Multi-modal Fusion Framework with Particle Filter for Speaker Tracking

Anwar Saeed, Ayoub Al-Hamadi, and Michael Heuer
Institute for Electronics, Signal Processing and Communications (IESK)
Otto-von-Guericke-University Magdeburg
D-39106 Magdeburg, P.O. Box 4210 Germany
{Anwar.Saeed, Ayoub.Al-Hamadi}@ovgu.de

Abstract

In the domain of Human-Computer Interaction (HCI), the main focus of the computer is to interpret the external stimuli provided by users. Moreover in the multi-person scenarios, it is important to localize and track the speaker. To solve this issue, we introduce here a framework by which multi-modal sensory data can be efficiently and meaningfully combined in the application of speaker tracking. This framework fuses together four different observation types taken from multi-modal sensors. The advantages of this fusion are that weak sensory data from either modality can be reinforced, and the presence of noise can be reduced. We propose a method of combining these modalities by employing a particle filter. This method offers satisfied real-time performance. We demonstrate results of a speaker localization in two- and three-person scenarios.

***Keywords**: Speaker tracking, Human skin detection, Face detection, Particle filter, Time difference of arrival.*

1: Introduction

This work represents an example of using multiple modalities within a particle filter. We describe a particle filter by which multi-modal sensory data are fused together to track a speaker in a scene. For designing a particle filter for an application, it is necessary to introduce feedback to the approach. This feedback should describe the scene. In the case of speaker tracking, we have two types of sensory input: video and audio input devices. Image processing methods operate on image sequence captured from the video input to detect the human faces and other approaches use the audio input for the speaker localization.

Tracking the speaker using audio-visual information is an active research topic in the computer vision due to its importance to various applications such as smart video-conferencing and surveillance security systems. Shivappa et al. [1] explored different strategies for audio-visual fusion. Vermaak et al. [2] described a method in which a standard contour tracking algorithm, consisting of an edge detector and a particle filter, is used in conjunction with a Time Difference of Arrival (TDOA) calculation to deduce the speaker location from auditory data received by a pair of microphones. In their approach, the audio data are used for

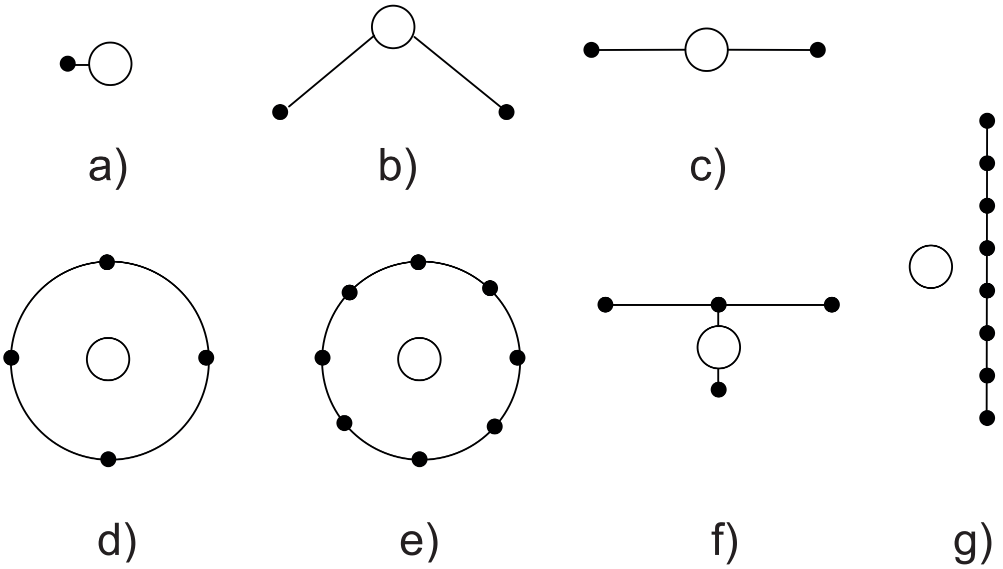

Figure 1. Samples of sensor configurations for audio-visual speaker tracking (filled circles indicate microphones; empty circles indicate cameras): (a) single microphone-camera pair; (b-c) Stereo Audio and Cycloptic Vision (STAC) sensors; (d-e) circular microphone array with single camera; (f) triangular microphone array with single camera; (g) linear microphone array with single camera [3].

initialization and video data for localization in an attempt to utilize the strengths of each modality. This method enhances the existing visual tracking successfully and can detect speaker 'ping-pong'. However, this implementation is not a real-time solution. Zhou et al. [3] employed a histogram matching based technique for image based speaker detection and a TDOA algorithm once again for audio localization. The audio undergoes pre-processing to remove noise and a Kalman filter is used to further reduce spurious detections before both audio and video observations are passed to a Weighted Probabilistic Data Association (WPDA) filter for fusion and tracking. The aforementioned approach fused different types of sensory data; however, it did not provide a real-time level of performance. In this paper, we address the speaker tracking with the help of audio-visual sensory data in real-time level.

The audio-visual sensors number and configuration play a major role in the performance of the speaker tracking approach. Various configurations were proposed, as shown in Figure 1. A very simple configuration is composed of a single camera-microphone pair and the complex one composed of stereo cameras with stereo, circular arrays of microphones. Cutler et al. [4] used a single camera-microphone pair to automatically detect a talking person (both spatially and temporally). Kapralos et al. [5] used a panoramic visual sensor, which captures a 360^o view of the speakers environment, along with a directional audio system based on beamforming that used to confirm potential speakers. Multi-camera with eight microphones are used by Gatica-perez et al. [6] to track the speaker in meeting scenarios.

The remainder of this paper is structured as follows. In section 2, we describe the proposed approach. In this section, we present three main parts of the approach: the video modality (2.1), audio modality (2.2), and an overview of the particle filter implementation (2.3). Experimental results are discussed in section 3. Finally, the conclusion and future perspectives are given in section 4.

2: The Proposed Approach

We employ the STAC sensor configuration (Figure 1.c) to solve the speaker tracking and detection issue. This configuration provides two modalities: video modality from the mono-camera and audio modality from the two microphones. We extract two observation data types from the video modality. The first one represents the human skin blob that fits the human face shape. The second observation type is the output of a human face detector, in which texture features are employed. Additionally, two observation types are extracted from the audio modality: the first one utilizes the TDOA of an audio signal at two microphones, and the second one uses Received Signal Strength (RSS) at the two microphones to estimate the speaker location. These four observation types from the two modalities will be combined in a useful and meaningful fashion using a particle filter and then used to detect and track the speaker in the scene.

2.1: Video Modality

As mentioned previously, the used sensor configuration is a mono-camera centered between two microphones. In the video modality, we segment the human faces using two methods. The first one utilizes the human skin color, while the second approach uses texture features.

2.1.1: Face Detection using Skin Color

Several approaches were proposed to classify each pixel as skin or non-skin [7, 8, 9]. Two different approaches were investigated in this work. In the first approach, we used the multivariate Gaussian mixture model (GMMs). We built two GMMs for skin and non-skin pixels. The two models were built of the CrCb channels from the YCrCb color space, where Y is the luminance component, and both Cb and Cr represent the chrominance (color difference). Ignoring Y channel makes the models insensitive to the luminance variation. The probability that pixel $\mathbf{p}$ with color $\mathbf{p} = [cb \;\; cr]^{\mathrm{T}}$ is skin would be given by

$$\begin{aligned} p(\mathbf{p}|S) &= \sum_{i=1}^{m} \gamma_i p(\mathbf{p}|S_i) \\ &= \sum_{i=1}^{m} \frac{\gamma_i}{2\pi|\Sigma_{S_i}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{p}-\mu_{S_i})^{\mathrm{T}}\Sigma_{S_i}^{-1}(\mathbf{p}-\mu_{S_i})\right). \qquad (1) \end{aligned}$$

m Gaussian Models S_i, each characterized by its mean vector μ_{S_i}, covariance matrix Σ_{S_i}, and mixture weight γ_i, where

$$\sum_{i=0}^{m} \gamma_i = 1, \qquad \gamma_i > 0.$$

Similarly, another GMM model was built to calculate the probability that pixel $\mathbf{p}$ is non-skin $p(\mathbf{p}|\text{non}-S)$. Then, each pixel is classified as skin if it satisfies

$$\frac{p(\mathbf{p}|S)}{p(\mathbf{p}|\text{non}-S)} > \tau_1, \qquad\qquad \tau_1 \in \mathbb{R}.$$

The parameters of the GMMs are computed using the expectation-maximization (EM) algorithm through training with the help of collection of skin and non-skin images. τ_1 is an empirically determined threshold. However, this method is time-consuming in contrast to other methods that compare the pixel value with pre-learnt threshold values. This comparison could be carried out in different color spaces such as RGB, HSL, HSV, YCrCb, etc. [8, 9]. In the second approach, we used a combination of threshold filters in HSV and YCrCb to segment the skin pixels. The threshold values define boundaries for the pre-learnt skin color. In this work, we chose the second approach of skin segmentation technique due to its efficiency and to meet our requirements for building real-time speaker tracking approach. We opted for the chrominance channels (cr, cb) from YCrCb color space and the Hue channel (h) from HSV color space [9]. Each pixel $\mathbf{p}(h, cr, cb)$ is classified as follows.

$$\mathbf{p}(h, cr, cb) = \begin{cases} 1 & HSV(h) \wedge CbCr(cr, cb) \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where $HSV(h)$ represents the skin segmentation in HSV color space. The Hue component is proven to be a good discriminator for the human skin tone. $HSV(h)$ is calculated by

$$HSV(h) = (h < 25) \vee (h > 230).$$

$CbCr(cr, cb)$ represents the skin segmentation in YCrCb color space. The luminance component (Y) is ignored to have skin detection immune to the luminance variation. $CbCr(cr, cb)$ is calculated by

$$CbCr(cr, cb) = \left\{ \begin{array}{l} (cr \leq 1.5862 \times cb + 20) \wedge \\ (cr \geq 0.3448 \times cb + 76.2069) \wedge \\ (cr \geq -4.5652 \times cb + 234.5652) \wedge \\ (cr \leq -1.15 \times cb + 301.75) \wedge \\ (cr \leq -2.2857 \times cb + 432.85) \end{array} \right\}.$$

At the end of the skin pixel classification, we operate morphology operations to remove the outlier skin detection and to close misdetected pixels. The resulting skin regions are then examined for contours which may describe faces in the scene. The contours which do not fit the facial shape characteristics will be discarded.

2.1.2: Face Detection using Texture Features

The human face detection based on the skin color is error-prone due to cluttered environments and due to the existence of objects that are human skin colored with the same face shape. Hence, we enhance the face detection by the use of texture features. This detection is more accurate; however, it detects the face only in a frontal upright pose with $\pm 20^o$ head rotation angles (yaw, pitch, and roll). Thus, the two methods of face detection will complement each other using the particle filter framework. We assign the detection using texture features more weight than that using skin color. We pass the same video frame,

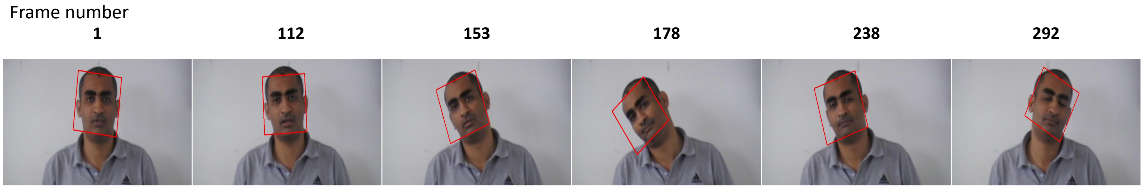

Figure 2. Coping with in-plane (roll) rotated faces.

which is processed by skin segmentation, into a well-trained Haarcascade classifier [10, 11]. This classifier utilizes the Haar-like features, which are defined as the ratio of intensities of adjacent rectangles of different locations [12]. We proposed a method to cope with the issue of detecting the face with in-plane rotation (roll angle) more than $\pm 20^o$. For each detected face, we estimate the in-plane rotation angle with the help of the eyes position. Where the eyes are detected using another Haar-like classifier. Then, we use the estimated roll angle to guide the face detector at the following frames. Figure 2 shows an image sequence in which we cope with the in-plane rotation.

2.2: Audio Modality

The audio data are processed by fast but rather imprecise RSS based location scheme, and by much slower but far more accurate TDOA algorithm. Different precision and confidence factors are assigned to the locations estimated by TDOA and RSS according to the algorithms accuracy and reliability.

2.2.1: Audio TDOA

It is possible to locate the origin of a sound source by observing the TDOA of a signal at audio sensors placed in differing locations. The number of the used audio sensors determines the accuracy, dimensionality, as well as the performance of the system. The processing using TDOA method is relatively costly, increasing exponentially with sensor count. To maintain real-time levels of performance, only two audio sensors are used in this implementation. This limits the output of the audio modality to a single-dimension location estimate (x-coordinate). However, as this dimension has the most variance, the chosen configuration gave an excellent balance of performance versus contribution. The algorithm used to estimate (TDOA) exploits the fact that the sound arriving at each microphone will be delayed according to the speaker position, given that the position of the microphones is known. The audio data from the two microphones are stored in two different buffers of size M, as follows.

$$\mathbf{s}_i = \big(s_i[0], ..., s_i[M-1]\big), \qquad i = 1, 2. \tag{3}$$

Then, to find the time delay, a windowed cross-correlation of size N is used, where $M > 2N$. The discrete cross correlation function (CC) for a fixed window size of N is given by

$$CC(d) = \frac{1}{N}\sum_{n=0}^{N-1} s_1[n + \frac{M}{2}]s_2[n + \frac{M}{2} + d], \qquad d = \frac{-M}{2}, \cdots, \frac{M}{2} - N.$$

The sample delay d is used together with the sampling rate to deduce the time delay. Due to the nature of audio sensory information, multiple detections are possible from a single

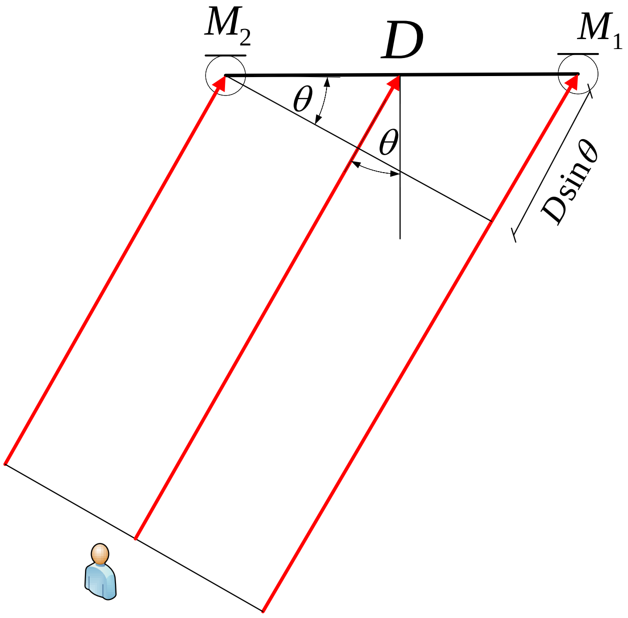

Figure 3. Audio-sensor geometry of the far-field. Two microphones M_1 and M_2 are separated by a distance D. The sound wave travels an additional distance of $D\sin\Theta$ to reach M_1. Θ denotes the arrival angle of the audio signal.

source. These ghost detections are the result of reverberations being interpreted falsely as signals originating directly from the source. To minimize the effect of these reverberations, we used a Generalized Cross Correlation function using a Phase Transform (GCC-PHAT), which was introduced by Knapp et al. [13]. This is a computationally expensive operation so the Discrete Fourier Transform (DFT) is used for efficiency. Given two signals $\mathbf{s}_1$ and $\mathbf{s}_2$, the weighted correlation function at time delay τ is

$$GCC_{PHAT}(\tau) = \mathbf{FFT}^{-1}\left(\frac{F_1(f)[F_2(f)]^*}{|F_1(f)[F_2(f)]^*|}\right), \tag{4}$$

where F_1 and F_2 are the Fourier transforms of $\mathbf{s}_1$ and $\mathbf{s}_2$, respectively. FFT^{-1} is the inverse Fourier Transform and $[\,]^*$ denotes the complex conjugate. Only the frames containing speech will be processed. To determine the speech frames, we used Signal to Noise Ratio (SNR) along with Zero Crossing Rate [14]. In order to simplify the speaker x-coordinate estimation from TDOA (τ), it is assumed that the speaker is far enough away to be considered as a "far-field" source. The result of this assumption is that the audio waves can be assumed to arrive at each microphone in a planar fashion rather than in circular waves, as shown in Figure 3. Then, with the knowledge of the speed of sound c, we can derive the arrival angle of the audio signal Θ as follows.

$$\tau = \frac{D\sin\Theta}{c}.$$

Where D denotes the distance between the two microphones. Finally, mapping Θ to a corresponding x-coordinate position is done using a calibration process. Multiple observations

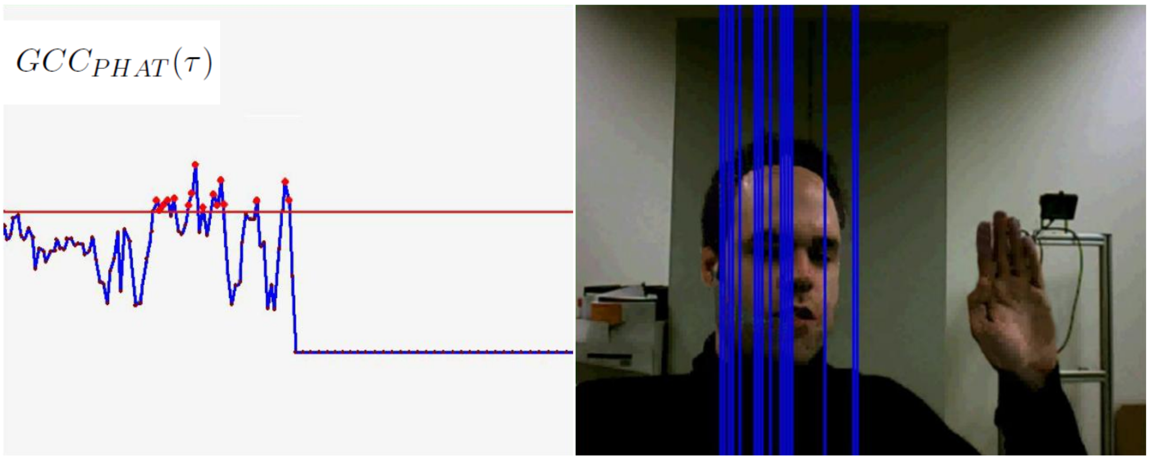

Figure 4. Mapping τ to a corresponding x-coordinate position. The local maxima value of $GCC_{PHAT}(\tau)$ (4) which is above an experimentally set threshold is mapped onto an estimate of the speaker position in x-coordinate.

of τ lead to multiple estimates of x-coordinate positions, as shown in Figure 4, where the vertical blue lines represent the estimated x-coordinate positions of the speaker. All these observations are used in the calculation of the particle filter likelihood function, as will be discussed in section 2.3.

2.2.2: Audio RSS

Locating the speaker using audio RSS is far simpler than the previously described TDOA routine. However, in most cases, it gives a decent approximation of the speaker position, which is not precise as TDOA. Consequently, this will be reflected in the particle filter parameters. Similar to TDOA, audio RSS is used to estimate the speaker position just in one dimension (x-coordinate). To measure audio RSS, the total energy is calculated for the signals contained in both left and right audio buffers. The energy of the signal contained in a buffer of size M (3) is defined as

$$e(\mathbf{s}) = \sum_{n=0}^{M-1} [s(n)]^2. \tag{5}$$

By comparing the energy of each buffer, we can arrive at x-coordinate representing the relative position of the speaker in the scene. In other words, $\frac{e(\mathbf{s}_1)}{e(\mathbf{s}_2)}$ is mapped onto an estimate of the speaker x-coordinate position through a calibration process.

While being less precise, a greater confidence is placed in this Audio RSS compared to TDOA. This is due to the fact that RSS is less prone to noise such as reverberations. The audio signals are continuously measured and buffered for (10ms-41ms) before using them for the location estimation using TDOA and RSS. Obviously, the buffers are cleared after each estimation, and the process iterates.

2.3: Particle Filter Implementation

Before discussing the fusion of multiple modalities, it is necessary to understand the means by which sensory data are incorporated into a particle filter. The sensory data are used in the measurement and selection stages within the filter. In the measurement stage, the particle set generated by the filter is examined and each particle is given a score, or weight, based on how accurately said the particle describes the scene. During the selection stage, the particle set is refined to accentuate those particles which best describe the scene, and new particles are introduced based on the current sensory data. The parameters of these steps vary on what the purpose of the filter is. In the case of the filter implementation given in this paper, it is intended that the particles should describe a bounding box relative to a camera snapshot showing where the current speaker is. In this case, each particle needs to be given some form of score based on how accurate the bounding box describes the real speaker face position. The location of this real speaker from the raw sensory data and the choice of what weight to assign to each particle is our concern.

The general particle filtering scheme in [15, 16] is used in this work. This scheme is particularly effective at tracking objects in substantial cluttered environments. Hence, it is very desirable when dealing with multiple modalities each potentially generates many observations and hence noise. The problem of speaker tracking can be formulated as a continuous estimation of speaker state $\mathbf{x}_t$ at each time t. The state forms a vector containing a union of relevant parameters obtained from all modalities of feature extractors. The state $\mathbf{x}_t$ is defined here as

$$\mathbf{x}_t = \big(x_t(1), x_t(2), x_t(3), x_t(4), x_t(5)\big).$$

$x_t(1), \ldots, x_t(5)$ denote x-position, y-position, head width, head height, and head orientation angle, respectively. Particle Filter, a Quasi-Monte Carlo solution, solves the tracking issue. Let state $\mathbf{x}_t$ in the Markov state-space model represent a possible speaker configuration at time t. And $\mathbf{y}_t$ is the observation obtained using the two modalities, as discussed in sections 2.1 and 2.2. Then, the distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ can be calculated by

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t) \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \mathrm{d}\mathbf{x}_{t-1} \qquad (6)$$

(6) consists of likelihood function $p(\mathbf{y}_t|\mathbf{x}_t)$ multiplied by an integral representing the prediction step. The particle filtering method used here is an approximation to (6). At each time t, there are N samples $\{\mathbf{x}_t^{(i)}, \quad i = 1, ..., N\}$ each associated with weight value $w_t^{(i)}$. To avoid the degeneracy problem caused in re-sampling, the weight value is calculated as follows [17].

$$w_t^{(i)} \propto p(\mathbf{y}_t|\mathbf{x}_t^{(i)}). \qquad (7)$$

In the particle prediction step, the state of each particle is altered according to an underlying temporal model, as given in (8).

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{n}_t. \qquad (8)$$

Where $\mathbf{A}$ and $\mathbf{B}$ are experimentally set parameters for the model. $\mathbf{n}$ is a normalized white Gaussian noise vector. Within the update step, we assign each sample $\mathbf{x}_t^{(i)}$ new weight

Modality	Parameter List	Precision Factor (σ^{-1})	Confidence Factor (γ)
Audio (TDOA)	$x(1)$	1.0	0.5
Audio (RSS)	$x(1)$	0.2	1.0
Video (Skin Color)	$x(i), i = 1, ..., 5$	0.5	0.5
Video (Texture)	$x(i), i = 1, ..., 5$	1.0	0.75

Table 1. The used modalities and their respective parameter definitions. The precision and confidence factors were arrived at as a result of experimentation.

value according to the observation likelihood as in (7). Where the observation likelihood function is defined as an averaged sum of likelihood functions of the particle components,

$$p(\mathbf{y}_t|\mathbf{x}_t) = \frac{1}{5}\sum_{i=1}^{5} p(\mathbf{y}_t|x_t(i)). \tag{9}$$

As we mentioned in sections 2.2 and 2.1, we have four different observation types. In addition, we could have multiple observations from each type. For example, more than one face could be detected and may be many locations could be estimated for the speaker in each observation type. The component $x(1)$ is fused from the four observation types, while the other four components are fused only from the video modality. Each component $x(i)$ has precision factors $\sigma_{i,m}^{-1}$ and confidence factors $\gamma_{i,m}$ reflecting the observation type m accuracy and reliability, respectively, as shown in Table 1. Let us have Z observations for the component $x_t(i)$. These observations are denoted by $(y_{t,1}^m(i), \ldots, y_{t,Z}^m(i))$, where m denotes the observation type. Then, we formulate the observation likelihood function for each component as GMM given by

$$p(\mathbf{y}_t|x_t(i)) = K\sum_{o=1}^{Z} \frac{\gamma_{i,m}}{\sqrt{2\pi\sigma_{i,m}^2}} \exp\left(-\frac{\left(y_{t,o}^m(i) - x_t(i)\right)^2}{2\sigma_{i,m}^2}\right), \tag{10}$$

where K is a normalization factor, which depends on the observation number Z. Obviously, K is the reciprocal of the sum of the confidence factors $\gamma_{i,m}$ for all observations. Finally, the density distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ in (6) is approximated by a sum of N Dirac functions as follows.

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N} w^{(i)}\delta(\mathbf{x}_t - \mathbf{x}_t^{(i)}).$$

3: Experimental Results

The performance of the particle filter is more than sufficient for real-time operation. This filter is employing the four input data, as described in section 2. An internal timer of the application iterates the filter every 10ms, which is enough for most iterations; however, when a slowdown occurs due to the presence of more observations, it has never been significant enough to go beyond the 24 frames per second (41ms per iteration) limit. The main observable distinction between the multi-modal approach as opposed to single modal one is that the system is much more robust when dealing with noise or non detection. For example, the observations from the texture-based face detector are not always present due to the orientation of speaker face in the scene; however, both the audio and skin color modalities allow tracking to continue. Conversely, over abundance of observations generated by the skin color segmentation is complemented by the accuracy texture-based face detection meaning that the tracker does not get distracted by the multiple false positives. We demonstrate two experiments in this work. In the first experiment, we use just two inputs: the face detector using skin segmentation from the video modality and the TDOA from the audio modality. Figure 5 shows captured images from a conversation of two persons,

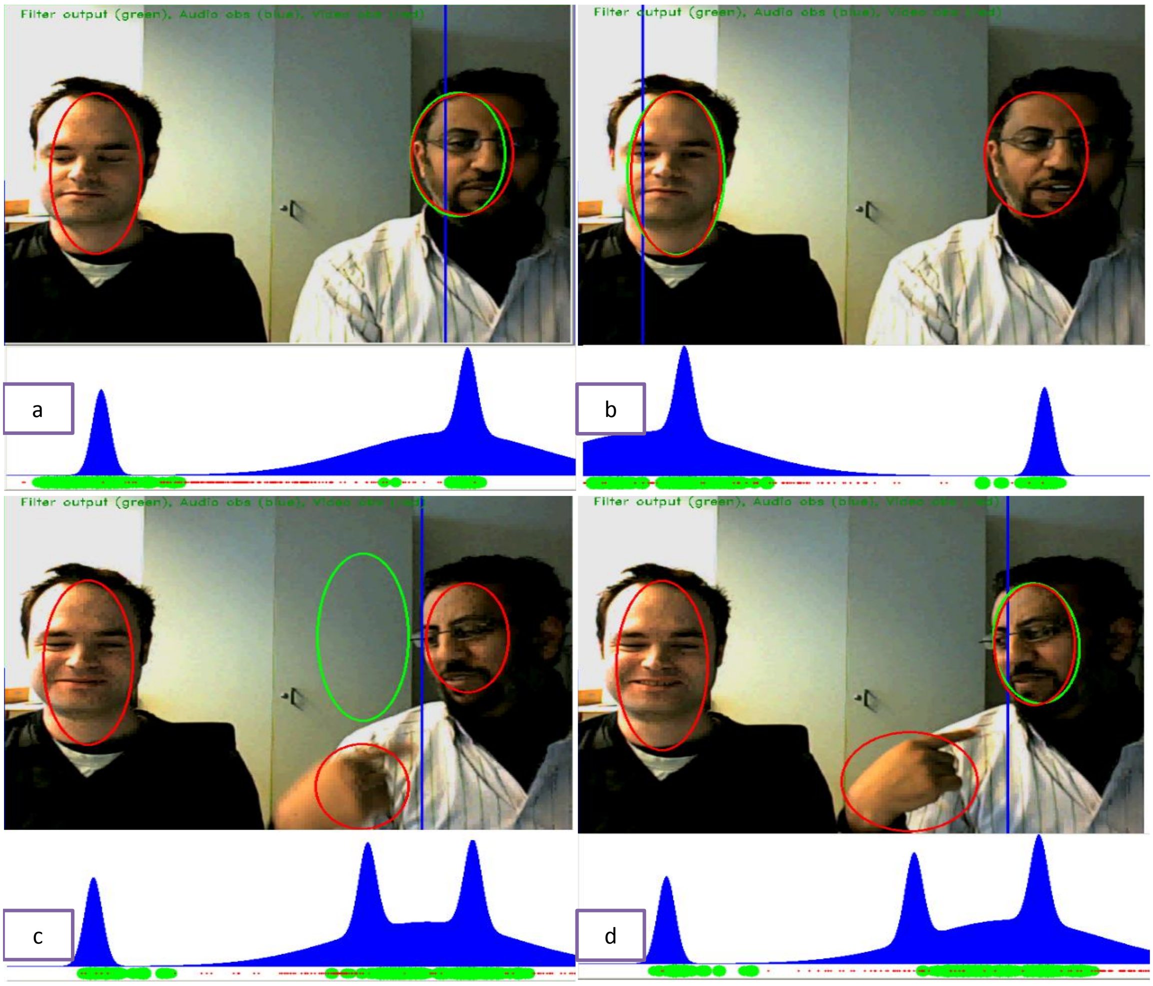

Figure 5. The speaker tracking in a conversation of two persons. The red ellipse encloses the detected face using skin color, and the blue line represents the estimated speaker position in x-coordinate. Below each image is the probability density distributions for the x-component given the measurements shown. The speaker was tracked and detected correctly in (a,b,d). In (c), the particle filter fails to detect the speaker because an inaccurate estimation using the audio signal and a false face detection using skin color.

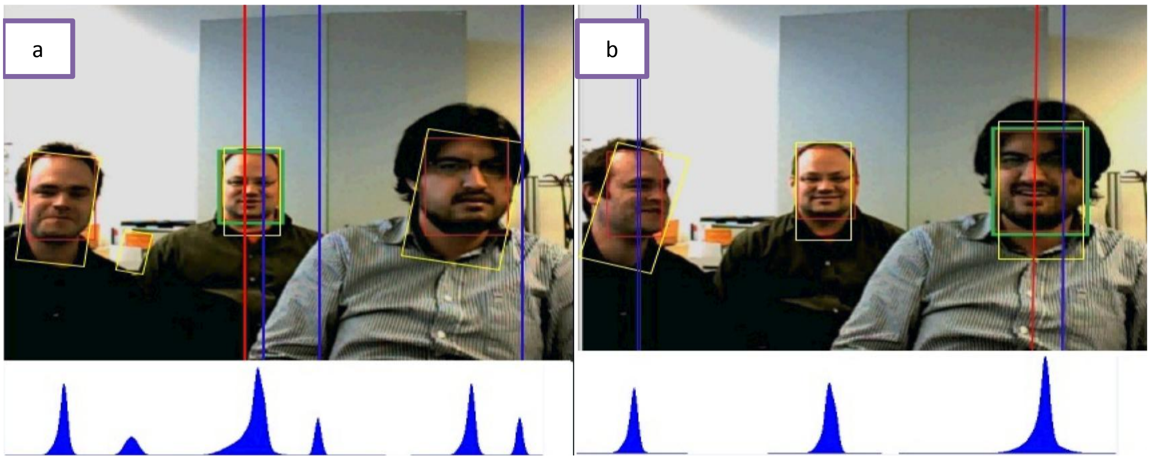

Figure 6. The speaker tracking in a conversation of three persons. The red and yellow rectangles show measurements from the video modality and the vertical blue and red lines from the audio modality. Below each image is the probability density distributions for the x-component given the measurements shown. The green rectangle is the filter output corresponding to the greatest peak in the probability distribution. In (a), the center person is speaking whereas in (b) the right person is doing the majority of the speaking with some sound coming from the left.

where the red ellipse encloses the detected face using skin color, the blue line indicates the estimated position (in x-coordinate) of the speaker using the audio TDOA, and the green ellipse represents the speaker state as the results of the particle filter. The probability density distribution of the x-component is given below each image. In Figures 5.a and 5.b, the speaker was detected accurately. In Figures 5.c and 5.d, the hand is detected as a face. Consequently, the particle filter is prone to be misguided, especially when the estimate from the audio input is inaccurate as in Figure 5.c. The second experiment is done with the help of the four proposed inputs in a conversation of three persons. Figure 6 shows two samples of this experiment. The two images are overlaid with data extracted from the audio and video modalities. The speaker was detected and tracked successfully. Integrating the four inputs using the particle filter makes the speaker tracking more robust to the false positive detection by the video modality and the inaccurate estimation by the audio modality.

4: Conclusions and Future Work

We have shown an approach for speaker tracking. Instead of using a single estimate approach (as Kalman filter), we employed the particle filter, which can track a multimodal condition probability. We combined four observation types from two modalities (audio-visual). Each observation could not be used alone; however, when we combined them within a particle filter, we achieved a robust performance. The observations are modeled by GMMs with experimentally set parameters. Our proposed approach is able to perform in real-time on standard workstation. In future, we will consider adding more microphones. Hence, we will be able to estimate the speaker location in y-coordinate as well.

Acknowledgments

This work is supported by Transregional Collaborative Research Centre SFB/TRR 62 "Companion-Technology for Cognitive Technical Systems" funded by DFG and OvG-University Magdeburg.

References

[1] S.T. Shivappa, M.M. Trivedi, and B.D. Rao, "Audiovisual information fusion in human computer interfaces and intelligent environments: A survey," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1692 –1715, oct. 2010.

[2] J. Vermaak, M. Gangnet, A. Blake, and P. Perez, "Sequential monte carlo fusion of sound and vision for speaker tracking," in *ICCV*, 2001, pp. 741–746.

[3] Huiyu Zhou, M. Taj, and A. Cavallaro, "Target detection and tracking with heterogeneous sensors," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 4, pp. 503 –513, aug. 2008.

[4] Ross Cutler and Larry Davis, "Look who's talking: Speaker detection using video and audio correlation," in *in IEEE International Conference on Multimedia and Expo*, 2000, pp. 1589–1592.

[5] Bill Kapralos, Michael R. M. Jenkin, and Evangelos Milios, "Audio-visual localization of multiple speakers in a video teleconferencing setting," 2003.

[6] Daniel Gatica-perez, Guillaume Lathoud, Jean marc Odobez, and Iain Mccowan, "Audio-visual probabilistic tracking of multiple speakers," in *in Meetings,? IEEE Trans. on Audio, Speech, and Language Processing*, 2007, pp. 601–616.

[7] A. Saeed, R. Niese, A. Al-Hamadi, and B. Michaelis, "Coping with hand-hand overlapping in bimanual movements," in *Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on*, nov. 2011, pp. 238 –243.

[8] R. Schettini and F. Gasparini, "Skin segmentation using multiple thresholding," *Internet Imaging VII, IS and T/SPIE (pp.60610F-1-60610F-8). SPIE.*, 2006.

[9] Nusirwan A. Rahman, Kit C. Wei, and John See, "RGB-H-CbCr Skin Colour Model for Human Face Detection," in *Proceedings of The MMU International Symposium on Information & Communications Technologies (M2USIC 2006)*, 2006.

[10] Anwar Saeed, Robert Niese, Ayoub Al-Hamadi, and Axel Panning, "Hand-face-touch measure: a cue for human behavior analysis," in *Intelligent Computing and Intelligent Systems (ICIS), 2011 IEEE International Conference on*, 2011, vol. 3, pp. 605–609.

[11] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[12] Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features," 2001, pp. 511–518.

[13] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, pp. 320–327, 1976.

[14] Bachu R G, Kopparthi S, Adapa B, and Barkana B D, "Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal," *American Society for Engineering Education ASEE Zone Conference Proceedings*, pp. 1–7, 2008.

[15] Andrew Blake and Michael Isard, "The CONDENSATION algorithm - conditional density propagation and applications to visual tracking," in *NIPS*, 1996, pp. 361–367.

[16] M.A. Steer, A. Al-Hamadi, and B. Michaelis, "Audio-visual data fusion using a particle filter in the application of face recognition," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, aug. 2010, pp. 4392 –4395.

[17] Arnaud Doucet, Nando De Freitas, and Neil Gordon, Eds., *Sequential Monte Carlo methods in practice*, 2001.