This is me: Using Ambient Voice Patterns for In-Car Positioning

Michael Feld[1], Tim Schwartz[2], Christian Müller[1]

[1] German Research Center for Artificial Intelligence (DFKI),
Intelligent User Interfaces Department
[2] MMCI Cluster of Excellence, Saarland University
Saarbrücken, Germany

Abstract. With the range of services that can be accessed inside a car constantly increasing, so are the opportunities for personalizing the experience for both driver and other passengers. A main challenge however is to find out who is sitting where without asking explicitly. The solution presented in this paper combines two sources of information in a novel way: Ambient speech and mobile personal devices. The approach offers improved privacy by putting the user in control, and it does not require specialized positioning technologies such as RFID. In a data-driven evaluation, we confirm that the accuracy is sufficient to support a ten-speaker scenario in practice.

Keywords: Positioning, Speaker Recognition, Automotive HMI, User Modeling, Personalization

1 Introduction

[5] introduces a scheme of positioning systems, which distinguishes exocentric and egocentric approaches. With exocentric approaches, a device carried on the user sends a unique signal to the environment that is used to calculate the current position of the device. The result is then sent back to the user like it is the case with cell-phone positioning based on cell-id. GPS is an example of the other category, egocentric: the device receives data from the environment and calculates the position by itself, which has clear advantages with respect to privacy, since the users can control if they want to share their position information with a third-party. The system described in this paper is a hybrid approach, since sensors – in this case microphones – are installed in the car and pick up signals generated by the user, e.g. casual conversations or voice commands. With this information the car's system can determine which seats are taken, but it cannot identify *who* is sitting there. The information that enables this identification is stored on the user's personal device. Thus, the device is the link between the location information (provided by the car) and the user information (stored on the device) (see Fig. 2 (*left*)). By combining an ambient information source with existing technology, positioning can occur in a fully automated and non-intrusive way.

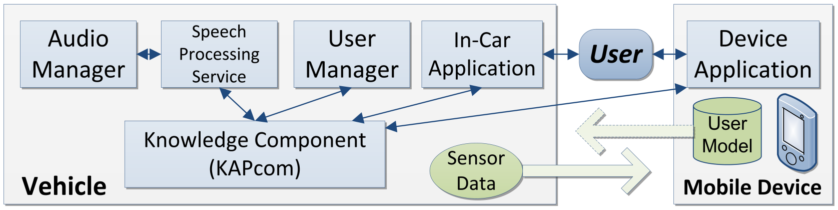


Fig. 1. The In-Car Positioning Architecture. Components are running either in the *vehicle* or *mobile device* context.

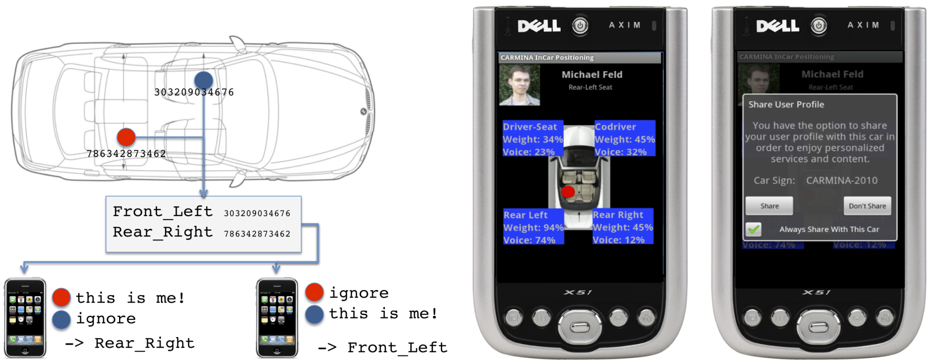
Knowing what seat a passenger occupies enables a multitude of possibilities for personalization. One safety-related application are adaptive driver warnings [1]. Other features such as controlling car comfort functions benefit as well from knowing who sits where (e.g. temperature being configured to the user's preferred setting).

In the remainder of this paper, the in-car positioning approach based on ambient speech is described from a technical point of view. Since the reliability of such a system strongly depends on the accuracy of the speaker recognition process, empirical data is provided in Section 3. The major question is: how much speech do we need to reliably make that decision? As we will show, the system needs approximately one second of speech.

2 The Proposed In-Car Positioning Approach

The actual architecture of the proposed system consists of several hardware and software components that have all been installed into a real demonstration vehicle (Mercedes R-Class). The hardware set-up consists of two 7" screens (front seats located, center console) as well as two 10" screens (rear seats, head rest). All seats are equipped with hi-fidelity directional microphones (*Sennheiser ME 105-NI*). A Wireless LAN allows nomadic devices to easily join the car network. In our case, several Windows Mobile-based smartphones and PDAs represent the drivers' personal devices running the mobile client software.

Fig. 1 illustrates the software architecture. The central component that links together the other modules is the *Knowledge, Adaptation and Personalization component (KAPcom)*. It is a multi-purpose service maintaining a knowledge base with an automotive-domain ontology. Different applications can retrieve and store knowledge, e.g. user preferences or traffic information. In the positioning scenario, *KAPcom* is used as a "blackboard" – a design pattern for problem solving often applied in Artificial Intelligence [2]). On the data acquisition side, an *Audio Manager* obtains raw audio data from the microphones. It then streams the data to a *Speech Processing Service* responsible for detecting speech and – if speech is present – for computing a set of characteristic high-level audio features, the so-called voiceprint (see also Section 3). This evidence is stored in the knowledge base together with the information on which seat it was recorded.



You have the option to share your user profile with this car in order to enjoy personalized services and content.

Car Sign: CARMINA-2010

☑ Always Share With This Car

Fig. 2. *Left:* In-Car positioning. Sensor data in the environment (car) is transmitted to the user device(s). There, it is used as a key in order to find out what value belongs to the respective user. *Middle*: The system has identified its owner as the passenger on the rear left seat. *Right*: After positioning, the system asks if the user wants to share their data with the car.

When a mobile phone or PDA connects to the car's network, it automatically starts receiving notifications using a mobile client application when new data become available. Given the user profile that is stored on the personal device, it can then locally perform the detection step with the sensor data from any seat, i.e. determine in how far a given voiceprint matches the signature stored in the user profile. Data is aggregated until a match can be confirmed with reasonable confidence. Afterwards, the device owner's location inside the car is known and the client can use this knowledge to perform device-side adaptation, such as changing the phone's ringing scheme and other notifications when the user is the driver. The current UI on the device (Fig. 2 (*middle*)) provides a visualization of the match likelihood for the evidence coming from each seat. The device owner can additionally decide to share his identity with the car, thereby allowing applications running on its system to also take advantage of personalization. Another component running in the car: the User Manager manages the association of user profiles to seats based on evidence and authentication data.

The proposed approach differs from other positioning methods in that it does not require the user's device to broadcast information, hence adopting the privacy advantage of egocentric positioning. However, when the positioning data is used to adapt car functions, information sharing concepts are needed to deal with the potential issues. Most critically, users have the option to block all data sharing in the first place on their device. When information should be shared, the situation may depend on whether the user is the owner of or a regular passenger in the car, or not. If so, the user profile can either be automatically transmitted by default without user interaction, or it can be stored on the car or even an off-board service and be only activated by a token on a device.

3 Generating and Testing Speaker Models

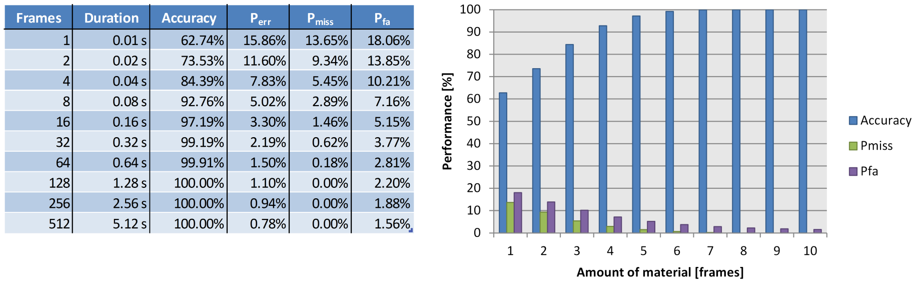
This work employs a speaker detection method that is motivated by creating a resource-efficient detection scheme for the mobile device implementation and an interchangeable model / voiceprint representation.

A basic GMM-UBM approach [4] makes the core of the recognizer. In this approach, speaker models are represented by Gaussian Mixture Models (GMMs), each model consisting of a fixed number of Gaussian probability distributions (in our case 1024) for each feature. The features are 12 coefficients of the mel-frequency cepstrum (MFCCs) indicating the short-term power spectrum of the voice. The GMMs are trained on pre-recorded speech and stored on the device in a compact binary format of approx. 200 KB each. In addition, a "background model" is trained from the speech of all speakers. The tool used to facilitate the training process is the speaker classification framework [3].

For any given audio stream from a live system and user model, a likelihood score can be computed that indicates to what extent the speech, which is segmented into frames, matches the model. The final acceptance of a speaker model occurs when a predefined minimum amount of material matches the speaker model more than the background model. In order to reduce sensitivity to noise and silence, a filter for voiced frames has been applied to all audio.

For training and evaluation, a corpus was recorded inside the standing vehicle described earlier. The corpus consists of ten adult speakers (7 male, 3 female) and a total of 76 minutes of material categorized by different conditions (seat, read vs. spontaneous, overlapping, doors closed vs. open). The training set consists of 30 minutes of speech, while the evaluation corpus uses 20 seconds (2000 frames) per speaker. There are two types of performance measures applied in this task: accuracy and error rates. The *accuracy* describes what percentage of test samples are assigned to the correct speaker, i.e. it measures how well the system can *identify* the correct speaker out of the full set of (in this case) ten speakers. The second category quantifies the errors a system can make when it tries to *detect* if a test sample is from a given target speaker, which are *misses* and *false alarms*. A mobile device as in our scenario only has a single user profile stored and hence performs the *detection* task. The false alarm rate is possibly the most critical measure here because it tells in how many cases the device will incorrectly report a match. Our evaluation therefore takes into account different numbers of frames over which scores were averaged.

Fig. 3 illustrates the results. In a single-frame scenario (10 ms of speech), already 62.7% of frames are classified correctly at a chance level of 10%. However, it is clearly evident that recognition rates improve significantly when more speech becomes available. At 64 frames, which corresponds to 640ms of pitched voice (roughly a second of ordinary speech), almost 100% accuracy are achieved in this test. The corresponding chance of a false alarm is still 2.8%. It can be lowered to 1.6% when using approx. one fourth of the eval material for averaging.



Frames	Duration	Accuracy	P_{err}	P_{miss}	P_{fa}
1	0.01 s	62.74%	15.86%	13.65%	18.06%
2	0.02 s	73.53%	11.60%	9.34%	13.85%
4	0.04 s	84.39%	7.83%	5.45%	10.21%
8	0.08 s	92.76%	5.02%	2.89%	7.16%
16	0.16 s	97.19%	3.30%	1.46%	5.15%
32	0.32 s	99.19%	2.19%	0.62%	3.77%
64	0.64 s	99.91%	1.50%	0.18%	2.81%
128	1.28 s	100.00%	1.10%	0.00%	2.20%
256	2.56 s	100.00%	0.94%	0.00%	1.88%
512	5.12 s	100.00%	0.78%	0.00%	1.56%

Fig. 3. Results of the in-car speaker recognition on a test corpus. The error rate P_{err} is defined as the mean of P_{miss} and P_{fa}.

4 Conclusion and Future Work

As documented in the previous section, the approach presented herein can indeed be used to rather quickly and reliably determine speakers' positions in a setting with a manageable number of speakers. Yet, since positioning/identification solely on voiceprints has some drawbacks, e.g. the passengers have to speak in order to be positioned, we will expand the presented system with weight sensors. Here, the basic idea is, that the users' profiles will be enriched by their body weight data. This value will of course change, even throughout the day, so appropriate estimation models will have to be found and evaluated. Both approaches, speaker recognition and weight measurements, will then be combined according to the "Always Best Positioned" paradigm, which was proposed in [5].

References

1. Brouwer, R.F.T., Hoedemaeker, M., Neerincx, M.A.: Adaptive interfaces in driving. In: HCI (16). pp. 13–19 (2009)
2. Corkill, D.D.: Blackboard systems. AI Expert 6, 40–47 (1991)
3. Feld, M., Müller, C.: An Integrated Development Environment for Speech-Based Classification. In: Proceedings of the 13th International Conference "Speech and Computer" SPECOM 2009. pp. 443 – 447. St. Petersburg, Russia (June 2009)
4. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted gaussian mixture models. In: Digital Signal Processing. p. 2000 (2000)
5. Schwartz, T., Stahl, C., Baus, J., Wahlster, W.: Resource-Adaptive Cognitive Processes, chap. Seamless Resource-Adaptive Navigation, pp. 239 – 265. Cognitive Technologies, Springer (2010)