Pervasive and Mobile Computing 8 (2012) 415–428

Contents lists available at SciVerse ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Collaborative personal speaker identification: A generalized approach

Mirco Rossi [a,*], Oliver Amft [a,b], Gerhard Tröster [a]

[a] *Wearable Computing Lab., ETH Zurich, Switzerland*[1]

[b] *ACTLab, Signal Processing Systems, TU Eindhoven, The Netherlands*[2]

ARTICLE INFO

Article history:
Received 11 April 2010
Received in revised form 1 September 2010
Accepted 15 February 2011
Available online 3 March 2011

Keywords:
Collaborative speaker identification
On-line learning
System collaboration
Open set
Unknown speaker identification

ABSTRACT

This paper introduces a collaborative personal speaker identification system to annotate conversations and meetings using speech-independent speaker modeling and one audio channel. This system can operate in standalone and collaborative modes, and learn about speakers online that were detected as unknown. In collaborative mode, the system exchanges current speaker information with personal systems of others to improve identification performance. Our collaboration concept is based on distributed personal systems only, hence it does not require a specific infrastructure to operate. We present a generalized description of collaboration situations and derive three use scenarios in which the system was subsequently evaluated.

Compared to standalone operation, collaboration among four personal identification systems increased system performance by up to 9% for 4 relevant speakers and up to 21% for 24 relevant speakers. Allowing unknown speakers in a conversation did not impede performance gains of a collaboration. In a scenario where individual systems had nonidentical speaker sets, collaboration gains were 16% for 24 relevant speakers.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Identifying speakers during meetings and conversations opens opportunities to analyze social relations, capture interesting moments in daily life, and can provide personal annotations of timing and content of conversations. The identification task consists of associating speakers with an unique identity and determining time segments when this speaker's identity was recorded. Stationary speaker identification systems have been realized in smart meeting rooms, where the identification relies on static setups and resource allocations. In contrast, mobile and wearable systems can enable personal speaker annotations without being constrained to particular locations. Systems such as the body-worn Sociometer [1] revealed the large potential of identifying speakers in social interactions. Moreover, personal annotations of social contacts and conversations allow one to search, select, and retrieve information from databases that could capture all audio and visual information acquired during everyday life [2]. Several projects have been initiated to provide this information, including MyLifeBits [3] and Interest-Based Life Logging [4].

A personal wearable speaker identification system requires one to cope with a number of challenges that affect its identification performance. First, the system has to be truly personal and autonomous, not dependent on tight collaboration with systems of others, since ad-hoc conversations may involve individuals without a compatible system. Second, the system has to be able to detect and learn about new speakers as conversations may involve new co-workers, friends, or strangers. The wide variety of scenarios where a personal identification system can be used renders a general system solution challenging.

* Corresponding author. Tel.: +41797748593.
E-mail addresses: mrossi@ife.ee.ethz.ch (M. Rossi), amft@tue.nl (O. Amft), troester@ife.ee.ethz.ch (G. Tröster).

1 http://www.wearable.ethz.ch.

2 http://www.actlab.ele.tue.nl.

1574-1192/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.pmcj.2011.02.005

For example, identification systems could be used by team workers, where members of a conversation are rather static and thus known in advance. Alternatively, systems may be used in an ad-hoc meeting with strangers, who do not use an identification system. There, speakers must be learned about before they can be identified. Consequently, the performance of an identification system can be severely limited if no additional information on the use condition is available.

An ad-hoc collaboration could help in many scenarios to improve standalone identification system performance. For example, personal systems could start with a speaker model for their owner only. When jointly exposed in a meeting, they would perform weakly in identifying other participants and in acquiring further speakers from the conversation. However, in this collaborative scope, relevant speakers are known already by each individual system, which could provide a crucial benefit for all participants.

In this work, we present a generalized approach to personal speaker identification that is independent of particular locations and can benefit from collaborative settings, in which multiple distributed systems share their recognition results. In our approach, a speaker can be modeled dynamically from voice data and subsequently identified using this model. While our system can be used in standalone operation, we foresee that systems exchange information to jointly recognize speakers and to decide whether a speaker is known to the collective. As our system concept supports learning of new speakers, collaboration is used as well to improve robustness for unsupervised speaker set extensions. We focus our evaluation on estimating empirical performance bounds for standalone and collaborative identification modes.

In particular this work provides the following contributions:

1. We present an unsupervised, text-independent speaker identification system using only one microphone. We show how our approach can be applied in different collaborative use scenarios, in which a personal speaker identification may be revealed. For this purpose, we introduce collaboration scenarios that account for unknown speakers and independent speaker model databases of the participating systems.
2. We study the performance of identification systems in collaborative operation, using a freely available multi-channel speaker corpus. We extracted 24 speakers with four speakers per meeting from this corpus. Our results confirm clear benefits, (1) for collaboratively recognizing speakers, (2) for unsupervised systems that collaborate during identification of new speakers, and (3) for mixtures of systems collaborating and systems "knowing" conversation-relevant speakers.

In our previous work we showed how a wearable speaker identification system could be realized that supports real-time operation [5]. We confirmed that a speaker identification could be efficiently performed on a wearable system. This present work substantially extends the wearable speaker identification approach by introducing a generalized collaborative identification system concept and an algorithmic solution that can deal with different collaborative use scenarios.

2. Related work

Automated speaker identification that could enable monitoring social interactions has been investigated from both application and technical perspectives for several years. These systems are either stationary and installed in rooms to annotate meetings or — as is aimed at in this work — the system can be worn as a daily personal accessory. The latter case allows one to identify interaction partners, annotate conversations, and build a personal diary of social activities.

Several smart meeting rooms have been proposed, such as at the Dalle Molle Institute [6] and at Berkeley [7]. These rooms are equipped with microphone arrays, typically at the table center and lapel microphones for each participant. To identify a speaking person, the lapel microphone having the highest input signal energy is chosen. The approaches are by far not restricted to monitoring using acoustic means alone. Approaches have been made to combine sensor information from multiple sources, including vision and audio [8,9]. An extensive review of these attempts is beyond the scope of this section however. As the systems are stationary their use is restricted to meetings and conversations held in the particular room. Wearable systems can capture conversations as they happen outside of these smart spaces.

An initial wearable system is the Sociometer developed by Choudhury and Pentland in 2002 [1]. This system can be attached to a person's shoulder. It includes an IR transmitter and receiver to communicate with people nearby. A microphone was used to separate speech from non-speech segments. The Sociometer is used for different kinds of social network analysis and organizational behavior, including analysis of social behavior in a research group [10], modeling of group discussion dynamics [11], and prediction of shopper's interest [12]. As the speaker identification with the Sociometer is achieved through IR communication, only individuals wearing this system can be recognized.

The works cited above impressively demonstrate the broad application potential of speaker identification. Nevertheless, these systems are limited by the prior knowledge and configuration required to operate them, such as the number and identity of speakers, and their location. Since those approaches did not use speaker modeling, the monitoring devices depend essentially on exchanging information on the current speaker. However, the availability of speaker models would allow one to use identification system while roaming between locations and continuously identifying speakers that have been modeled before. Subsequently, adding the capability to detect a new speaker allows one to learn about speakers dynamically and unsupervised.

Several procedures intended for unsupervised speaker recognition have been developed. Anliker [13] proposed an online speaker separation and tracking system based on blind source separation. The task of identifying speakers is largely facilitated by source separation, for which reason it had been used in many works. However, at least two microphones are required to perform source separation. This property imposes extended processing and power consumption requirements,

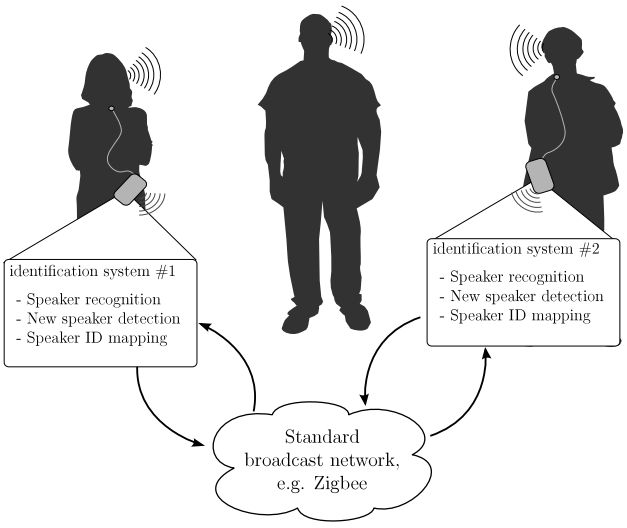


Fig. 1. Collaborative speaker identification architecture. Collaborating systems exchange information on speaker recognition, new speaker detection, and for speaker ID mapping.

which contradicts the viability of a wearable system implementation. Other algorithms that operate without speaker separation and, therefore, need one microphone only, have been proposed by Charlet [14], Lu and Zhang [15], Kwon and Narayanan [16], and Lilt and Kubala [17]. These works utilized different speech features including linear predictive cepstrum coefficients (LPCC), mel-frequency cepstrum coefficients (MFCC), and line spectrum pair (LSP). For modeling a speaker these systems typically use Gaussian Mixture Models (GMMs). It is known that GMMs may not be stably derived from small training data sizes. For unsupervised operation during conversations, however, only small data amounts may be available to learn about a new speaker online. In contrast, Vector Quantization (VQ) handles small training data sizes more effectively [18].

None of these single microphone systems investigated the benefit of collaborative speaker identification. Nonetheless, it can be expected that the reduced performance due to design choices for online and unsupervised operation could be compensated by ad-hoc collaboration of speaker identification systems. Anliker [13] addressed the case of collaborative information fusion with two and three systems performing source separation and speaker identification. However, the results did not show a clear improvement of the collaboration when compared to a standalone system. This observation may be attributed to the specific source separation and identification procedure considered in his work.

3. Collaborative speaker identification concept

The operation of a personal speaker identification system can change with the availability of collaboration partners and depends on the collaborative scenario. This section details our collaboration approach.

3.1. Collaborative speaker identification architecture

The principle function of a personal speaker identification system is to continuously annotate its users' conversations when worn. In standalone mode, a personal identification system analyzes the speech signal recorded from a microphone being worn. The speaker is identified using a speaker model (*Speaker recognition*) and it is detected whether a current speaker is known (*New speaker detection*). Unknown speakers are then automatically learned about by the system and stored as a speaker model in a system's database. This standalone mode does not involve collaboration with other systems at all and thus represents a baseline to study collaboration benefits.

In contrast, if two or more participants of a conversation use a personal speaker identification system, these systems could collaborate in their identification and detection tasks. In our approach systems periodically broadcast information of their current identification and detection results through an ad-hoc network, such as ZigBee [19]. An identification system can utilize information from other collaborative systems by fusing it with its own results. Thus, in collaborative mode, each system performs an individual speaker identification and new speaker detection as in standalone mode, while in addition, using information from others. To utilize information of others, a relation between the system's speaker identity (*Speaker ID*) representation and that of other systems must be derived (*Speaker ID mapping*). Fig. 1 illustrates the collaborative mode setting that is generally considered in this work.

In our implementation, the operation in standalone and collaborative mode can be switched at any time to ensure independence of a personal systems. Section 4 details the algorithms for speaker identification, new speaker detection, and speaker ID mapping.

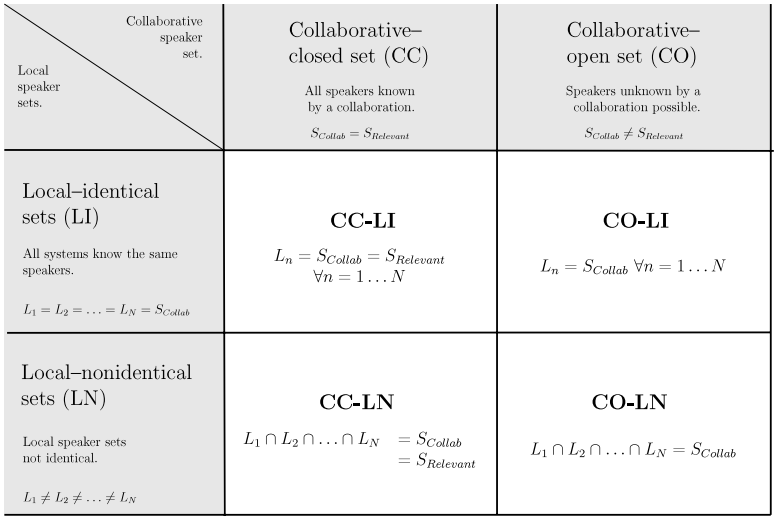


Fig. 2. Collaboration scenarios structured regarding speaker sets jointly known by a collaboration S_{Collab} (columns) and the relation of speakers sets known by individual systems $L_1, L_2, \ldots, L_N$ (rows).

Table 1
Speaker-related terminology and variables used in the collaboration scenario analysis.

Term	Description
Speaker model	A specific speaker model is used by an individual identification system to recognize the speaker. Speaker models are stored in the speaker database of an identification system
Speaker identity (speaker ID)	Speaker IDs are generated by individual speaker identification systems, corresponding to a speaker model. Speaker IDs are in general not compatible with IDs of other systems (see Section 4.2.3 for a description).
Relevant speakers S_{Relevant}	Denotes the speakers that actually participate in a conversation.
Local speaker set L_n	Contains speakers *known* by the system. A model for each of these speakers exists in the database of the identification system n.
Collaborative speaker set S_{Collab}	S_{Collab} denotes the set of speakers *known* in the joint set of systems that participate in a collaboration. Thus, a model for the speaker exists in the database of at least one system in a collaboration.

3.2. Collaboration scenario analysis and use cases

In collaborative mode, the state of each system's speaker model database essentially determines its benefit for others. Here, we consider all online information exchange between two or more speaker identification systems that participate in a conversation as a collaboration. In the most general case, no assumptions on the speaker model database can be made. However, as discussed below, specific collaboration applications exist, which could reduce identification uncertainty compared to this general case.

Systems that hold at least their wearer's speaker model in their speaker database can be of substantial benefit to others for recognizing this speaker. In contrast, systems that do not hold relevant speaker models for a conversation, cannot support the recognition step. In the worst case, a speaker is unknown by all systems in a collaboration (hence not in the speaker database of any collaborating system). This situation will be detected by each system, and corrected by learning a new model. Nevertheless, even in this situation systems can collaborate to ensure that the speaker is indeed unknown to all. Thus, in both situations, collaboration can support the operation of an individual system.

We structured the collaborative usage scenarios regarding the properties of speaker sets jointly known by the systems in a collaboration and regarding the relation of speakers sets known by individual systems. Fig. 2 provides an overview on the different collaborative use scenarios that result from these categories. We denote the set of speakers known by a collaboration with S_{Collab} and the total speaker set relevant in a collaboration with S_{Relevant}. The set of relevant speakers known by system n is referenced with L_n, where $L_n \subseteq S_{\text{Collab}}$ $\forall n = 1 \ldots N$, and N is the total number of systems in the collaboration. These terms and variables are summarized in Table 1.

We refer to scenarios where all speakers are know in a collaboration, thus $S_{\text{Collab}} = S_{\text{Relevant}}$, as *collaborative-closed* (CC). In contrast, $S_{\text{Collab}} \neq S_{\text{Relevant}}$ describes *collaborative-open* (CO) scenarios. In the latter scenarios, unknown, but relevant speakers exist. For local-identical (LI)-set systems, where $L_1 = L_2 = \cdots = L_N = S_{\text{Collab}}$, all identification systems have the set of relevant speakers in their databases. In contrast, in local-nonidentical (LN)-set systems, $L_1 \neq L_2 \neq \cdots \neq L_N$, the databases do not contain the same speakers.

CC–LI scenario. In this scenario all system databases contain identical speaker sets and all speakers in a collaboration are known. A typical use case for this scenario can be the use of identification systems by team workers, where the members

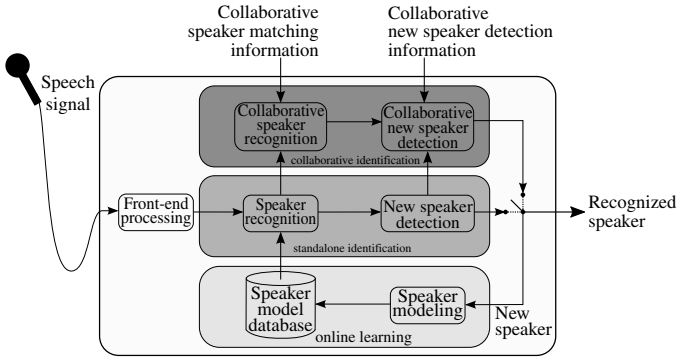


Fig. 3. Individual system design of a personal wearable speaker identification system supporting standalone and collaborative modes, as well as online learning of new speaker models.

of a conversation are known. The scenario applies as well for meetings, where all speakers use a collaborative system that learns about all speakers.

CC–LN scenario. In this scenario databases contain different relevant speaker sets. The speakers are known by at least one of the collaborative systems. We have not found a typical application for this scenario and thus assume that it is less likely to occur in practice.

CO–LI scenario. In this scenario all system databases contain identical speaker sets, but relevant speakers can be unknown to all systems. A typical situation occurs when identification systems start into a collaboration with a speaker model of their owner only. Thus, collectively, all relevant speakers are available.

CO–LN scenario. In addition to the open-set speaker problem of CO–LI, the system databases contain different speakers in this scenario. This is the most challenging scenario and applies to arbitrary conversations or meetings, where either not all speakers use a collaborative system or systems have acquired relevant speakers independently before entering in the current collaboration.

3.3. Individual system architecture

The previous sections focused on our collaboration architecture. In this section, we outline the functions and design of a personal speaker identification system that can operate in standalone and collaborative modes. Moreover, this system can learn new speaker models online.

In standalone mode, a local speaker recognition is performed by matching phonemes of continuous speech data against speaker pattern models stored in the system's speaker model database. Subsequently, a new speaker detection is used to estimate whether a relevant match with the database was found. Depending on this latter result, the speaker ID is returned or a new speaker model is learned.

In collaborative mode, a system provides its individual speaker recognition results to all collaboration partners and a collaborative recognition is performed. Subsequently, a collaborative new speaker detection is performed. Similar to the standalone operation, a decision is made to return a speaker ID or derive a new speaker model. With this distributed recognition approach, no central collaboration management instance is needed.

The feasibility of learning speaker models online was shown in our previous work [5]. Thus, we focus our evaluations in this work on the generalized collaboration scenarios. Fig. 3 illustrates the main components of the identification system. In Section 4, all system components are described in detail.

4. Identification algorithms

The implementation of all identification system functions (see Fig. 3) for standalone and collaborative mode operations are detailed in this section.

4.1. Standalone mode identification

Speaker identification in standalone mode refers to the base functionality of our system design. This functionality is used when no collaboration is active as well as when in collaboration mode.

4.1.1. Front–end audio processing

Front–end processing extracts speaker-dependent and text-independent features from the acquired audio signal using pre-processing, feature extraction, and channel compensation steps.

To minimize system complexity, we chose an 8 kHz sampling and 16 bit quantization rate. During pre-processing the raw audio signal was filtered with a transfer function $H(z) = 1 - \alpha z^{-1}$, where $\alpha = 0.97$. This filter emphasizes higher frequency bands and removes speaker-independent glottal effects [20].

Subsequent to this pre-processing, a feature vector $\mathbf{x} = (x_1, \ldots, x_N)$ was derived from the audio signal. In our previous work [5] we found that linear predictive cepstrum coefficients (LPCC) and mel-frequency cepstrum coefficients (MFCC) performed similarly well for recognizing speakers. However, the MFCC algorithm uses FFT, and thus has a larger computational complexity than LPCC [20]. In addition, we observed that for more than 12 coefficients, performance increased only marginally. Therefore, we chose LPCC with $N = 12$ coefficients in this present work. LPCC captures phonetic speaker properties, where phonemes are speech segments of about 20–30 ms [20]. We used a sliding window with 30 ms length and 20 ms step size to derive LPCC feature vectors.

A linear channel compensation was utilized to minimize device-dependent effects. We used here the short-term cepstral mean subtraction approach [20]. To account for an operation on continuous data, we adapted this method to be applied on sliding windows: $\widetilde{\mathbf{x}}^t = \mathbf{x}^t - \overline{\mathbf{x}}^t$, with $\overline{\mathbf{x}}^t = \frac{1}{T}\sum_{j=t-T}^{t} \mathbf{x}^j$. This corresponds to subtracting feature vector averages of the last T features from feature vector $\mathbf{x}^t$ generated at time t. T was set to the recognition epoch size (see Section 4.1.4).

4.1.2. Speaker modeling and recognition

During identification, feature vectors of a speech segment are compared with stored database models to identify known speakers.

With regard to a real-time operation of our speaker recognition and learning system, we chose Vector-Quantization (VQ) over other model learning techniques. VQ was found to require short training times [21] and has a low algorithm complexity compared to other approaches, which is a critical concern for a mobile system implementation.

With VQ, speaker models are formed by clustering a set of L training feature vectors $\{\mathbf{x}_i\}_{i=1}^{L}$ in K non-overlapping clusters. Each cluster is represented by a code vector $\mathbf{c}_i$ of a cluster centroid. A set of code vectors (codebook) $C = \{\mathbf{c}_i\}_{i=1}^{K}$ serves as a speaker model during recognition.

For this work we used the Generalized Lloyd algorithm (GLA) for clustering [22], which has low complexity compared to the other techniques. Moreover, modeling parameters including codebook size K and number of feature vectors L determine system complexity. We set the codebook size to $K = 16$, since larger codebook sizes had a marginal influence on performance [5]. We set $L = 1000$ which corresponds to a speech signal of length $t_{\text{train}} = 20$ s. This minimizes the training length, while keeping the speaker recognition accuracy at 80% [5].

To match a speaker, we determined the distance between a set of M test feature vectors $X = \{\mathbf{x}_i\}_{i=1}^{M}$ and a speaker codebook C. Distance d_q of $\mathbf{x}_i$ with respect to C was defined as $d_q(\mathbf{x}_i, C) = \min_{\mathbf{c}_j \in C}(d(\mathbf{x}_i, \mathbf{c}_i))$. Here $d(\mathbf{x}_i, \mathbf{c}_i)$ is a distance measure defined between two feature vectors, which was determined using the Euclidean distance. The mean of all individual distances was used as matching metric of a speaker model during recognition (Eq. (1)).

$$D(X, C) = \frac{1}{M}\sum_{i=1}^{M} d_q(\mathbf{x}_i, C). \tag{1}$$

Speakers are recognized by calculating distances of X to all codebooks C_i stored in a system's database. The recognized speaker is the codebook of speaker model C_{rec}, for which the smallest D was obtained.

Speaker recognition performance is proportional to the length of a recognition epoch, hence, the number of feature vectors M considered for each recognition. Nevertheless, long epochs could prevent a system from identifying rapid speaker changes in conversations. Based on an evaluation in our previous work [5] and chose $M = 250$, which corresponds to a speech signal length of $t_{\text{rec}} = 5$ s. Similar to the training length (determined by L), we chose M to minimize the recognition epoch while keeping the speaker recognition accuracy at 80%.

4.1.3. New speaker detection

In collaborative-open set scenarios, a speaker may be initially unknown to a system. The new speaker detection determines whether an observation belongs to the set of known speakers. For this purpose we defined a binary decision function:

$$f_{\text{NSD}}(X, C_{\text{rec}}) = \begin{cases} 1, & \text{if score } (X, C_{\text{rec}}) \geq \Delta_{\text{NSD}} \\ 0, & \text{else.} \end{cases} \tag{2}$$

X is a set of feature vectors of a tested speaker, C_{rec} is the recognized speaker model, score(X, C_{rec}) is a score function, and Δ_{NSD} is a threshold. When the score of a tested speaker is equal or larger than Δ_{NSD}, this tested speaker is classified to speaker model C_{rec}. However, if score(X, C_{rec}) is smaller than Δ_{NSD}, observation X will be classified as unknown speaker. As score function score$(\cdot)$ we used the negated D(X, C), normalized through distortions of an arbitrary speaker model set ("impostor speakers"):

$$\text{score}(X, C_{\text{rec}}) = -\frac{\text{D}(X, C_{\text{rec}}) - \mu_I}{\sigma_I}, \tag{3}$$

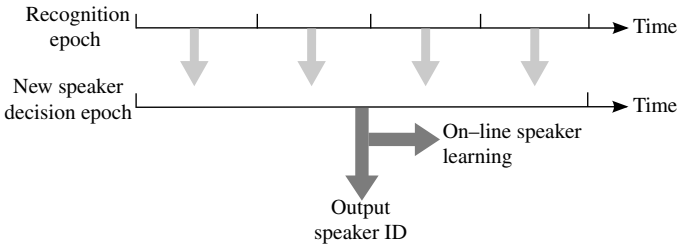


Fig. 4. Illustration of the timing relations between recognition, new speaker detection, and online learning for a real-time operation of our identification system.

Table 2
Example of shared normalized speaker model distances during a collaborative speaker recognition. The speaker with ID 3 is speaking. Systems 1 and 3 miss speaker models for ID 2 and 3, respectively. The collaborative speaker recognition can cope with variable sets of systems that collaborate and systems that know a speaker.

	Speaker IDs			
	ID 1	ID 2	ID 3	ID 4
System 1	−3.4	N/A	−8.4	11.9
System 2	−6.6	2.9	−0.8	4.5
System 3	−0.3	8.7	N/A	−8.3
System average	**−3.5**	**5.8**	**−4.6**	**2.7**

with mean μ_I and standard derivation σ_I of the impostor distortions. This score function corresponds to an impostor cohort normalization (ICN), a standard method in the speaker verification community [23,24]. We used $\Delta_{\text{NSD}} = 1.57$, which was obtained in our previous work by maximizing the new speaker detection accuracy for a given dataset [5]. The same threshold was used for collaborative new speaker detection (see Section 4.2.2).

4.1.4. Online learning procedure

When an unknown speaker is detected, this new speaker is enrolled in the system using online learning. All feature vectors that have been collected during recognition and new speaker detection are reused to derive the new speaker model.

For a real-time operation, timing constraints exist between recognition, new speaker detection, and online learning. Since an identified new speaker is instantly enrolled, the new speaker detection epoch was set to equal the training set size. Fig. 4 illustrates these timing relations. The current speaker is recognized in every recognition epoch of length $t_{\text{rec}} = 5$ s (see Section 4.1.2), whereas new speakers are detected on a sliding window of length $t_{\text{NSD}} = t_{\text{train}} = 20$ s and a window shift of one recognition epoch. If the current speaker is classified as known, a speaker ID is returned every $t_{\text{rec}} = 5$ s, whereas if the current speaker is classified as unknown, online learning of a new model is triggered.

4.2. Collaborative mode identification

This section details the algorithm implementation for collaborative speaker recognition and new speaker detection. In addition, we present a mapping method to exchange speaker IDs among collaborating systems.

4.2.1. Collaborative speaker recognition

The goal of collaborative speaker recognition is to improve the recognition performance of individual identification systems. For this purpose, systems in a collaboration exchange their locally obtained speaker recognition results at each recognition epoch $t_{\text{rec}} = 5$ s (see Section 4.1.2). More specifically, each identification system broadcasts the matching distances between speaker feature set X and all models in the system's model database $DB^{\text{sys}} = \{C_1^{\text{sys}}, \ldots, C_{N_{\text{sys}}}^{\text{sys}}\}$ (see Eq. (1)). To reduce channel differences of individual systems, each system transmits distances, normalized by subtracting the mean of all locally calculated distances for X. This normalization is needed to create comparable distances among the systems. In addition, channel effects were minimized in the front–end processing, as described in Section 4.1.1. Every collaborative system sorts the shared distances according to the speaker IDs and calculates the mean distance for each speaker. A speaker is recognized as the speaker ID exhibiting the smallest distance among all evaluated speakers. Since the collaborating systems exchanged sufficient information, the recognition can be performed by each system and the same speaker will be recognized.

Table 2 exemplarily illustrates the information sharing algorithm for a situation, in which a speaker with ID 3 is speaking. System 1 shares normalized matching distances of speaker ID 1, 3 and 4. Since system 1 has no model for speaker 2, no distance is shared for this ID. Similarly, system 3 cannot contribute to a collaboration regarding speaker ID 3. In this example, the collaborative recognition selects the correct speaker ID 3 as a final result, since ID 3 obtained the smallest normalized

matching distance. Although system 3 does not know ID 3, its sharing of distances is valuable for collaborative recognition. Without the contribution of system 3, speaker ID 1 would have obtained the best score in this example.

4.2.2. Collaborative new speaker detection

The collaborative new speaker detection aims at improving new speaker detection performance through collaboration with other systems. In analogy to the collaborative speaker recognition, new speaker detection results of local and remote systems are fused to obtain a collective decision, if a speaker is known to the collective.

Collaborative new speaker detection is used in collaborative-open set scenarios, subsequent to a collaborative recognition (see Fig. 3). In these scenarios both collaborative functions are performed consecutively at each recognition epoch t_{rec}. In contrast, for collaborative-closed set scenarios, all speakers are available in a collaboration set already, and thus the new speaker detection is not needed.

During collaborative new speaker detection, the identification systems are broadcasting model scores of the speaker ID id_{rec}, which was obtained in the preceding collaborative speaker recognition. The model score of a system sys, $\text{score}^{\text{sys}}(X, C^{\text{sys}}_{\text{id}_{\text{rec}}})$, is further described in Section 4.1.3. In a CO–LI scenario, all models know the speaker and can share their model score. However, in a CO–LN scenario, some collaborative systems might not have a model for the speaker ID (id_{rec}) and thus will not share any scores. Each system calculates the mean of the shared model scores: $\text{score}^{\text{mean}} = \text{mean}(\text{SCORE}_{\text{shared}})$, where $\text{SCORE}_{\text{shared}}$ is the set of shared scores. Finally, the decision function $f_{\text{NSD}}(\text{score}^{\text{mean}})$ is used with the threshold Δ_{NSD} to obtain a collective detection (see Eq. (2)). If an unknown speaker is detected, all systems train a new model. Additionally, in a CO–LN scenario, systems having no speaker model for speaker with ID id_{rec} train a new model independently of the collaborative new speaker detection outcome.

4.2.3. Speaker ID mapping for collaboration

Speaker IDs are created locally by a personal identification system to uniquely identify speaker models. Since they are generated independently, speaker IDs are not comparable between systems. For collaborative recognition and new speaker detection, a relation of speaker IDs between systems of a collaboration is nevertheless needed.

Our algorithm approach targets to obtain a mapping between speaker IDs of one system and those of another one. The algorithm compares all speaker models of these systems to create the ID mapping. For this purpose, distances between two models are computed and used as a metric denoting model similarity.

As detailed in Section 4.1.2, we model speaker phonemes using a codebook $C = \{\mathbf{c}_i\}_{i=1}^{K}$, which is a set of K code vectors $\mathbf{c}_i$. The distance between two models $C_1 = \{\mathbf{c}_{1i}\}_{i=1}^{K}$ and $C_2 = \{\mathbf{c}_{2i}\}_{i=1}^{K}$ is

$$D(C_1, C_2) = \frac{1}{2}\left\{\sum_{i=1}^{K} \min(\mathbf{c}_{1i}, C_2) + \sum_{i=1}^{K} \min(C_1, \mathbf{c}_{2i})\right\}. \tag{4}$$

Subsequently we distinguish two mapping types for local-identical (LI) and local-nonidentical (LN) speaker sets. For LI-sets, the two systems are assumed to have speaker models for the same speaker set. In contrast, LN-sets allow systems to have models for arbitrary independent subsets of relevant speakers.

4.2.3.1. LI-set mapping. In LI-sets, there exists for each model of a system exactly one corresponding model in the other system, thus resulting in a one-to-one mapping. Algorithm 1 illustrates the pseudo code for LI-set mapping. The algorithm first calculates all distances between models of both systems. Subsequently, models with minimum distances are determined under the one-to-one mapping constraint.

Algorithm 1 Speaker ID mapping for local-identical (LI) speaker ID sets.

1. Create distance matrix $D_{i,j} = D(C_i^1, C_j^2)$ of all distances between models of the first system ($C_i^1\ \forall i = 1, \ldots, N_1$) and models of the second system ($C_j^2\ \forall j = 1, \ldots, N_2$)
2. $k = 1$
3. Search in matrix $D_{i,j}$ for the indices of the smallest element: $(id_1, id_2) = \text{argmin}_{i,j}(D_{i,j})$
4. Store the indices as a new mapping: $\text{map}(k) = (id_1, id_2)$
5. Remove all distances of models C_{id_1} and C_{id_2} from $D_{i,j}$
6. $k = k + 1$
7. If notEmpty($D_{i,j}$). Then goto(3) Else return(map)

4.2.3.2. LN-set mapping. In LN-sets, not every model of one system has a corresponding model in another system. Speaker models may be missing in one of the systems and thus no connection to another system's models can be made.

For LN-sets we used a two step approach to derive a mapping. The first step consists of applying the LI-set algorithm (see Algorithm 1). As result, a mapping under the one-to-one mapping constraint $N_{\min} = \min(N_1, N_2)$ is obtained, with N_1

and N_2 numbers of models in each systems' databases. Thus, $|N_1 - N_2|$ models without a mapping can be identified by this step. In a second step, all $N_{\min}$ mappings are tested. A mapping between C_i^1 and C_j^2 is accepted or rejected with the decision function:

$$f_{\mathrm{NMD}}(C_i^1, C_j^2) = \begin{cases} 1, & \text{if } D(C_i^1, C_j^2) \leq \Delta_{\mathrm{MND}} \\ 0, & \text{else.} \end{cases} \tag{5}$$

Here Δ_{NMD} is the decision function's threshold. If a mapping distance is less than or equal to Δ_{NMD}, this mapping (i, j) is accepted, otherwise it is removed from the mapping table. The collaborative identification algorithms can handle NL-set mappings as illustrated in the example shown in Section 4.2.1.

5. Evaluation dataset

We selected the freely available Augmented Multiparty Interaction (AMI) corpus [25] for evaluations of our approach, which ensures reproducibility of analysis results. This dataset provides more than 200 individual English speakers and contains ∼100 h of conversation/meeting scenes recorded from ambient far-field microphones and close-talk lapel microphones worn by each participant. The corpus is subdivided into meeting sets, each containing four meetings of a group of four participants. Two meeting types were recorded and transcribed: ad-hoc and scenario-based meetings, where people had been briefed to talk about a particular topic beforehand.

We targeted to evaluate situations where up to 24 speakers are involved. Thus, to analyze the performance of our collaborative systems approach, we extracted speech data from six meeting sets of the original corpus, in total 24 speakers (9 female, 15 male). We used a mixture of ad-hoc and scenario-based meetings.

To maximize the analysis dataset we extracted 8 min of speech for each speaker out of the four separate meetings in a set. We used audio data recorded from individual lapel microphones to realistically match the situation of a wearable speaker annotation system. Every single lapel microphone system acquired the signal of the owning speaker as well as that of other speakers in the meeting. Thus, for each speaker four recording channels exist, which represent the same situation. With this setup, we could simulate up to four parallel collaborative identification systems in one meeting.

The AMI corpus provides annotations for words and other sounds, however it lacks ground truth information on the actual speaker. Consequently we annotated each individual speaker of the selected dataset. In total, our evaluation is based on 192 min of annotated speech data. Speech segments that were annotated by AMI as cross-talk and non-speech gaps of larger than 1 s were omitted. The audio files of AMI were originally recorded with 16 kHz. We downsampled the data to 8 kHz, since this band provides the most relevant speaker information. An anti-aliasing FIR filtering was applied prior to downsampling.

As described in Section 4.1.2, a new speaker model was trained with a speech segment of $t_{\mathrm{train}} = 20$ s. This results in 24 training segments for the 8 min speech data available from each speaker.

To evaluate the collaborative operation, speaker databases for N_{sys} collaborative systems were generated. Each system database consisted of N_{sp} models, where each model was created using training segments of the system-specific channel. The collaboration performance was then evaluated on the remaining data. Recognition was performed on speech segments of $t_{\mathrm{rec}} = 5$ s (as described in Section 4.1.2). The total accuracy for $N_{\mathrm{sys}} = \{1, 2, 3, 4\}$ and $N_{\mathrm{sp}} = \{4, 8, 12, 16, 20, 24\}$ was calculated by simulating all possible combinations.

6. Results

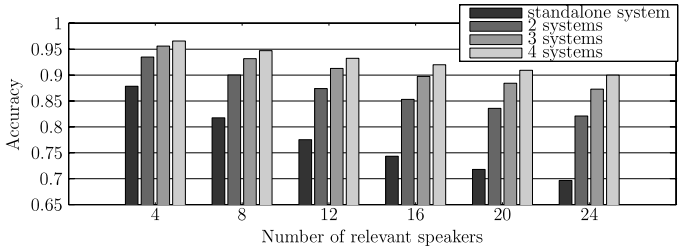
This section details evaluation results for collaborative speaker identification in the use scenarios, presented in Section 3. We utilized the approach described in Section 5 to evaluate the collaborative use scenarios.

For evaluation we used Matlab and Simulink as our simulation environment. The speaker identification system, as detailed in Section 4, was implemented in Simulink. Collaborations between the systems were simulated in Matlab using the Simulink model. We focus on evaluating performance bounds for these use scenarios. Sections 6.1–6.3 present the results for the use scenarios CC–LI, CO–LI, and CO–LN. In Section 6.3 performance of the three use scenarios are compared. Finally, in Section 6.5 results for acquiring a speaker ID mapping are presented.

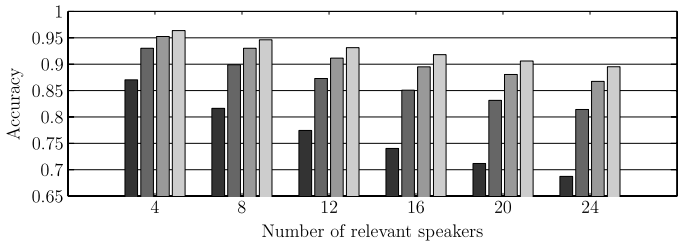
6.1. Collaborative-closed, local-identical (CC–LI) analysis

In a CC–LI use scenario, collaborative systems share speaker matching distances during recognition for all relevant speakers. In a collaborative speaker recognition, these results are weighted, resembling a voting by all collaborative systems.

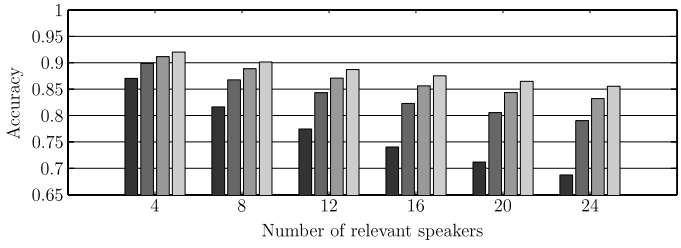
Fig. 5(a) shows the collaborative speaker recognition performance in a CC–LI use scenario for 2–4 collaborating systems, and with the standalone system performance as reference. Relevant speaker sets were varied between 4 and 24. These results show that performance continuously increases with the number of collaborating systems. Hence, collaboration in a CC–LI scenario provides a clear benefit compared to a standalone system's performance. Moreover, benefits of the collaboration are larger for settings with a high number of relevant speakers. With four relevant speakers, a collaboration of four systems can increase accuracy from 0.88 to 0.97 (+9%), whereas with 24 relevant speakers an improvement from 0.70 to 0.90 was observed (+20%).



(a) CC–LI use scenario.



(b) CO–LI use scenario.



(c) CO–LN use scenario.

Fig. 5. Performance of multi-system collaboration in the three use scenarios, in comparison to a standalone system. Performance is shown for varying the number of relevant speakers. Fig. 5(a) shows the results for the CC–LI case, Fig. 5(b) for CO–LI, and Fig. 5(c) for CO–LN.

6.2. Collaborative-open, local-identical (CO-LI) analysis

For CO-scenarios, collaboration is performed in speaker recognition and new speaker detection functions. We evaluated two CO-scenarios regarding local speaker sets, thus CO–LI and CO–LN, as introduced in Section 3. In a CO–LI use scenario, collaborative recognition was performed as in the CC-scenario, presented above: all collaborating systems share their matching distances during recognition. However, since the tested speakers can be unknown to all collaborating systems, a collaborative new speaker detection was performed subsequent to the recognition step.

Fig. 5(b) presents the results of CO–LI. Here, performance of 2–4 collaborating systems are compared to a standalone system, for varying the number of relevant speakers. We analyzed the CO–LI condition by iteratively leaving each speaker out of the collaborative set once. Models of all other relevant speakers were maintained in the databases. This evaluation provided a worst-case performance of the new speaker detection, since each left-out speaker should be detected as a new one, while all other models were available.

Similar to the CC-scenarios presented above, collaboration in CO–LI use scenario improves standalone system performance. Clear performance increases were observed for large numbers of relevant speakers. Four collaborative systems in a setting with 4 relevant speakers improved accuracy from 0.87 to 0.96 (+9%). In a setting with 24 relevant speakers, the improvement was from 0.69 to 0.90 (+21%).

6.3. Collaborative-open, local-nonidentical (CO-LN) analysis

CO–LN is the most general collaboration use scenario. Here, the collective collaborates for speaker recognition as well as for new speaker detection. As opposed to CO–LI, systems miss relevant speaker models to fully collaborate. Consequently, collaborative recognition and new speaker detection need to rely on unbalanced voting. A distance normalization helped to reduce system dependency, as described in Section 4.2.1.

Fig. 5(c) presents the results for CO–LN. In this analysis, the probability that a test speaker is known by N_{known} systems was assumed to be $p_{\text{known}}(N_{\text{known}}) = \frac{1}{N_{\text{sys}}+1}$ for $N_{\text{sys}} = \{0, \ldots, N_{\text{sys}}\}$.

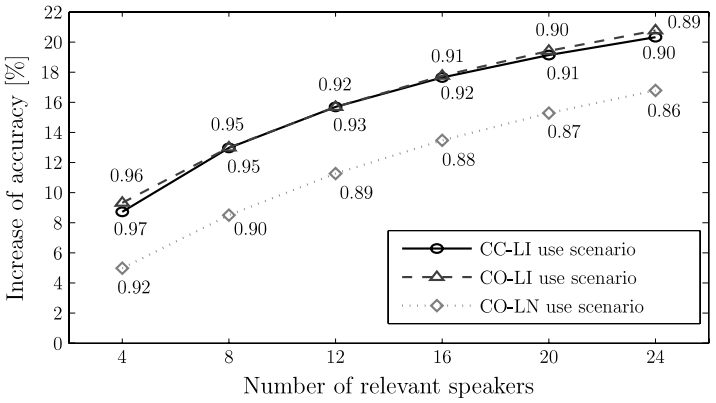


Fig. 6. Comparison of all three use scenarios with 4 collaborating systems. Performance is shown as accuracy gains through collaboration for varying the number of relevant speakers. Points are denoted by their absolute accuracy.

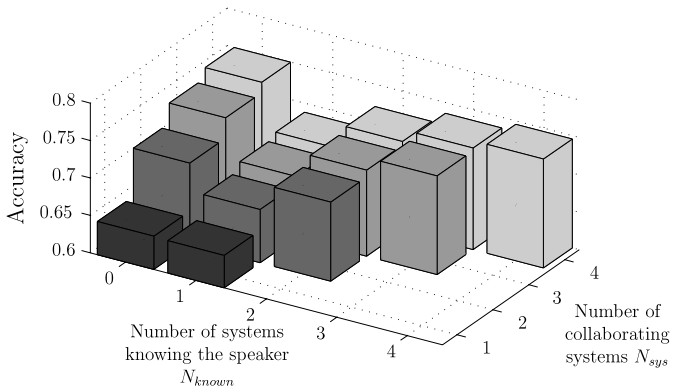


Fig. 7. Performance of multi-system collaboration in CO-based use scenarios with 24 relevant speakers, in comparison to a standalone system. Recognition accuracy is shown for 0–4 collaborative systems. The number of systems known the speaker was varied from 0 to 4.

In a setting with 4 relevant speakers, 4 collaborative systems improve accuracy from 0.87 to 0.92 (+5%), whereas with 24 relevant speakers, accuracy is improved from 0.69 to 0.85 (+16%). We attributed the lower performance of CO–LN as compared to CO–LI to the reduction of collaboration information.

6.4. Collaborative scenario comparison

A comparison of collaboration performance among all use scenarios is shown in Fig. 6. Here, performance gains for 4 collaborating systems are shown for varying the number of relevant speakers. It can be observed that gains for both LI-set based use scenarios are similar. In contrast, gains for CO–LN are ∼4% lower.

Fig. 7 presents a detailed performance analysis for collaborations in CO-based use scenarios with 24 relevant speakers. The identification performance was analyzed here regarding the number of collaborative systems knowing the tested speaker, N_{known}, and the total number of collaborating systems N_{sys}. We observed that performance improvements of collaborating systems strongly depends on N_{known}. With 4 collaborative systems in CO–LI, only performances with $N_{\text{known}} = \{0, 4\}$ are relevant. For $N_{\text{known}} = 0$ only the new speaker detection was activated, since no models existed. Thus, this condition does not reveal speaker IDs and can be seen as starting point to learn new models.

The CO–LI results show a performance boost through collaboration. In CO–LN, performance for $N_{\text{known}} : 0 < N_{\text{known}} < N_{\text{sys}}$ are relevant as well. Here, lower performance improvements can be observed due to the challenge of nonidentical system databases. For increasing N_{known}, performance improves, as expected. Moreover, for constant N_{known}, increasing the number of collaborating systems leads to performance gains as well. This is due to the fact that systems which do not know a speaker help indirectly to elevate the correct speaker model (see Section 4.2.1 for an example illustrating this system behavior).

6.5. Speaker ID mapping

The developed speaker ID mapping algorithms are described in Section 4.2.3. To evaluate speaker ID mapping performance, two speaker model databases were created by using training segments of N_{sp} speakers from two different

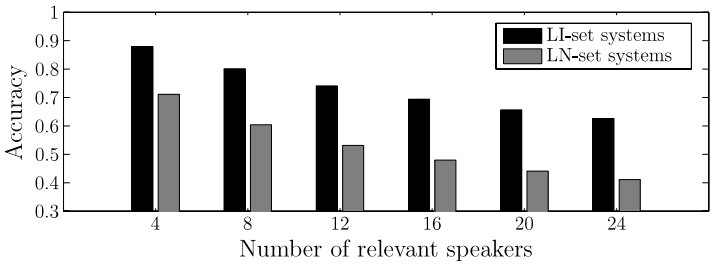


Fig. 8. Performance analysis of speaker ID mapping between two databases for LI- and LN-set systems. The performance is shown for different numbers of relevant speakers in one meeting.

channels. These generated databases were used with the mapping algorithms to determine mapping error counts. We simulated every possible combination of pairs of databases to evaluate mapping performance for $N_{sp} = \{4, 8, 12, 16, 20, 24\}$. For LN-set speaker ID mapping, we compared two model databases, once with all speakers available in both databases and subsequently with a missing speaker in one of the two databases, where each speaker was removed once.

Fig. 8 shows the mapping performance obtained between two systems for LI- and LN-sets and for varying the number of relevant speakers. For both mapping algorithms, performance decreases with increasing relevant speakers in the databases. For the LI-set, using one-to-one mapping, accuracy drops from 0.88 for 4 relevant speakers to 0.63 for 24 relevant speakers. LN-set mapping performance drops from 0.71 to 0.41 for 4 and 24 relevant speakers respectively. These results clearly show the benefit of using the less complex LI-set mapping, where an unknown model detection algorithm is not needed. It can be concluded that the LI-set mapping can certainly be used with 12–16 relevant speakers in one meeting at an accuracy $\geq$0.7. Using the LN-set mapping, relevant speakers in one meeting are constrained to 4, at an accuracy $>$0.7.

7. Discussion

Our evaluation revealed that a collaboration on speaker recognition and new speaker detection among personal speaker identification systems can substantially increase performance. Results for CC–LI and CO–LI show gains of $\sim$20% at 24 relevant speakers.

7.1. Information exchange in collaboration

Collaboration in mobile and wearable systems is constrained by wireless communication bandwidth and power consumption. To this end, collaboration could be performed by fusing information at different levels of the processing stack, including raw audio data, processed sound features, recognition and detection result, and speaker model levels. Clearly, a viable collaboration concept for mobile systems should make use of a compressed information exchange. However, this inherently limits collaborative information to improve performance.

For our system architecture, fusion at raw data, processed sound features, and speaker model levels would require a collaborating system to transmit net data rates of 128 kbit/s, 83.4 kbit/s, and 12.29 kbit/model, respectively. In contrast, information fusion at the level of speaker recognition and new speaker detection requires 64 bits each for recognition and new speaker detection. As both were calculated every 5 s, a bandwidth far below the rates stated above is required.

7.2. Challenges in collaborative identification systems

Channel properties are a critical concern for collaborative speaker identification systems that do not share hardware. In combination with differences in speaker distance and room reflection effects, the recorded sound data and derived speaker models could differ substantially. This condition critically constrains collaboration options and required a speaker ID mapping. e.g. it is not feasible to compare complete databases between systems. Our choice to solely exchange recognition and detection results, reflects these constraints.

When owners of personal speaker identification systems enter into a conversation or meeting, their systems need to perform an initial speaker ID mapping. In our approach, this mapping enables a collaboration. Our performance results show that this mapping is feasible. However, the mapping is more challenging for LN-set systems, in which individual systems have different speaker databases. While the implementations presented in this work would permit collaborations with 16 relevant speakers in LI-set systems, it is limited to 4 relevant speakers for LN-set systems at a bound of 70% accuracy. Although further work is needed to improve speaker ID mapping performance, this function is not often used. Typically, a speaker ID mapping would be performed upon initiating a collaboration only, e.g. at the beginning of a meeting. Any subsequent ID mapping, e.g. when a new speaker was detected, would use the collaborative new speaker detection to determine the ID mapping.

Noise disturbing the speech signal is a key challenge in speaker recognition. This work did not focus on analyzing the effect of noise in particular. Our evaluation dataset was however composed from real indoor meetings, including typical noise

levels (e.g. street, noise from participants). Thus, all performance results presented reflect natural meeting environments. A further, dedicated noise analysis could reveal additional benefits of our collaboration approach. Often audio channels of individual sound recording systems have different noise and channel properties. Thus, collaborative systems could improve individual identification performance in environments with high background noise even more than in rather silent settings.

We assumed in this work that the analyzed audio data contains speech information only. We expect that a voice activity detection (VAD) procedure can be used to perform an a priori speech segmentation.

While the system can operate robustly with our chosen training time, a faster enrollment may be desirable. For this purpose the GLA algorithm would need to be replaced by another clustering approach that permits an incremental model creation. A weaker model could then serve to recognize speakers during the first few seconds already.

7.3. Prototype implementation

Our personal speaker identification and collaboration approach is designed to be used with standard smart phones. Such mobile and wearable devices are limited in processing capabilities and power consumption. Thus, minimizing algorithmic complexity and communication bandwidth between collaborative partners is essential.

In our previous work we confirmed the feasibility of a speaker identification and learning system working in standalone mode [5]. This system was implemented on a custom wearable device prototype, based on a TI TMS320C67 DSP, audio interface, USB host connection, and battery power supply. The system was designed to be worn as belt attachment. With this system we were able to train and recognize up to 150 speakers in real-time. The device could continuously operate for 8.6 h between battery recharges.

We expect that this system could be extended to operate in collaborative mode as targeted in this work by adding the 'collaborative identification' function block and a wireless transceiver. Given that every system is sending 16 bit/s and conversation partners are in a range of typical meeting room sizes of about 15 m, we expect that ultra low-power radio solutions, such as ZigBee are feasible. Since ultra low-power transceivers are not yet common for smart phones, Bluetooth could be used as intermediate alternative for collaborative communication.

8. Conclusions and future work

In this work we introduced a collaborative personal speaker identification approach that can be generally applied in different use scenarios. Due to the diversity of situations in which a mobile or wearable identification system can be used, operation conditions and collaboration options vary widely. For this purpose we introduced a collaboration use scenario concept that accounts for unknown speakers and independent speaker model databases of participating systems. Our analysis confirmed that the scenarios have practical applications in different conversation and meeting situations. Furthermore, evaluations of different use scenarios showed that our speaker identification system provides useful performance in standalone and collaborative operation modes.

When compared to a standalone operation, the collaboration among four personal identification systems increased system performance. Gains were up to 9% at 4 relevant speakers and up to 21% at 24 relevant speakers for systems with locally identical speaker sets. For the most challenging scenario of collaborative open and locally nonidentical speaker sets, still gains of 5% and 16% at 4 and 24 relevant speakers respectively, were achieved. We concluded that both collaborations, to recognize known speakers and to detect new speakers, provide substantial benefits regarding system robustness.

From our performance comparison among collaboration scenarios we concluded that allowing unknown speakers in a conversation does not hamper system performance and gains achieved through collaboration. In contrast, allowing systems to have nonidentical speaker sets clearly reduced collaboration gains. Moreover, we found that our collaborative fusion provides benefits even in situations, where only one system knows the actual speaker. In this situation, collaborating systems indirectly elevate the correct speaker by returning low matching scores for their models.

We specifically developed the system architecture to cope with all use scenarios considered during system evaluation. Moreover, system architecture and implementation considered the requirements of mobile and wearable systems regarding communication and algorithm complexity. In particular, efficient solutions were found to exchange collaboration information while minimizing bandwidth requirements. The choice for exchanging speaker recognition and detection results represents a tradeoff between system dependency, due to channel properties and information detail. We concluded that a collaborative personal speaker identification system can be realized with currently available audio, communication, and processing capabilities in mobile devices.

With the performance of smart phones, our speaker identification approach, using one microphone only, could be implemented to realize convenient personal annotation systems. Further work should address speaker ID mapping approaches to optimize the performance of ad-hoc mappings when system owners enter into a conversation or meeting. Additionally, the benefit of collaboration in different noise environments should be investigated.

Acknowledgement

This work was supported by the EU project SENSEI, contract number 215923 (www.sensei-project.eu).

References

[1] T.K. Choudhury, A. Pentland, The sociometer: a wearable device for understanding human networks, in: Proceedings of ACM Conference on Computer Supported Cooperative Work, CSCW 2002, Workhop on Ad Hoc Communications and Collaboration in Ubiquitous Computing Environments, 2002.
[2] N. Kern, B. Schiele, H. Junker, P. Lukowicz, G. Tröster, Wearable sensing to annotate meeting recordings, Personal Ubiquitous Computing 7 (5) (2003) 263–274.
[3] J. Gemmell, G. Bell, R. Lueder, Mylifebits: a personal database for everything, Communications of the ACM 49 (1) (2006) 88–95.
[4] M. Blum, A. Pentland, G. Troster, Insense: interest-based life logging, IEEE Multimedia 13 (4) (2006) 40–48.
[5] M. Rossi, O. Amft, M. Kusserow, Troster, Collaborative real-time speaker identification for wearable systems, in: Eighth Annual IEEE International Conference on Pervasive Computing and Communications, PERCOM'10, 2010.
[6] Smart meeting room, Dalle Molle Institute. http://www.idiap.ch/scientific-research/smart-meeting-room.
[7] ICSI smart meeting room, Berkeley. http://www.icsi.berkeley.edu/Speech/mr/.
[8] Y. Chen, Y. Rui, Real-time speaker tracking using particle filter sensor fusion, Proceedings of the IEEE 92 (3) (2004) 485–494.
[9] D. Gatica-Perez, G. Lathoud, J.-M. Odobez, I. McCowan, Audiovisual probabilistic tracking of multiple speakers in meetings, IEEE Transactions on Speech and Audio Processing 15 (2) (2007) 601–616.
[10] T.K. Choudhury, Sensing and modeling human networks, Ph.D. Thesis, Massachusetts Institute of Technology, 2004.
[11] D. Olguin, P.A. Goor, A. Pentland, Capturing individual and group behavior withwearable sensors, in: Proceedings of AAAI Spring Symposium on Human Behavior Modeling, 2009.
[12] T. Kim, O. Brdiczka, M. Chu, J. Begole, Predicting shoppers' interest from social interactions using sociometric sensors, in: Extended Abstracts of 27th Annual CHI Conference on Human Factors in Computing Systems, CHI 2009, 2009.
[13] U. Anliker, Speaker separation and tracking, Ph.D. Thesis, Swiss Federal Institute of Technology Zurich, 2005.
[14] D. Charlet, Speaker indexing for retrieval of voicemail messages, in: Proceesings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'02, vol. 1, 2002, pp. 121–124.
[15] L. Lu, H.-J. Zhang, Speaker change detection and tracking in real-time news broadcasting analysis, in: Proceedings of the Tenth ACM International Conference on Multimedia, MULTIMEDIA'02, ACM, New York, NY, USA, 2002, pp. 602–610.
[16] S. Kwon, S. Narayanan, A method for on-line speaker indexing using generic reference models, in: Proceedings of the 8th European Conference on Speech Communication and Technology, EUROSPEECH 2003, 2003.
[17] D. Lilt, F. Kubala, Online speaker clustering, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'04, vol. 1, 2004, pp. 333–336.
[18] T. Matsui, S. Furui, Comparison of text independent speaker recognition methods using VQ distortion and discrete/continuous HMMS, in: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 1992, 1992.
[19] J.-S. Lee, Y.-W. Su, C.-C. Shen, A comparative study of wireless protocols: bluetooth, UWB, zigbee, and wi-fi, in: Proceedings of the IEEE Industrial Electronics Society, IECON 2007, 2007, pp. 46–51.
[20] J.R. Deller, J.G. Proakis, J.H. Hansen, Discrete-Time Processing of Speech Signals, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2000.
[21] M. Do, M. Wagner, Speaker recognition with small training requirements using a combination of VQ and DHMM, in: Proceedings of Speaker Recognition and its Commercial and Forensic Applications, 1998, pp. 169–172.
[22] Y. Linde, A. Buzo, R. Gray, An algorithm for vector quantizer design, IEEE Transactions on Communications 28 (1980) 84–95.
[23] R.A. Finan, A.T. Sapeluk, R.I. Damper, Impostor cohort selection for score normalisation in speaker verification, Pattern Recognition Letters 18 (9) (1997) 881–888.
[24] A.M. Ariyaeeinia, J. Fortuna, P. Sivakumaran, A. Malegaonkar, Verification effectiveness in open-set speaker identification, IEE Proceedings-Vision, Image and Signal Processing 153 (2006) 618–624.
[25] E. funded AMI project, The AMI Meeting Corpus, 2008. http://corpus.amiproject.org/.