An Audio-visual Particle Filter for Speaker Tracking on the CLEAR'06 Evaluation Dataset

– Draft Version –

Kai Nickel, Tobias Gehrig, Hazim K. Ekenel, John McDonough, Rainer Stiefelhagen

Interactive Systems Labs - University of Karlsruhe
Am Fasanengarten 5, 76131 Karlsruhe, Germany
nickel@ira.uka.de

Abstract. We present an approach for tracking a lecturer during the course of his speech. We use features from multiple cameras and microphones, and process them in a joint particle filter framework. The filter performs sampled projections of 3D location hypotheses and scores them using features from both audio and video. On the video side, the features are based on foreground segmentation, multi-view face detection and upper body detection. On the audio side, the time delays of arrival between pairs of microphones are estimated with a generalized cross correlation function. In the CLEAR'06 evaluation, the system yielded a tracking accuracy (MOTA) of 71% for video-only, 55% for audio-only and 90% for combined audio-visual tracking.

1 Introduction

Person tracking is a basic technology for realizing context-aware human-computer interaction applications. The scenario addressed in this work is a smart lecture room, where information about the lecturer's location helps to automatically create an audio-visual log of the presentation. As we have shown in [18], tracking accuracy has an direct impact on the recognition rate of beamformed speech. Other applications include active camera control in order to supply high-resolution images of the speaker, thus facilitating person identification and audio-visual speech recognition.

The task of lecturer tracking poses two basic problems: localizing the lecturer (in terms of 3D head coordinates) and disambiguating the lecturer from other people in the room. In the proposed approach, we jointly process images from multiple cameras and the signal from multiple microphones in order to track the lecturer both visually and acoustically. The algorithm is based on the assumption, that the lecturer - among all other people in the room - is the one that is speaking and moving most of the time, i.e. exhibiting the highest visual and acoustical activity.

The central issue in audio-visual tracking is the question of how to combine different sensor streams in a beneficial way. In our approach, we integrate audio

and video features such that the system does not rely on a single sensor or a certain combination of sensors to work properly. In fact, each single camera and each microphone pair alone can contribute to the track. The core of the proposed algorithm is a particle filter for the computationally efficient integration of acoustic source localization, person detection (frontal face, profile face, upper body) and foreground segmentation. The 3D position of the lecturer is robustly being determined by means of sampled projection instead of triangulation.

1.1 Related Work

For acoustic source localization, several authors have proposed solving this optimization problem with standard gradient based iterative techniques. While such techniques typically yield accurate location estimates, they are typically computationally intensive and thus ill-suited for real-time implementation [2, 3]. Other recent work on acoustic source localization includes that by Huang *et al* [7], who developed an iterative technique based on a spherical least square error criterion, that is nonetheless suitable for real-time implementation, as well as the work by Ward *et al* [17], who proposed using particle filter together with both time delay of arrival estimation and steered beamformers. In other work by the same authors [10], a variant of the extended Kalman filter was used for acoustic speaker tracking. This approach was extended in [9] to add video features.

Particle filters [8] have previously been used for audio-visual tracking for example by [15] for a video telephony application, by [4] for multi-person tracking or by [6] for multi-party conversation in a meeting situation. The particle filter's capability of representing arbitrary distributions is of central importance for the proposed feature fusion scheme.

Concerning video features, it has often be proposed to use color models for the task of tracking articulated objects like the human body. Unfortunately, the appearance of color in real-world scenarios is fragile because of different light sources, shadowing, and – specific for our lecture scenario – by the bright and colorful beam of the video projector that often overlays the lecturer.

Color-invariant approaches that rely on background subtraction, e.g. Mikic et al. [13], often suffer from over- or under-segmentation as an effect of noisy foreground classification. In the proposed method, we avoid this problem: instead of triangulating connected foreground segments, our algorithm performs sampled projections of 3D hypotheses, as proposed by Zotkin et al. [19], and gathers support for the respective sample in the resulting image region in each view. It is thus less dependent on the quality of the segmentation.

Face-detection cascades as proposed by Viola and Jones [16] are known to be both robust and fast, which makes them a good feature to support a person tracker. However, searching high-resolution camera images exhaustively for faces in multiple scales goes beyond the current possibilities of real-time operation. The particles, however, cluster around likely target positions and are thus a good approximation of the search space.

2 An Audio-visual Particle Filter

Particle filters [8] represent a generally unknown probability density function by a set of m random samples $s_{1..m}$. Each of these particles is a vector in state space and is associated with an individual weight π_i. The evolution of the particle set is a two-stage process which is guided by the observation and the motion model:

1. The prediction step: From the set of particles from the previous time instance, an equal number of new particles is generated. In order to generate a new particle, a particle of the old set is selected randomly in consideration of its weight, and then propagated by applying the motion model.
2. The measurement step: In this step, the weights of the new particles are adjusted with respect to the current observation z_t: $\pi_i = p(z_t|s_i)$. This means computing the probability of the observation given that the state of particle s_i is the true state of the system.

Each particle $s_i = (x, y, z)$ hypothesizes the location of the lectures head centroid in 3D space. The particles are propagated by Gaussian diffusion, i.e. a 0-th order motion model. If a particle leaves the boundaries of lecture room, it gets re-initialized to a random position within the room. A certain percentage of particles (in our case 5%) are not drawn from the previous particle set, but are also initialized randomly. This way it is guaranteed, that the entire space is roughly searched - and the tracker does not stick to a local maximum.

Using the features described in Sections 3 and 4, we calculate the weight π_i for each particle s_i by combining the normalized probabilities of the visual observation V_t and the acoustical observation A_t.

$$\pi_i = c_A \cdot p(A_t|s_i) + c_V \cdot p(V_t|s_i) \tag{1}$$

The dynamic mixture weights c_A and c_V can be interpreted as confidence measures for the audio and video channel respectively. In order to determine the values of $c_{A,V}$, we consider the spread of the audio and video scores on the ground plane (x and y components of the state vector). Let for example σ_x^A denote the standard deviation of the particle set's x-components weighted with the audio scores, then the audio channel confidence is given by:

$$c_A = \sqrt{(\sigma_x^A)^2 + (\sigma_y^A)^2}^{(-1)} \tag{2}$$

The video confidence c_V is calculated in the same way. In order to generate the final tracker output, we shift a $1m^2$ sized search window over the ground plane and look for the region with the highest accumulated particle scores. The weighted mean of all particles within that region is then the final hypothesis.

3 Video Features

As lecturer and audience cannot be separated reliably by means of fixed spatial constraints as, e.g., a dedicated speaker area, we have to look for features that are

more specific for the lecturer than for the audience. Intuitively, the lecturer is the person that is standing and moving (walking, gesticulating) most, while people from the audience are generally sitting and moving less. In order to exploit this specific behavior, we use foreground segmentation based on adaptive background modeling as primary feature, as described in Section 3.1. In order to support the track indicated by foreground segments, we use detectors for face and upper body (see Section 3.2). Both features – foreground F and detectors D – are linearly combined[1] using a mixing weight β. So the probability of the visual information V_t^j in view j, given that the true state of the system is characterized by s_i, is set to be

$$\overline{p}(V_t^j|s_i) = \beta \cdot p(D_t^j|s_i) + (1-\beta) \cdot p(F_t^j|s_i) \tag{3}$$

By means of the sum rule, we integrate the weights from the v different views in order to obtain the total probability of the visual observation:

$$\overline{p}(V_t|s_i) = \frac{1}{v} \sum_{j=1..v} \overline{p}(V_t^j|s_i) \tag{4}$$

To obtain the desired (pseudo) probability value which tells us how likely this particle corresponds to the visual observation we have to normalize over all particles:

$$p(V_t|s_i) = \frac{\overline{p}(V_t|s_i)}{\sum_i \overline{p}(V_t|s_i)} \tag{5}$$

3.1 Foreground Segmentation

In order to segment the lecturer from the background, we use a simple background model $b(x, y)$ that is updated with every new frame $z(x, y)$ using a constant update factor α:

$$b(x,y) = (1-\alpha) \cdot b(x,y) + \alpha \cdot z(x,y) \tag{6}$$

The foreground map $m(x, y)$ is made up of pixel-wise differences between the current image $z(x, y)$ and the background model $b(x, y)$. It is scaled using minimum/maximum thresholds τ_0 and τ_1:

$$m(x,y) = \frac{|z(x,y) - b(x,y)| - \tau_0}{\tau_1 - \tau_0} \cdot 255 \tag{7}$$

The values of $m(x, y)$ are clipped to a range from 0 to 255.

However, as Fig. 1 shows, the resulting segmentation of a crowded lecture room is far from perfect. Morphological filtering of the foreground map is generally not sufficient to remove the noise and to create a single connected component for the lecturer's silhouette. Nonetheless, the combination of the foreground maps from different views contains enough information to locate the speaker. Thus, our approach gathers support from all the views' maps without making any "hard" decisions like a connected component analysis.

[1] Note that $p(D_t^j|s_i)$ and $p(F_t^j|s_i)$ respectively have to be normalized before combination so that they sum up to 1.

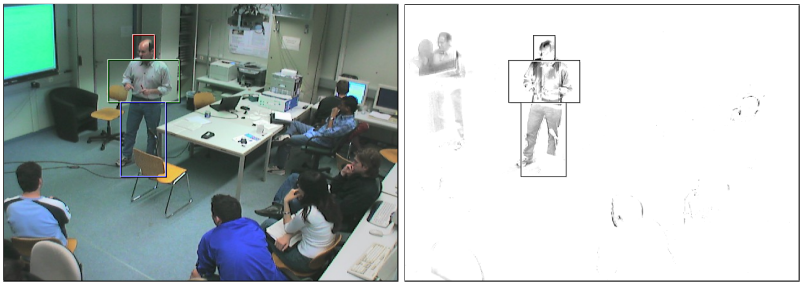

Fig. 1. Foreground segmentation is performed by means of an adaptive background model. A "3-boxes model" approximates the speaker's appearance.

As described in Section 2, the particle filter framework merely requires us to assign scores to a number of hypothesized head positions. In order to evaluate a hypothesis $s_i = (x, y, z)$, we project a "3-boxes person model" (see Fig. 1) centered around the head position to the image plane of each camera view, and sum up the weighted foreground pixels $m(x, y)$ inside the projected polygons: The topmost box, representing the head, has a height of $28cm$ and a width/depth of $18cm$. The torso box has a width and depth of $60cm$, whereas the box for the legs spans $40cm$. The accumulated weights of the foreground pixels within the projected polygons are then used as the particle's score.

As this calculation has to be done for each of the particles in all views, we use the following simplification in order to speed up the procedure: we assume that all cameras are set upright with respect to the ground plane, so the projection of a cuboid can be approximated by a rectangle orthogonal to the image plane, i.e. the bounding box of the projected polygon (see Fig. 2).

The sum of pixels inside a bounding box can be computed efficiently using the integral image introduced by [16]. Given the foreground map $m(x, y)$, the integral image $ii(x, y)$ contains the sum of the pixels above and to the left of (x, y):

$$ii(x, y) = \sum_{y'=0}^{y} \sum_{x'=0}^{x} m(x', y') \tag{8}$$

Thus, the sum of the rectangle (x_1, y_1, x_2, y_2) can be determined by four lookups in the integral image. So the particle score for the foreground feature is defined by the sum of pixels inside the bounding boxes normalized by the size of the bounding boxes:

$$p(F_t^j|s_i) = \sum_{b=H,T,L} \frac{ii(x_2^b, y_2^b) - ii(x_1^b, y_2^b) - ii(x_2^b, y_1^b) + ii(x_1^b, y_1^b)}{(x_2^b - x_1^b + 1)(y_2^b - y_1^b + 1)} \tag{9}$$

The index b specifies the head box (H), torso box (T), and legs box (L).

Using the recurrent formulation from [16], the generation of the integral image only takes one pass over the foreground map, so the complexity of the

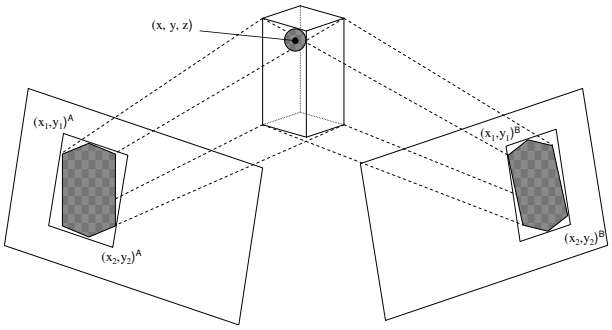

Fig. 2. For each particle and each body segment (head, torso, legs), a cuboid centered around the hypothesized head position (x, y, z) is projected into the views A and B. The resulting polygon is approximated by a bounding box $(x_1, y_1, x_2, y_2)^{A/B}$.

foreground feature preparation is linear to the image size. The evaluation of one particle can then be done in constant time, and is thus independent of the image resolution and the projected size of the target.

3.2 Face and Upper Body Detection

As we aim at tracking the coordinates of the lecturer's head – serving as model point for the full body –, we need a feature that gives evidence for the head position. The face detection algorithm proposed by Viola and Jones [16] is known to be both robust and fast: it uses Haar-like features that can be efficiently computed by means of the integral image, thus being invariant to scale variations. The features are organized in a cascade of weak classifiers, that is used to classify the content of a search window as being face or not.

Typically, a variable-size search window is repeatedly shifted over the image, and overlapping detections are combined to a single detection. Exhaustively searching a $W \times W$ image region for a $F \times F$ sized face while incrementing the face size n times by the scale factor s requires the following number of cascade runs (not yet taking into account post-filtering of overlapping detections):

$$\#cascade\ runs = \sum_{i=0}^{n-1} \left(W - F \cdot s^i\right)^2 \tag{10}$$

In case of for example a 100x100 pixel image region, and a face size in between 20 and 42 ($n = 8, s = 1.1$), this results in 44368 cascade runs.

In the proposed particle filter framework however, it is not necessary to scan the image exhaustively: the places to search are directly given by the particle set. For each particle, a head-sized cuboid (30cm edge length) centered around the hypothesized head position is projected to the image plane, and the bounding box of the projection defines the search window that is to be classified. Thus,

the evaluation of a particle takes only one run of the cascade:

$$\#cascade\ runs = \#particles \tag{11}$$

The face detector is able to locate the vertical and horizontal position of the face precisely with respect to the image plane. However, the distance to the camera, i.e. the scaling, cannot be estimated accurately from a single view. In order to achieve tolerance against scale variation and to smooth the scores of nearby particles, we set the i-th particle's score to the (average) overlap[2] between the particle's head rectangle $r_i = (x_1, y_1, x_2, y_2)$ and all the positively classified head rectangles $r'_{0..N}$ by any of the other particles:

$$p(D_t^j | s_i) = \frac{1}{N} \sum_{n=0}^{N} overlap(r_i, r'_n) \tag{12}$$

A detector that is trained on frontal faces only is unlikely to produce many hits in our multi-view scenario. In order to improve the performance, we used two cascades for face detection: one for frontal faces in the range of $\pm 45°$ and one for profile faces $(45° - 90°)$[3]. Our implementation of the face detector is based on the OpenCV library, that implements an extended set of Haar-like features as proposed by [12]. This library also includes a pre-trained classifier cascade for upper body detection [11]. We used this detector in addition to face detection, and incorporated it's results using the same methods as described for face detection.

4 Audio Features

The lecturer is the person that is normally speaking, therefore we can use audio features using multiple microphones to detect the speaker position. Consider the j-th pair of microphones, and let $\mathbf{m}_{j1}$ and $\mathbf{m}_{j2}$ respectively be the positions of the first and second microphones in the pair. Let $\mathbf{x}$ denote the position of the speaker in a three dimensional space. Then the *time delay of arrival* (TDOA) between the two microphones of the pair can be expressed as

$$T_j(\mathbf{x}) = T(\mathbf{m}_{j1}, \mathbf{m}_{j2}, \mathbf{x}) = \frac{\|\mathbf{x} - \mathbf{m}_{j1}\| - \|\mathbf{x} - \mathbf{m}_{j2}\|}{c} \tag{13}$$

where c is the speed of sound. To estimate the TDOAs a variety of well-known techniques [14, 5] exist. Perhaps the most popular method is the *phase transform* (PHAT), which can be expressed as

$$R_{12}(\tau) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{X_1(e^{j\omega\tau}) X_2^*(e^{j\omega\tau})}{|X_1(e^{j\omega\tau}) X_2^*(e^{j\omega\tau})|} \, e^{j\omega\tau} \, d\omega \tag{14}$$

[2] The auxiliary function $overlap(a, b)$ calculates the ratio of the shared area of two rectangles a and b to the sum of the areas of a and b.

[3] The profile face cascade has to be applied twice: to the original image and to a horizontally flipped image.

where $X_1(\omega)$ and $X_2(\omega)$ are the Fourier transforms of the signals of a microphone pair in a microphone array. Normally one would search for the highest peak in the resulting cross correlation to estimate the position. But since we are using a particle filter, as described in Section 2, we can simply set the PHAT value at the time delay position $T_j(\mathbf{x} = \mathbf{s}_i)$ of the MA pair j of a particular particle s_i as

$$\overline{p}(A_t^j|\mathbf{s}_i) = max(0, R_j(T_j(\mathbf{x} = \mathbf{s}_i))) \tag{15}$$

As the values returned by the PHAT can be negative, but probability density functions must be strictly nonnegative, we set negative values of the PHAT to zero.

To get a better estimate we repeat this over all m pair of microphones, sum their values and normalize by m:

$$\overline{p}(A_t|\mathbf{s}_i) = \frac{1}{m}\sum_{j=1}^{m}\overline{p}(A_t^j|\mathbf{s}_i) \tag{16}$$

Just like for the visual features, we normalize over all particles in order to get the acoustic observation likelihood for each particle:

$$p(A_t|\mathbf{s}_i) = \frac{\overline{p}(A_t|\mathbf{s}_i)}{\sum_i \overline{p}(A_t|\mathbf{s}_i)} \tag{17}$$

5 Experiments on the CLEAR'06 Evaluation Dataset

The performance of the proposed algorithm has been evaluated on the single person tracking tasks of the CLEAR'06 Evaluation Campaign [1]. The dataset consists of recordings of actual lectures and seminars that were held at different sites.

The evaluation dataset comprises a total number of 14 recordings, each featuring a different speaker (see Fig. 3). From each recording typically two segments of 5 minutes length are to be processed (26 segments in total). The lectures are complemented by slides which are projected to a whiteboard next to the speaker. Apart from the lecturer, there is a number of about 5-20 people in the audience. In many recordings, there is no clear separation between speaker area and audience. It must further be noted that every once in a while, auditors cross the speaker area in order to enter or to leave the room.

The details of the sensor setup vary among the sites. There is, however, a common setup of 4 fixed cameras in the room corners with full room coverage, and 4-6 microphone arrays mounted on the walls. Each array consists of 4 microphones: 3 in a row with a distance of 20 cm and one 30 cm above the center microphone. The lecturer's head centroid was labeled manually every 10th frame. By means of the calibration information, a 3D label file was generated and serves as ground truth for the evaluation. A separate development dataset has been provided to tune tracking parameters and to train the face detection

Fig. 3. Snapshot from a lecture showing all 4 camera views.

cascades. This development set consists of different lectures that were collected using the same setup as for the evaluation set.

The system has not been hand-tuned to the different data collection sites. This means in particular:

- no images of the empty scene have been used, the background model initializes automatically
- no cameras or microphones were excluded - all sensors were used
- no speaker area has been defined, the tracker scans the entire room

5.1 Results

The evaluation results presented in Table 1 are average values of all 26 lecture segments that were provided in the single-person tracking task of CLEAR'06. Two scores were defined to rate the tracking systems:

- MOTA: the multi-person tracking accuracy accumulates misses and false positives and relates them to the number of labeled frames. In this evaluation, a miss is defined as a hypothesis outside a $500mm$ radius around the labeled head position. Note that each hypothesis outside the radius is a miss and a false positive by the same time, and is thus counted as 2 errors.
- MOTP: the multi-object tracking precision is the mean error of the hypotheses within the $500mm$ radius around the labeled head position.

Note that the most relevant score in the table is the miss rate and the MOTA score respectively, whereas MOTP only measures the precision for those hypotheses that are inside the $500mm$ range. For the video-only evaluation, all labeled frames were used for scoring, whereas the audio-only condition was scored exclusively on frames in which the lecturer actually speaks. The multi-modal system was scored on both conditions.

It can be seen in the table that the video-only tracker outperforms the audio-only tracker. The combination of both performs clearly better than the unimodal systems as long as scoring is done on speech frames only. When being evaluated on all frames, the audio-only tracker performs much worse than the video tracker, so that the combination of both is not beneficial anymore. As a comparison, Table 2 shows the results on the CLEAR'06 development set. Here, the combination of audio and video is beneficial even when being evaluated on all frames.

Tracking mode	Misses	MOTA	MOTP
Video only	14.3%	71.4%	127mm
Audio only	22.6%	54.8%	186mm
Video + Audio (speech frames)	5.1%	89.8%	140mm
Video + Audio (all frames)	14.6%	70.8%	143mm

Table 1. Results in 3D speaker tracking on the CLEAR'06 evaluation set.

Tracking mode	Misses	MOTA	MOTP
Video only	16.7%	66.5%	141mm
Audio only	15.3%	69.4%	138mm
Video + Audio (all frames)	8.1%	84.0%	125mm

Table 2. Comparative results on the CLEAR'06 development set.

In 3 of 26 evaluation segments, the audio-visual system has a miss rate of 55% or higher, whereas the miss rate on the other segments is always $< 30\%$. An in-depth look at those segments with worst performance reveals some reasons for this behavior: In the first of these three underperforming segments, the speaker is often standing in a corner of the room, speaking into the direction of the wall. Both audio and video fail here. The other two segments actually show the question-and-answer phase of a presentation. The speaker is standing still, while the participants are having a discussion. When being evaluated on all frames, the audio tracker tracks the current speaker, which is most of the time not the labeled presenter. Segments like this were not included in the development set.

5.2 Implementation and Complexity

For maximum precision, the experiments on the CLEAR'06 dataset have been conducted with full image resolution and a number of 500 particles. The processing time for 1sec of data (all sensors together) on a single 3GHz PC was 2.3sec (audio-only), 4.8sec (video-only) and 11.6sec (audio-visual).

On the video side, the proposed algorithm consists of two parts that can be characterized by their relation to the three factors that determine the runtime of the algorithm. The feature preparation part (foreground segmentation, integral image calculation) is related linearly to the image size S and a constant time factor t_{VS}. In contrast, the particle evaluation part is independent from S and related linearly to the number of particles P and a constant t_{VP}. Both parts are likewise related linearly to the number of views V. On the audio side, the runtime is linearly related to the number of microphone pairs M. Like in the video case, this can be further decomposed into a constant preprocessing part t_{AM} and a part t_{AP} that has to be repeated for each particle. Thus, the total processing time per frame is determined by:

$$t_{total} = (t_{VS} \cdot S + t_{VP} \cdot P) \cdot V + (t_{AM} + t_{AP} \cdot P) \cdot M \tag{18}$$

As this equation indicates, the visual part can be intuitively parallelized for the number of views V. We implemented such a video-only tracker using 4 desktop PCs, each connected to a camera, in a way that the image processing is done locally on each machine. Because only low-bandwidth data (particle positions and weights) are shared over the network, the overhead is negligible, and a speed of 15 fps (including image acquisition) could easily be achieved. In the live system, image downsampling by a factor of $2-4$ and a number of $100-200$ particles performs reasonably well.

6 Conclusion

We presented an algorithm for tracking a person using multiple cameras and multiple pairs of microphones. The core of the proposed algorithm is a particle filter that works without explicit triangulation. Instead, it estimates the 3D location by sampled projection, thus benefiting from each single view and microphone pair. The video features used for tracking are based on foreground segmentation and the response of detectors for upper body, frontal face and profile face. The audio features are based on the time delays of arrival between pairs of microphones, and are estimated with a generalized cross correlation function.

The audio-visual tracking algorithm was evaluated on the CLEAR'06 dataset and outperformed both the audio- and video-only tracker. One reason for this is that the video and audio features described in this paper complement one another well: the comparatively coarse foreground feature along with the audio feature guide the way for the face detector, which in turn gives very precise results as long as it searches around the true head position. Another reason for the benefit of the combination is that neither motion and face detection nor acoustic source localization responds exclusively to the lecturer and not to people from the audience – so the combination of both increases the chance of actually tracking the lecturer.

Acknowledgments

This work has been funded by the European Commission under Project CHIL (http://chil.server.de, contract #506909).

References

1. CLEAR 2006 Evaluation and Workshop Campaign. http://clear-evaluation.org. April 6-7 2006, Southampton, UK.
2. M. S. Brandstein. *A framework for speech source localization using sensor arrays.* PhD thesis, Brown University, Providence, RI, May 1995.
3. M. S. Brandstein, J. E. Adcock, and H. F. Silverman. A closed-form location estimator for use with room environment microphone arrays. *IEEE Trans. Speech Audio Proc.*, 5(1):45–50, January 1997.

4. N. Checka, K. Wilson, V. Rangarajan, and T. Darrell. A probabilistic framework for multi-modal multi-person tracking. In *IEEE Workshop on Multi-Object Tracking (in conjunction with CVPR)*, 2003.
5. J. Chen, J. Benesty, and Y. A. Huang. Robust time delay estimation exploiting redundancy among multiple microphones. *IEEE Trans. Speech Audio Proc.*, 11(6):549–57, November 2003.
6. D. Gatica-Perez, G. Lathoud, I. McCowan, and J.-M. Odobez. A mixed-state i-particle filter for multi-camera speaker tracking. In *Proc. IEEE ICCV Workshop on Multimedia Technologies in E-Learning and Collaboration (ICCV-WOMTEC)*, 2003.
7. Yiteng Huang, Jacob Benesty, Gary W. Elko, and Russell M. Mersereau. Real-time passive source localization: A practical linear-correction least-squares approach. *IEEE Trans. Speech Audio Proc.*, 9(8):943–956, November 2001.
8. M. Isard and A. Blake. Condensation–conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.
9. T. Gehrig, K. Nickel, H. K. Ekenel, U. Klee, and J. McDonough. Kalman filters for audio-video source localization. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2005.
10. U. Klee, T. Gehrig, and J. McDonough. Kalman filters for time delay of arrival-based source localization. *EURASIP Special Issue on Multichannel Speech Processing*, submitted for publication.
11. H. Kruppa, M. Castrillon-Santana, and B. Schiele. Fast and robust face finding via local context. In *IEEE Intl. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, October 2003.
12. R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *ICIP*, volume 1, pages 900–903, September 2002.
13. I. Mikic, S. Santini, and R. Jain. Tracking objects in 3d using multiple camera views. In *ACCV*, 2000.
14. M. Omologo and P. Svaizer. Acoustic event localization using a crosspower-spectrum phase based technique. *Proc. ICASSP*, II:273–6, 1994.
15. J. Vermaak, M. Gangnet, A. Blake, and P. Pez. Sequential monte carlo fusion of sound and vision for speaker tracking. In *Proc. IEEE Intl. Conf. on Computer Vision*, volume 1, pages 741–746, 2001.
16. P. Viola and M. Jones. Robust real-time object detection. In *ICCV Workshop on Statistical and Computation Theories of Vision*, July 2001.
17. D. B. Ward, E. A. Lehmann, and R. C. Williamson. Particle filtering algorithms for tracking an acoustic source in a reverberant environment. *IEEE Trans. Speech Audio Proc.*, 11(6):826–836, 2003.
18. M. Wölfel, K. Nickel and J. McDonough. Microphone Array Driven Speech Recognition: Influence of Localization on the Word Error Rate. *2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, Edinburgh, 11-13 July 2005.
19. D. Zotkin, R. Duraiswami, and L. Davis. Joint audio-visual tracking using particle filters. *EURASIP journal on Applied Signal Processing*, 2002(11), 2002.