task we have used to carry out the experiments. The training sequence (TRA) is of around 5 minutes of duration, and it is used to determine the model parameters.

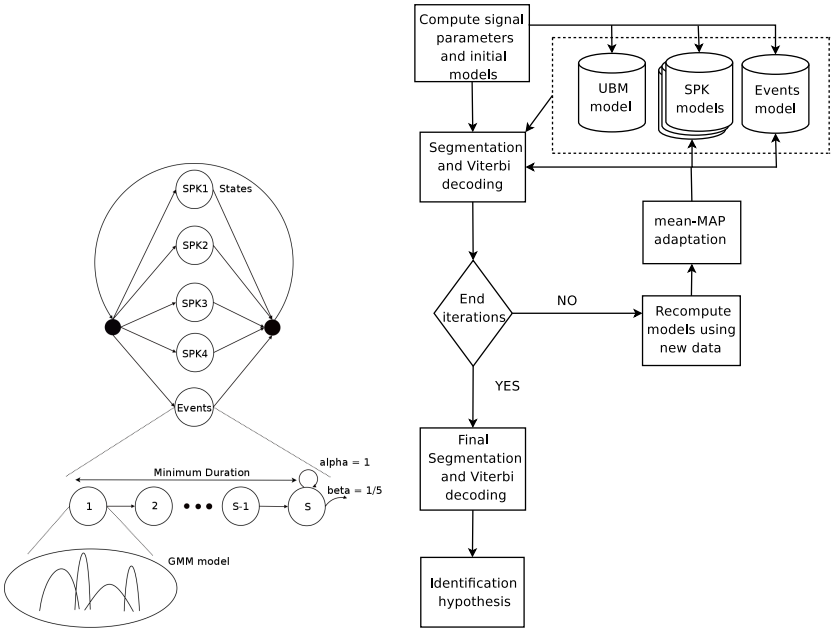


Fig. 9 (a) An ergodic HMM models the acoustic data with 5 states, each composed of *S* sub-states sharing the same GMM model. (b) Steps of the iterative segmentation and identification algorithm.

Table 1 SAD performance

Dataset	EVAL TIME	EVAL SPEECH	MISSED SPEECH	FA SPEECH
TRA	281.19 secs	201.39 secs	9.65 secs	1.82 secs
TEST	422.93 secs	371.42 secs	10.68 secs	6.82 secs

All modeled speakers and other classes of acoustic events are present in the recordings. Background noise, overlapping speech, door slam, laugh, and steps are some examples of these events. Brief silences of a speaker between talk segments were not labelled if smaller than 500 ms. One minute of audio per class was used for the training of each speaker and the event class. Finally, an audio sequence of roughly 7 minutes was employed to benchmark the performance of the approach (TEST).

The metric used to evaluate the performance of the system is the Diarization Error Rate (DER), which is also used in the NIST RT 2006 evaluation campaign. It is computed by first finding an optimal one-to-one mapping of the reference speaker ID to the system output ID, and then obtaining the error as the percentage of time that the system assigns a wrong speaker label. The results given in Table 2 are the time-weighted DER averages for the two evaluation sets. The relation of DER with the number of performed iterations is given in Fig. 10.

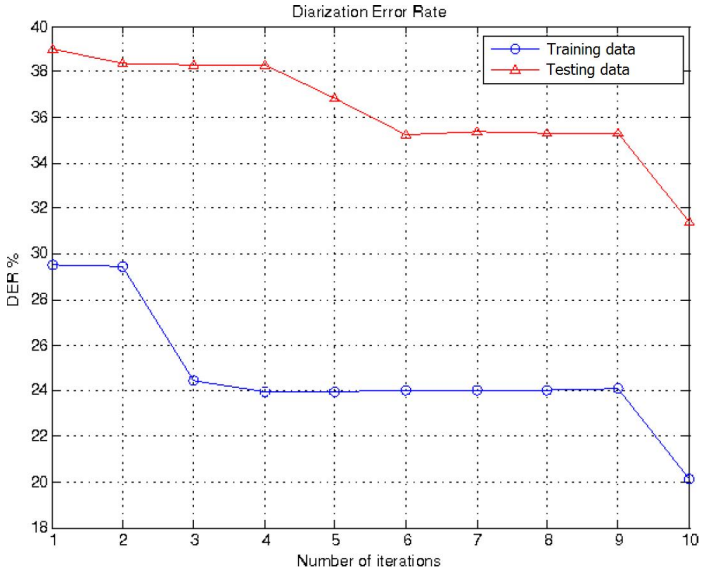


Fig. 10 Diarization Error Rate (DER) for training and test datasets. The final gap in the DER at the last iteration is due to the change (1.5 seconds instead of 3 seconds) in the minimum duration of the speaker turn.

Table 2 Diarization performance

Dataset	SCORED	MISSED	FA	SPK ERRORS	DER
TRA	164.28 secs	12.50 secs	1.82 secs	18.75 secs	20.12 %
TEST	315.99 secs	11.08 secs	6.82 secs	81.28 secs	31.39%

5 Multimodal Tracking and Identification

The purpose of the system is to identify each user as they enter through the door and then track them during the complete session. To have robust tracking and identification of all persons at the same time, we have used a multimodal approach. In this section we describe this approach.

The most reliable component of the system is identified as the face detection and recognition module. For this reason, the identification starts at the door, where the PTZ camera identifies the person. Typically, a single identification result is enough to bootstrap the system, although multiple subsequent identifications can be used in conjunction to bolster the confidence of the system. With the present setup, we have not seen any need for further fusion of results, as we have perfect identification at this stage.

Once the person is identified, one FBI model per camera is created on the fly. This operation is fast, and based on results shown in Section 3.5, we use 3-component Gaussian mixtures on the HSV colour space as FBI models for each person. The POM module (Section 3.2) provides the motion blobs used for FBI modeling.

When acoustic information is available, the acoustic localization of the sound source is jointly considered with speaker identification output. We use a particle filter based tracker to combine audio and vision based identification and localization information [21, 12]. This algorithm is used to estimate the 2D position and the identity of a person jointly.

5.1 Particle Filter-Based Tracking

The particle filter (PF) is a useful technique for tracking and estimation tasks, especially when the estimated motion uncertainty is not accurately modeled with Gaussian distributions [23]. They were previously employed for audio-visual multi-person tracking [20, 55], multimodal event detection [54] and for active speaker tracking [14, 34] successfully. In [55] the PF model incorporates the 3D location and velocity of tracked objects, and also employed for calibrating the sensors. In [34] the PF model is used for 3D locations only, and the motion of the object is modeled with a Gaussian diffusion equation. In [14], the 2D ground position of the person and the height are tracked for a multiple person activity tracking system. While these kind of systems produce good tracking results, the spatio-temporal covariance matrices used for differentiating tracked persons are not robust enough to recover tracks once two tracks are confused. The tracking system proposed in this section includes the ground position, the identity, and whether a person is speaking or not. The addition of independent identification gives the system flexibility to recover from tracking errors, and the identity of the tracked persons can be preserved in cases of track confusion.

Our approach is based on a multi-hypothesis tracker that approximates the filtered posterior distribution by a set of weighted particles. The standard particle filter assigns weights to particles based on a likelihood score, and then propagates these weighted particles according to a motion model. The optimal solution to multi-target tracking using the PF is the joint particle filter, but its computational load increases dramatically with the number of tracked targets, since every particle estimates the location of all targets in the scene simultaneously. To deal with the computational complexity, a set of decoupled PFs are used (one per target) in [27], and an interaction model is defined to ensure track coherence.

Let us denote the position of a person at an instant t as $\mathbf{x}_t$. Estimating the position given a set of observations $\mathbf{z}_{1:t}$ can be formulated as a state space estimate problem, described by the following state process equation:

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{v}_t), \tag{15}$$

and the observation equation:

$$\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{n}_t), \tag{16}$$

where $\mathbf{f}$ is a function describing the state propagation, and $\mathbf{h}$ is an observation function modelling the relation between the the hidden state $\mathbf{x}_t$ and its observable counterpart $\mathbf{z}_t$. The functions $\mathbf{f}$ and $\mathbf{h}$ are possibly nonlinear and the noise components $\mathbf{v}_t$ and $\mathbf{n}_t$ are assumed to be independent stochastic processes with a given distribution.

To solve the tracking problem from a Bayesian perspective, we need to calculate the probability distribution function $p(\mathbf{x}_t|\mathbf{z}_{1:t})$, and this can be done recursively. The *prediction* step uses the process equation Eq. 15 to obtain the prior probability distribution function (henceforth denoted as pdf) by means of the Chapman-Kolmogorov integral:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})\, d\mathbf{x}_{t-1}, \tag{17}$$

with $p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$ known from the previous iteration and $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is determined by Eq.15. When a measurement $\mathbf{z}_t$ becomes available, it may be used to update the prior pdf via Bayes' rule:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t|\mathbf{x}_t)\, p(\mathbf{x}_t|\mathbf{z}_{1:t-1})}{\int p(\mathbf{z}_t|\mathbf{x}_t)\, p(\mathbf{x}_t|\mathbf{z}_{1:t-1})\, d\mathbf{x}_t}, \tag{18}$$

where $p(\mathbf{z}_t|\mathbf{x}_t)$ is the likelihood statistics derived from Eq.16. However, the posterior pdf $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ in Eq.18 can not be computed analytically unless linear-Gaussian models are adopted, in which case the Kalman filter provides the optimal solution.

Particle Filtering (PF) is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The posterior density function $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ is represented by a set of random samples (particles) with associated weights:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{j=1}^{N_s} w_t^j \delta(\mathbf{x}_t - \mathbf{x}_t{}^j), \tag{19}$$

where w_t^j are the weights associated to the particles satisfying $\sum_{j=1}^{N_s} w_t^j = 1$. As the number of samples N_s increases, the characterization of posterior pdf improves, and the PF approaches the optimal Bayes estimate.

The principal steps in the PF algorithm are:

1. *Resample*: the particles are resampled according to their scores. This operation results in the same number of particles, but particles with high scores are duplicated, while particles with low scores are dropped.
2. *Apply motion model*: predict the new set of particles by propagating the resampled set according to a model of the target's motion.
3. *Score*: Form the likelihood function $p(\mathbf{z}_t|\mathbf{x}_t)$, assign weights to the new particles according to the likelihood function $w_t^i = p\left(\mathbf{z}_t|\mathbf{x}_t^i\right)$, and normalize the weights so that $\sum_i w_t^i = 1$.
4. *Average*: the location of the target is estimated as the weighted sum of all the particles. $E[\mathbf{x}_t] = \sum_{i=1}^{N} w_t^i * \mathbf{x}_t^i$

The state we want to estimate is the position and identity of each person. The propagation model for the particles is modelled by adding white noise to the positions and by randomly changing a given percentage of the identities. Likelihood functions used for scoring are computed separately for each modality. For a given filter, the best state at time t is obtained by computing an histogram of the identities of the particles and selecting the maximum to decide the identity. The mean and the variance for the particle positions are estimated, and used in deciding whether the filter is tracking a person or not.

In our implementation, the interaction between filters is modelled by iterating the last two steps. If a particle enters the quantized space occupied by another filter tracking a person, the weight of this particle is set to zero. When a particle randomly changes its identity and takes on the identity of another filter with a higher score, the weight of this particle is also set to zero.

5.1.1 Video Evaluation

In order to implement a PF that takes into account visual information solely, the visual likelihood evaluation function must be defined. We used a combination of POM and FBI by multiplying both terms. POM gives a probability of occupancy for each grid location and FBI gives the posterior probabilities for all subjects at each grid position.

Function $p(\mathbf{z}_t|\mathbf{x}_t)$ can be seen as the likelihood of a particle belonging to the position corresponding to a person. For a given particle j occupying a room position, the likelihood may be formulated by multiplying the probabilities from POM and FBI, which will give the weight for the particle.

The particle filters are initializated with 10,000 particles distributed randomly across the room. Each particle is characterized by its position and its identity. The number of particles,

Table 3 Tracking performance

	Visual		Visual+Acoustic	
Data sets	MOTA	MOTP	MOTA	MOTP
TRA	65.4%	337 mm	67.2%	307 mm
TEST	23.0%	291 mm	40.9%	256 mm

as well as the other parameters of the PF are optimized on the training set, and used without any modifications on the test set.

5.1.2 Multimodal Evaluation

In the multimodal case, acoustic information is added to the visual information. Each modality (visual localization - POM, visual identification - FBI, acoustic localization - AcLoc, and acoustic identification - SpkId) provides probabilities for localization and identification. Combination of modalities is achieved by means of a weighted sum rule, with weights obtained experimentally on the training set.

$$w_t^i = w_1 * (POM * FBI) + w_2 * (AcLoc * SpkId) \tag{20}$$

The same number of particles and same parameters are used as in the video evaluation.

5.2 Experimental results

For both tracking evaluations, we have used the metrics proposed in [9]. The Multiple Object Tracking Precision (MOTP) shows a tracker's ability to estimate precise object positions, whereas the Multiple Object Tracking Accuracy (MOTA) expresses the performance for estimating the correct number of objects, and keeping to consistent trajectories. MOTP scores the average metric error when estimating multiple target centroids, while MOTA evaluates the percentage of frames where the targets have been missed, wrongly detected or mismatched. MOTA is the sum of all errors made by the tracker (i.e. false positives, misses, mismatches) over all frames, averaged by the total number of ground truth points. It is similar to accuracy metrics widely used in other domains and gives a very intuitive measure of the tracker's performance, independent of its ability to determine exact person locations. Low MOTP and high MOTA scores are preferred, indicating low metric error when estimating multiple target positions and high tracking performance, respectively. These metrics are preferred over receiver-operator characteristic (ROC) curves, since tracking imposes several constraints (e.g. one and only one track per person) on each detected person, independent of a confidence threshold that can be varied to generate the ROC curve.

Results in Table 3 show that the tracking is highly successful on the training recording. Even though video tracking is good, the audio modalities result in an improvement of the performance and precision of the system. For the test recording, the results are poorer, as the visual identification is much more difficult. The FBI module has problems discriminating the individuals as they wear similar clothing. Subsequently, the benefit of using additional acoustic information is marked for this recording. The results indicate that persons can be detected with a precision in position of about 30 cm or less, which is sufficient for discrimination unless they stand very close to one another. In [44], multiperson tracking and identification results for the CLEAR 2006 campaign are reported. The MOTP values of the

Table 4 Identification performance

	Visual			Visual+Acoustic		
Data sets	Matches	Misses	FP	Matches	Misses	FP
TRA	81.4%	18.6%	0.0%	83.8%	16.2%	4.2%
TEST	49.3%	50.7%	3.0%	64.5%	35.5%	1.0%

submitted systems are between $195mm - 233mm$, whereas the MOTA values vary between $4.33\% - 62.20\%$[2]. MOTA will be equal to 100% only if there are no false positives, no misses and no identity mismatches in the tracking. For the reported results, the MOTA loss is mainly due misses, whereas the percentage of false positives is very low.

For multimodal identification, the scoring must take into account the position of the identified target. An identification hypotesis is considered a correct match if the identity agrees with the ground truth and the detected position of the identified target is within a certain distance (i.e. 50cm) to the actual position of the individual. We also report misses and false positives.

Table 4 shows the results for identification, for both training and test recordings. Results are similar to the ones obtained for tracking, with very good performance on the training set and a passable performance for the test set. The improvement obtained by using the acoustic modality is much more obvious in the more difficult test set. The presence of tracking (as opposed to frame by frame processing) ensures that the number of false positives are small.

6 Conclusions and Future Work

In this paper we have evaluated several methods for monitoring a room for the purposes of locating and identifying a set of individuals. We worked with different modalities from multiple sensors to observe a single environment and to generate multimodal data streams. These streams are processed with the help of a generic client-server middleware called SmartFlow and signal processing modules. The system is designed to operate in a completely automatic fashion, there is no manual segmentation or user intervention.

Modules for visual motion detection, visual face tracking, visual face identification, visual feature-based identification, audio-based localization, and audio-based identification were implemented. Our proposed system is based on multi-tier information processing, where available modalities are fused into a final decision through a particle-filter based tracker. The simple tracking and identification system that is based on a probabilistic occupancy map and on-the-fly colour feature modeling works effectively, as demonstrated by its stand-alone accuracy. The presence of additional information from the auditory modality increases the accuracy.

The collected database consists of two relatively small audio-visual streams, but it represents a realistic application testbed with multiple persons interacting and moving around a room. Both the visual and auditory channels result in high identification rates, with some superiority of visual information over acoustic information.

The present work represents an initial step in a series of increasingly challenging research questions, including automatic recognition of a set of gestures and pose estimation. One obvious future direction is to work with more difficult datasets (e.g. the CLEAR 2006

[2] These values are obtained on different experimental conditions, and only reported to give an idea of the expected range for MOTA and MOTP metrics.

and 2007 datasets). The testing of the proposed methods on these proprietory benchmark datasets is planned. The components that make up the system are not computationally intensive, and a real-time implementation of the approach is currently under study.

Acknowledgements Currently suppressed in accordance with blind reviewing conditions.

References

1. European union, 6^{th} framework integrated project CHIL URL http://chil.server.de
2. NIST SmartFlow system URL http://www.nist.gov/smartspace/nsfs.html
3. Proceedings of the DARPA/NIST smart spaces workshop. National Institute of Standards and Technology **3**, 1–14 (1998)
4. Adami, A., Burget, L., Dupont, S., Garudadri, H., Grezl, F., Hermansky, H., Jain, P., Kajarekar, S., Morgan, N., Sivadas, S.: Qualcomm-ICSI-OGI Features for ASR. Proc. ICSLP pp. 21–24 (2002)
5. Ajmera, J., McCowan, I., Bourlard, H.: Robust HMM-based speech/music segmentation. Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing **1** (2002)
6. Anguera, X.: Beamformit: The robust acoustic beamforming toolkit (2005). URL http://www.icsi.berkeley.edu/ xanguera/beamformit
7. Anguera, X., Wooters, C., Hernando, J.: Robust speaker diarization for meetings: ICSI RT06s evaluation system. Proc. ICSLP (2006)
8. Barras, C., Zhu, X., Meignier, S., Gauvain, J.: Improving Speaker Diarization. RT-04F Workshop (2004)
9. Bernardin, K., Elbs, A., Stiefelhagen, R.: Multiple object tracking performance metrics and evaluation in a smart room environment. IEEE Int. Workshop on Vision Algorithms pp. 53–68 (2006)
10. Bimbot, F., Bonastre, J.F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., Reynolds, D.: A tutorial of text-independent speaker verification. EURASIP Journal on Applied Signal Processing **4**, 430–451 (2004)
11. Black, J., Ellis, T., Rosin, P.: Multi-view image surveillance and tracking. IEEE Workshop on Motion and Video Computing (2002)
12. Carpenter, J., Clifford, P., Fearnhead, P.: Improved particle filter for nonlinear problems. IEE Proceedings Radar, Sonar and Navigation **146**(1), 2–7 (1999)
13. Casas, J., Stiefelhagen, R.: Multi-camera/multi-microphone system design for continuous room monitoring. In: CHIL Consortium Deliverable D4.1 (2005)
14. Checka, N., Wilson, K., Siracusa, M., Darrell, T.: Multiple person and speaker activity tracking with a particle filter. Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on **5** (2004)
15. Chen, J., Huang, N., Benesty, J.: An adaptive blind SIMO identification approach to joint multichannel time delay estimation. In: Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. iv–53–iv–56 (2004)
16. Davis, S., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. ASSP (28), 357–366 (1980)
17. DiBiase, J., Silverman, H., Brandstein, M.: Microphone Arrays. Robust Localization in Reverberant Rooms. Springer Verlag (2001)
18. Fleuret, F., Berclaz, J., Lengagne, R., Fua, P.: Multi-camera people tracking with a probabilistic occupancy map. to appear in IEEE: Trans. Pattern Analysis and Machine Intelligence
19. Fung, G., Mangasarian, O.: Proximal support vector machine classifiers. Proc. KDDM pp. 77–86 (2001)
20. Gatica-Perez, D., Lathoud, G., Odobez, J.M., McCowan, I.: Audiovisual probabilistic tracking of multiple speakers in meetings. IEEE Trans. Audio, Speech and Language Processing **15**(2), 601–616 (2007)
21. Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F Radar and Signal Processing **140**(2), 107–113 (1993)
22. Haritaoğlu, S., Harwood, D., Davis, L.: W4: Real-time surveillance of people and their activities. IEEE Trans. Pattern Analysis and Machine Intelligence **22(8)**, 809–830 (2000)
23. Isard, M., Blake, A.: CONDENSATION–conditional density propagation for visual tracking. International Journal of Computer Vision **29**(1), 5–28 (1998)
24. Kang, J., Cohen, I., Medioni, G.: Tracking people in crowded scenes across multiple cameras. Asian Conference on Computer Vision (2004)
25. Katsarakis, N., Souretis, G., Talantzis, F., Pnevmatikakis, A., Polymenakos, L.: 3D Audiovisual Person Tracking Using Kalman Filtering and Information Theory. LNCS **4122**, 45 (2007)

26. Khalaf, R.Y., Intille, S.S.: Improving multiple people tracking using temporal consistency. MIT Dept. of Architecture, House_n Project Technical Report (2001)
27. Khan, Z., Balch, T., Dellaert, F.: Efficient particle filter-based tracking of multiple interacting targets using an MRF-based motion model. Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems **1**, 254–259 (2003)
28. Kirby, M., L.Sirovich: Application of the Karhunen-Loeve procedure for the characterization of human faces. IEEE Trans. Pattern Analysis and Machine Intelligence **12**(1), 103–108 (1990)
29. Luque, J., Anguera, X., Temko, A., Hernando, J.: Speaker Diarization for Conference Room: The UPC RT07s Evaluation System. Proc. CLEAR, LNCS (2007)
30. Luque, J., Morros, R., Garde, A., Anguita, J., Farrus, M., Macho, D., Marqués, F., Martínez, C., Vilaplana, V., Hernando, J.: Audio, video and multimodal person identification in a smart room. Proc. CLEAR 2006, LNCS 4122 (2006)
31. Mittal, A., Davis, L.: M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. International Journal of Computer Vision, 51(3):189-203 (2003)
32. Moraru, D., Ben, M., Gravier, G.: Experiments on Speaker Tracking and Segmentation in Radio Broadcast News. Ninth European Conference on Speech Communication and Technology (2005)
33. Mostefa, D. et al.: CLEAR evaluation plan v1.1. In: http://isl.ira.uka.de/ nickel/clear/downloads/chil-clear-v1.1-2006-02-21.pdf (2006)
34. Nickel, K., Gehrig, T., Stiefelhagen, R., McDonough, J.: A joint particle filter for audio-visual speaker tracking. Proceedings of the 7th International Conference on Multimodal Interfaces pp. 61–68 (2005)
35. Omologo, M., Svaizer, P.: Use of the crosspower-spectrum phase in acoustic event location. IEEE Trans. on Speech and Audio Processing **5**(3), 288–292 (1997)
36. Potamitis, I., Tremoulis, G., Fakotakis, N.: Multi-speaker DOA tracking using interactive multiple models and probabilistic data association. In: Proceedings of European Conference on Speech Communication and Technology (2003)
37. Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE **17**(2) (1989)
38. Rabinkin, D.: A framework for speech source localization using sensor arrays. Ph.D. thesis, Brown University (1995)
39. Reynolds, D., Torres-Carrasquillo, P.: Approaches and Applications of Audio Diarization. IEEE Int. Conf. Acoustics, Speech, and Signal Processing **5** (2005)
40. Salah, A., Alpaydın, E.: Incremental mixtures of factor analyzers. Int. Conf. on Pattern Recognition **1**, 276–279 (2004)
41. Schölkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambridge, MA (2002)
42. Stanford, V., Garofolo, J., Galibert, O., Michel, M., Laprun, C.: The NIST smart space and meeting room projects: Signals, acquisition, annotation, and metrics. Proc. ICCASP **4**, 736–739 (2003)
43. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition (1999)
44. Stiefelhagen, R., Bernardin, K., Bowers, R., Garofolo, J., Mostefa, D., Soundararajan, P.: The CLEAR 2006 Evaluation. Proc. CLEAR, LNCS pp. 1–44 (2007)
45. Szeder, G., Tichy, W.: A Communication Middleware for Smart Room Environments. Proc. European Conference on Ambient Intelligence, LNCS 4794 pp. 195–210 (November 2007)
46. Tangelder, J., Schouten, B.: Sparse face representations for face recognition in smart environments. International Conference on Pattern Recognition (2006)
47. Temko, A., Macho, D., Nadeu, C.: Enhanced SVM training for robust speech activity detection. Proc. ICCASP (2007)
48. Vilaplana, V., Martínez, C., Cruz, J., Marques, F.: Face recognition using groups of images in smart room scenarios. International Conference on Image Processing (ICIP'06) (2006)
49. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. Proc. IEEE Conf. Computer Vision and Pattern Recognition **1**, 511–518 (2001)
50. Wei Niu Long Jiao, D.H., Wang, Y.F.: Real time multi person tracking in video surveillance. Pacific Rim Multimedia Conference, Singapore (2003)
51. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: Real-time tracking of the human body. IEEE Trans. Pattern Analysis and Machine Intelligence **19**(7), 780–785 (1997)
52. Zhao, T., Nevatia, R., Wu, B.: Segmentation and tracking of multiple humans in crowded environments. to appear in IEEE: Trans. Pattern Analysis and Machine Intelligence
53. Zhou, S., Krueger, V., Chellappa, R.: Probabilistic recognition of human faces from video. Computer Vision and Image Understanding **91**(1), 214–245 (2003)
54. Zotkin, D., Duraiswami, R., Davis, L.: Multimodal 3-d tracking and event detection via the particle filter. IEEE Workshop on Detection and Recognition of Events in Video pp. 20–27 (2001)
55. Zotkin, D., Duraiswami, R., Davis, L.: Joint Audio-Visual Tracking Using Particle Filters. EURASIP Journal on Applied Signal Processing **2002**(11), 1154–1164 (2002)