# LOW-LATENCY ONLINE SPEAKER TRACKING ON THE AMI CORPUS OF MEETING CONVERSATIONS

*Maider Zamalloa*[1,2], *Luis Javier Rodríguez-Fuentes*[1], *Germán Bordel*[1], *Mikel Penagarikano*[1], *Juan Pedro Uribe*[2]

[1] GTTS, Department of Electricity and Electronics, University of the Basque Country, Spain

[2] Ikerlan, Technological Research Centre, Spain

maider.zamalloa@ehu.es

## ABSTRACT

Ambient Inteligence aims to create smart spaces providing services in a transparent and non-intrusive fashion, so context awareness and user adaptation are key issues. Speech can be exploited for user adaptation in such scenarios by continuously tracking speaker identity. However, most speaker tracking approaches require processing the full audio recording before determining speaker turns, which makes them unsuitable for online processing and low-latency decision-making. In this work a low-latency speaker tracking system is presented, which deals with continuous audio streams and outputs decisions at one-second intervals, by scoring fixed-length audio segments with a set of target speaker models. A smoothing technique is explored, based on the scores of past segments, which increases the robustness of tracking decisions to local variability. Experimental results are reported on the AMI Corpus of meeting conversations, revealing the effectiveness of the proposed approach when compared to an offline speaker tracking approach developed for reference.

***Index Terms*** **—** Speaker Recognition, Low-latency, Speaker Tracking, AMI Corpus, Ambient Intelligence

## 1. INTRODUCTION

Ambient Intelligence (AmI) is an interdisciplinary applied research field, aiming to create smart spaces which provide services featuring user and context adaptation capabilities [1]. Speech is a natural interface for user interaction and adaptation. Speech streams can be exploited to extract user related information such as location, identity, emotional state, etc. Since user adaptation must be done in a continuous fashion, online processing and low-latency decision-making are key issues.

Speaker tracking is a well-known task which aims to detect speech segments corresponding to known target speakers in an audio resource [2]. Typically, there have been three main application domains for speaker tracking: broadcast news, meetings and telephone conversations. The methodologies applied in such domains [3][4][5] require processing the full audio recording before determining speaker turns, which makes them unsuitable for online processing and low-latency decision-making. Few works featuring low-latency speaker segmentation and tracking can be found in the literature [6][7][8]. In those works, low-latency was achieved by detecting speaker changes dynamically and by determining speaker labels in an unsupervised way, since neither the number nor the identity of speakers was known a priori. In this work, however, the speaker tracking system is required to continuously track a low number of known speakers (the members of a family) in a smart home environment.

Keeping in mind the particular conditions of this scenario, a very simple speaker tracking algorithm is proposed, where audio segmentation and speaker detection are jointly accomplished by processing fixed-length audio segments and scoring each of them to decide whether it belongs to a target speaker or to an impostor. Since speaker detection is done for very short (one-second length) segments, the performance of the online speaker tracking system may degrade due to local variability. So, a smoothing technique, based on a linear combination of present and past scores, is proposed to increase the robustness to such variability.

The performance of the proposed approach is compared to that of an offline system developed for reference, which follows a classical two-stage metric-based approach: change points are located in adjacent windows over the whole input stream by applying a BIC-like criterion, and speaker detection is performed on the resulting segments. Evaluation is carried out on the AMI Corpus (Augmented Multi-party Interaction) [9], which contains human conversations in the context of smart meeting rooms, close to the AmI scenario described above.

The rest of the paper is organized as follows. In section 2, the main features of the speaker tracking systems are described. Section 3 gives details about the experimental setup. Results using the online and offline speaker tracking systems are presented and discussed in Section 4. Finally, conclusions and future work are outlined in section 5.

## 2. SPEAKER TRACKING SYSTEMS

### 2.1. Acoustic front-end

In this work, 16 kHz audio streams are analyzed in frames of 20 milliseconds, yielding a vector of 12 Mel-Frequency Cepstral Coefficients (MFCC) per frame. To increase robustness against channel distortion, Cepstral Mean

Normalization (CMN) is applied. When the audio stream is processed on-the-fly, a dynamic CMN approach is applied, where the cepstral mean is updated at each time *i* as follows:

$$\mu_i = \alpha C(i) + (1-\alpha)\mu_{i-1} \quad , \qquad (1)$$

where $\alpha$ is a time constant (typically, around 0.001), *C(i)* is the vector of cepstral coefficients at time *i* and $\mu_{i-1}$ is the dynamic cepstral mean at time *i-1*. After CMN, the first derivatives of the MFCC are also computed, yielding a 24 dimensional feature vector.

### 2.2. Audio segmentation

In this work, audio segmentation is needed only by the offline speaker tracking system developed for reference. A simple and computationally efficient algorithm is applied, which segments the audio signal in a fully unsupervised way, by locating the most likely change points from a purely acoustic point of view. The algorithm, similar to other metric-based approaches [10][11], considers a sliding window *W* of *N* acoustic vectors and computes the likelihood of change at the center of that window. Then moves the window *K* vectors ahead and repeats the process until the end of the vector sequence. To compute the likelihood of change, each window is divided in two halves, $W_{\text{left}}$ and $W_{\text{right}}$, then a Gaussian distribution with diagonal covariance matrix is estimated for each half, and finally a cross-likelihood ratio [12] is computed and stored as likelihood of change. This yields a sequence of cross-likelihood ratios which must be post-processed to get the hypothesized segment boundaries. This involves applying a threshold $\tau$ and forcing a minimum segment size $\delta$. In practice, a boundary *t* is validated when its cross-likelihood ratio exceeds $\tau$ and there is no candidate boundary with greater ratio in the interval $[t-\delta, t+\delta]$ (see [13] for details).

### 2.3. Speaker detection

The real-time speaker tracking system proposed in this work computes a detection score per target speaker and outputs a speaker detection decision for fixed-length segments. That length has been empirically set to one second, which provides relatively good time resolution and spectral richness, and a reasonably small latency for most online speaker tracking scenarios. The offline system developed for reference applies the same speaker detection strategy, but using the segments produced by the algorithm described in Section 2.2.

Audio segments are scored by means of Gaussian Mixture Models (GMM) corresponding to target speakers, estimated via Maximum a Posteriori (MAP) adaptation of a Universal Background Model (UBM) [14]. In this work, 256-mixture GMMs are used. Speaker model adaptation is based only on *non-overlapped* training segments, i.e. those segments containing only speech from the target speaker, according to the time references of manual annotations. The MAP-UBM methodology allows for a fast scoring technique, which is a key feature in order to achieve a low-latency response. Briefly, the computation of speaker likelihoods involves only the top C (in this work, C=8) scoring mixtures in the UBM computation, which is done in first place (see [14]).

Given the acoustic observation X and the acoustic models $\lambda_t$ for the target speaker t, and $\lambda_{\text{UBM}}$ for the UBM, the detection score for the target speaker t, $\Delta_t(X)$, is computed as follows:

$$\Delta_t(X) = L(X|\lambda_t) - L(X|\lambda_{UBM}) \quad , \qquad (2)$$

where $L(X|\lambda)$ is the log-likelihood of X given $\lambda$.

### 2.4. Calibration of scores

Before taking a decision, detection scores are calibrated to compensate for differences in means and variances which may degrade performance. Calibration maps detection scores $\Delta_t(X)$ to likelihood ratios $C(\Delta_t(X))$, without any specific application in mind. The scaling parameters are computed over a development corpus by maximizing *Mutual Information*, which is equivalent to minimizing the so called $C_{\text{LLR}}$ (a metric defined in [15]), which integrates the expected cost over a wide range of operation points (representing specific applications). Since scaling parameters are computed beforehand, calibration does not significantly increase the computational cost of the speaker tracking system.

The final decision is taken by applying the minimum expected cost Bayes decision threshold to calibrated scores $C(\Delta_t(X))$. The most likely speaker $\hat{t}$ is detected if the following inequality holds:

$$C(\Delta_{\hat{t}}) \geq \ln\left(\frac{C_{fa}(1-P_{target})}{C_{miss}P_{target}}\right) \quad , \qquad (3)$$

where $C_{\text{miss}}$ and $C_{\text{fa}}$ are the miss and false-acceptance error costs, and $P_{\text{target}}$ is the prior probability of target speakers. Otherwise, X is marked as unknown (i.e. coming from a non-target speaker, noise, etc.). Calibration of scores is done by means of the FoCal toolkit [16] with a linear mapping strategy.

Note that, for any given segment X, there could actually be two or more speakers speaking at the same time. However, the detection approach described above cannot inform about speaker overlaps, because only the most likely target speaker can be detected.

### 2.5. Smoothing of scores

Since speaker detection is done for very short (one-second length) segments, the performance of the low-latency online speaker tracking system may degrade due to local variability. To increase the robustness to such variability, information from previous segments can be taken into account, that is, the acoustic scores of target speakers may be based on speech segments lasting more than one second. Assuming that no speaker change takes place in the previous segments, scores will be more accurate as more samples are used to compute them. On the other hand, this does not

affect the online processing and low-latency decision-making constraints. In practice, a smoothed score is computed by linearly combining the scores of the last $w$ (one-second length) segments, weighting them according to a rectangular (uniform) or a triangular (linearly decreasing as going back in time) function.

## 3. EXPERIMENTAL SETUP

### 3.1. The AMI Corpus

Experiments were carried out over the AMI meeting corpus, which is available as a public resource [17]. The AMI Corpus contains human interactions in the context of smart meeting rooms. Data, collected in three instrumented meeting rooms, include a range of synchronized audio and video recordings. Meetings contain speech in English, spoken by native and (mostly) non native speakers.

In this work, the development and evaluation of speaker tracking systems was based on a subset of the AMI corpus, the Edinburgh scenario meetings, including 15 sessions: ES2002-ES2016, with four meetings per session, each meeting being half an hour long on average. Training data were taken from meetings recorded at the three AMI sites. The audio stream was obtained by mixing the signals from the headset microphones of the participating speakers. Three of the four speakers participating in each session were taken as target speakers, the remaining one being assigned the role of impostor. Careful impostor selection (not random) was made, in order to avoid that gender favors impostor discrimination. For instance, in sessions containing just one female speaker, the impostor was forced to be male (and viceversa).

In order to avoid the evaluation to be tilted by tuning, two independent subsets were defined, consisting of different sessions (and therefore different speakers), for development and evaluation purposes. The development set, consisting of 8 sessions (32 meetings), was used to tune the configuration parameters of the speaker tracking systems. The evaluation set, including the remaining 7 sessions (28 meetings), was used only to evaluate the performance of the previously tuned speaker tracking systems. Both the development and evaluation subsets were further divided into train and test datasets. Two meetings per session were randomly selected for training speaker models, and the remaining two were left for testing purposes.

### 3.2. UBM estimation

Two speaker detection systems were developed, which differed in the data used to estimate the UBM: UBM-g used 15 gender-balanced AMI meetings from all sites except Edinburgh (so, a kind of room mismatch may be expected), whereas UBM-t used only speech from training meetings. UBM-g was estimated once and could be applied to whatever evaluation data and target speakers, whereas UBM-t had to be estimated specifically for each set of target speakers.

### 3.3. Performance measures

In the following, performance is analyzed by means of Detection Error Tradeoff (DET) plots. When a single figure is needed, the Equal Error Rate (EER) is used. Performance is measured in terms of time that is correctly or incorrectly classified as belonging to a target speaker. Therefore, miss and false alarm rates are computed as a function of time and not as a function of trial number, like in speaker detection experiments. Collar periods of 250 milliseconds at the end of speaker turns are ignored for scoring purposes. Thus, speaker turns of less than 0.5 seconds are not scored. Segments containing speech from two or more speakers are not scored either.

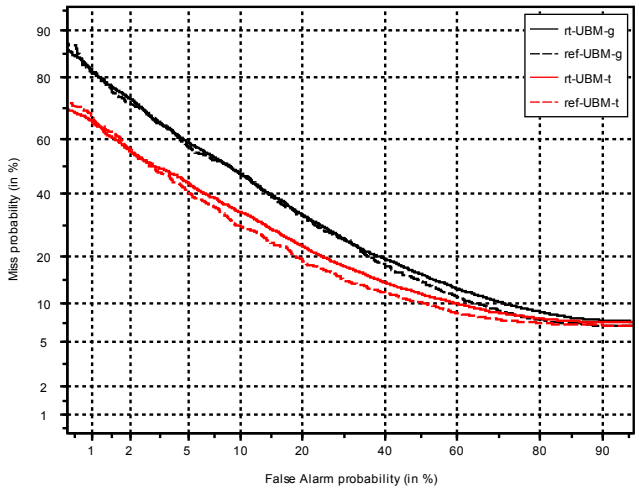


**Figure 1.** DET curves for the online (rt) and offline (ref) speaker tracking systems on the evaluation set of the AMI Corpus, using UBM-t (matching room and speakers) and UBM-g (general) as background models.

## 4. EXPERIMENTAL RESULTS

### 4.1. Online vs. Offline speaker tracking

Figure 1 shows the performance of the online and offline speaker tracking systems, using UBM-g and UBM-t as background models, for the evaluation set of the AMI Corpus. As expected, the offline system outperforms the proposed low-latency online system, but the performance of the latter is relatively good. The EER increases from 25.80 to 26.20 (1.55% relative degradation) when using UBM-g, and from 19.07 to 21.14 (10.85% relative degradation) when using UBM-t. Note that UBM-t outperforms UBM-g, probably due to the aforementioned room mismatch in UBM-g and the limited amount of training data. This suggests the use, whenever possible, of a room-specific UBM. In fact, UBM-t might be getting advantage not only from matching the room, but also from the consistency between the speakers in the UBM and the target speakers. In fact, 100% of the target speakers appearing in the test corpus contribute data to the UBM-t, increasing the consistency of speaker models estimated through MAP adaptation (because

a perfect match exists between the adaptation data corresponding to any target speaker and some of the component densities of the UBM).

### 4.2. Results with smoothed scores

In Figure 2, DET performance is shown when detection scores are computed as a linear combination of the scores for the last *w* (one-second length) segments, either with uniform weights (rectangular function, *fs*) or with linearly decreasing weights (triangular function, *ft*). The optimal *w* (which somehow depends on the average length of speaker turns) was heuristically determined on the development set. For the rectangular function, the optimal value was *w=2*. For the triangular function, it was *w=3*. As shown in Figure 2, smoothing the scores consistently improved the speaker tracking performance on the evaluation set of the AMI Corpus, the EER decreasing from 21.14% (no smoothing, *w=1*) to 19.37% (*fs, w=2*) and 18.64% (*ft, w=3*), respectively. The same behaviour was observed in the development set, the EER decreasing from 18.94% (no smoothing, *w=1*) to 16.91% (*fs, w=2*) and 16.26% (*ft, w=3*), respectively.

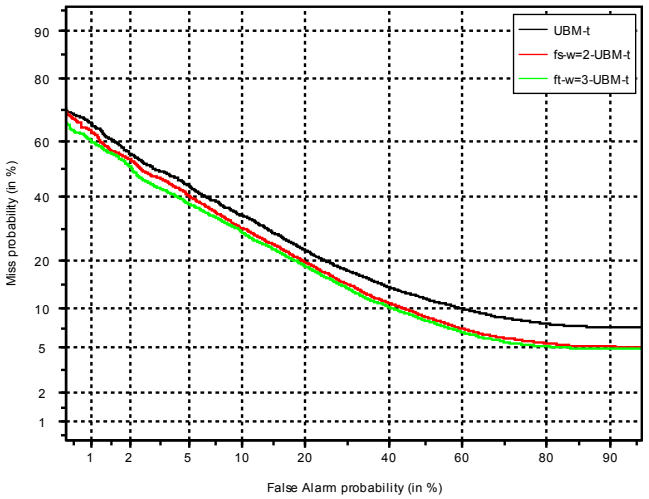


**Figure 2.** DET curves for the online speaker tracking system on the evaluation set of the AMI corpus: non-smoothed (UBM-t), and applying rectangular (fs, w=2) and triangular (ft, w=3) smoothing functions.

## 5. CONCLUSIONS AND FUTURE WORK

In this paper, a low-latency online speaker tracking system, specifically designed for an Ambient Intelligence scenario, has been described, and experimental results on a subset of the AMI corpus of meeting conversations have been presented. Results for an alternative speaker tracking system, based on an offline segmentation of the audio stream, followed by a MAP-UBM scoring backend, have been also presented for reference.

It has been found that the proposed system provides low-latency online speaker tracking with little performance degradation with regard to the reference system. A smoothing approach, consisting on linearly combining the scores of present and several past segments, has been also evaluated, yielding improved performance.

Future work includes using a more powerful speaker recognition backend (GMM-SVM), searching for more effective score smoothing schemes, using more realistic data (e.g. speech from distant microphones) and developing a low-latency online speaker tracking service, based on the approach presented in this paper, for an intelligent home environment.

## 6. REFERENCES


[1] ISTAG, "Scenarios for Ambient Intelligence in 2010". European Commission Report, 2001.

[2] Martin, A.F. and Przybocki, M.A., "Speaker Recognition in a Multi-Speaker Environment", in Proc. Eurospeech 2001.

[3] Moraru, D., Ben, M. and Gravier, G., "Experiments on Speaker Tracking and Segmentation in Radio Broadcast News", in Proc. Interspeech 2005, pp. 3049-3052.

[4] Istrate, D., Scheffer, N., Fredouille, C. and Bonastre, J.F., "Broadcast News Speaker Tracking for ESTER 2005 Campaign", in Proc. Interspeech 2005, pp. 2445–2448.

[5] Collet, M., Charlet, D. and Bimbot, F., "Speaker Tracking by Anchor Models using Speaker Segment Cluster Information", in Proc. ICASSP 2006, pp. 1009-1012.

[6] Liu, D., Kiecza, D., Srivastava, A. and Kubala, F., "Online Speaker Adaptation and Tracking for Real-time Speech Recognition", in Proc. of Interspeech 2005, pp. 281-284.

[7] Lu, L. and Zhang H.J., "Unsupervised Speaker Segmentation and Tracking in Real-time Audio Content Analysis", Multimedia Systems, 10:332-343, 2005.

[8] Wu, T.Y., Lu, L., Chen, K. and Zhang, H.J., "UBM-based Real-time Speaker Segmentation for Broadcasting News", in Proc. ICASSP 2003, Vol. 2, pp. 193-196.

[9] Carletta, J., "Unleashing the killer corpus: experiences in creating the multi-everything AMI Meeting Corpus". Language Resources and Evaluation Journal , 41(2): 181-190, 2007.

[10] Chen, S.C. and Gopalakrishnan, P.S., "Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion", in Proc. DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[11] Ajmera, J., McCowan, I. and Bourlard, H., "Robust Speaker Change Detection", IEEE Signal Proc. Letters, 11(8), 2004.

[12] Anguera, X., "XBIC: Real-Time Cross Probabilities measure for speaker segmentation", ICSI Technical Report TR-05-008, August 2005.

[13] Rodríguez, L.J., Peñagarikano, M. and Bordel, G., "A Simple But Effective Approach to Speaker Tracking in Broadcast News", Pattern Recognition and Image Analysis, LNCS 4478: 48-55, Springer-Verlag, 2007.

[14] Reynolds, D.A., Quatieri, T.F. and Dunn, R.B., "Speaker Verification Using Adapted Gaussian Mixture Models". Digital Signal Processing, 10:19-41, 2000.

[15] Brummer, N. and Preez, J., "Application Independent Evaluation of Speaker Detection", Computer Speech and Language, 20:230-275, 2006.

[16] FoCal Toolkit: "http://www.dsp.sun.ac.za/~nbrummer/focal/"

[17] AMI Corpus: "http://corpus.amiproject.org/"