WISS, a Speaker Identification System for Mobile Robots

François Grondin and François Michaud

***Abstract*— This paper presents WISS, a speaker identification system for mobile robots integrated to ManyEars, a sound source localization, tracking and separation system. Speaker identification consists in recognizing an individual among a group of known speakers. For mobile robots, performing speaker identification in presence of noise that changes over time is one important challenge. To deal with this issue, WISS uses Parallel Model Combination (PMC) and masks to update in real-time the speaker models (obtained in clean conditions) to both additive and convolutive noises. The results show that the weighted rate of good speaker identifications is 96% on average for a Signal-to-Noise Ratio (SNR) of 16 dB, whereas it only decreases to 84% when the SNR drops to 2 dB.**

I. INTRODUCTION

Autonomous interactive robots must be able to perceive and analyze sounds from real life settings. The focus so far has mostly been on speech recognition (*what* is being said [1], [2], [3], [4]) or sound localization (*where* is it coming from [5], [6]), but not so much on speaker identification (*who* is speaking), especially in noisy conditions. A 8-microphone array system for mobile robots that uses soft channel selection can identify a speaker among a group of 30 known speakers at a recognition rate of 90% when the robot is 2 meters away from the speaker, and 75% when it is 3 meters away [7]. Experiments are conducted in a quiet environment but the Signal-to-Noise Ratio (SNR) is not given. Another system identifies individuals with both their voice and facial features [8]. This fusion technique gives recognition rates which range from 88.25% when the speaker is three meters away from the robot, to 99.5% when the speaker is one meter away in clean, noiseless conditions. The SNR is not given in this experiment.

This paper presents a speaker identification system designed to work on mobile robots operating in noisy environments. The system is named WISS (for Who IS Speaking). WISS extends Many Ears, an artificial auditory system for mobile robots which uses an array of eight microphones [1], [5], [2]. ManyEars provides enhanced speaker signal for improved recognition in real world settings. Its low complexity and its capacity to localize, track and separate many simultaneous sound sources make it ideal for real life scenarios. ManyEars has been used on different platforms including Spartacus [9], SIG2 [3] and ASIMO [4], and is usually exploited as a pre-processing module for a speech recognition engine. WISS exploits the separated sound sources processed by ManyEars and already represented in the frequency domain, reducing the computational load. To deal with noisy conditions, WISS estimates the additive and convolutive noises from the speech signals obtained from ManyEars, and updates the speaker models (derived from training signals generated in clean conditions) to match the noisy environment.

The objective of the paper is to characterize WISS' speaker identification performance according to the SNR, to show that a robust speaker identification is possible in a noisy environment. The paper is organized as follows. Section II explains WISS, followed by Section III with results obtained and Section IV concludes the paper with upcoming work.

II. WISS, A SPEAKER IDENTIFICATION SYSTEM INTEGRATED TO MANYEARS

In clean (noiseless) conditions, speaker identification is usually done with the acoustics features of each phoneme, represented by the Mel-Frequency Cepstral Coefficients (MFCC). A Vector Quantization (VQ) model [10] or a Gaussian Mixture Model (GMM) [11] is generated for each set of features. The unique speaker's semantics (i.e., how phonemes are organized and not how they are pronounced) is more robust to noise but heavily relies on pre-trained language models [12]. With distant microphones, convolutive (room reverberation, channel distortion) and additive (incoherent background sound sources) noises need to be considered. Convolutive noise is usually removed with techniques derived from the Cepstral Mean Normalization (CMN) method [13]. On the other hand, additive noise can be lessened with spectrum subtraction [14]. Additive noise is usually pink as it is generated by fans or electronics. When both convolutive and additive noises are observed, it is possible to perform the CMN and spectrum subtraction techniques in cascade [15]. It is also common to update the speech model during the identification stage using Parallel Model Combination (PMC) so that the model matches the noisy environment conditions [16]. For speech recognition, a dynamic update of the Hidden Markov Models (HMM) is also proposed in order to match the environment conditions [17].

Training speaker models in noisy conditions is difficult as regions of low-energy are corrupted by additive noise, which distorts significantly the voice features. Training the models and identifying the speaker in the same noisy environment is desirable, but is not realistic as the noise is not stationnary

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT). F. Michaud holds the Canada Research Chair (CRC) in Mobile Robotics and Autonomous Intelligent Systems.

F. Grondin and F. Michaud are with the Department of Electrical Engineering and Computer Engineering, Université de Sherbrooke, 2500 boul. Université, Sherbrooke, Québec, CANADA, Francois.Grondin2@usherbrooke.ca Francois.Michaud@usherbrooke.ca

over long periods of time (e.g., a mobile robot interacts in a dynamic environment with moving speakers). However, it is possible to corrupt the models obtained (through the training phase) from clean features with the noise perceived in the operating environment to match the actual conditions.

Figure 1 illustrates the modules of our speaker identification system, which are described further in the following subsections. Training of speaker models is performed with a single close microphone in a quiet environment, with the same sampling rate and window size as of ManyEars, also used here as a pre-processing module to separate sound sources perceived in the environment. Features are then extracted for each speaker, and models are generated. The training signal is usually corrupted by convolutive noise whereas the identification signal is corrupted by both convolutive and additive noises. With ManyEars, localization is performed with a beamformer and each source is tracked with a particle filter. The source position is used to perform a Geometric Source Separation (GSS) and the separated signal is then enhanced with a post-filter. The separation stage in ManyEars improves the SNR of the source signal which is used to generate the features in the identification stage. The post-filtered signal is also used to generate a mask to identify noiseless dimensions in each feature. This signal is not used to generate the features since its time-varying gain makes it unsuitable for estimating the additive and convolutive noises. Features and masks are stored in a finite length buffer because a minimum number of elements is required to estimate noises. The models obtained during the training stage are then updated to match the noisy conditions in the identification stage and a score is computed. The identified speaker (id) and the level of confidence ($conf$) regarding this choice are thus obtained.

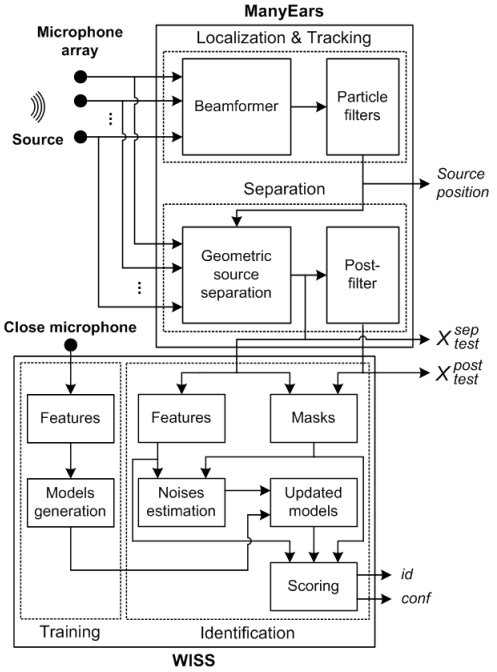

Fig. 1. Blocks diagram of WISS and ManyEars

A. ManyEars, A Sound Source Localization, Tracking and Separation System

In this section, we present a general overview of our implementation and configuration of ManyEars [1], [5], [2] for its uses with WISS. More specifically, ManyEars uses a cubic array of $M = 8$ microphones. The signal of each microphone is sampled at $F_s = 48000$ samples/sec and is windowed to generate frames of $N = 1024$ samples with a hop size of 512 samples.

Using the signals of each microphone, a Fast Fourier Transform (FFT) is performed on each frame and the power spectrum is computed. The expression $X^l_m[k]$ stands for the power spectrum of each frequency bin k, frame l and microphone m. A vector $\mathbf{X}^l[k]$ of M dimensions is defined in (1). The operator $(\cdot)^T$ stands for the transpose matrix.

$$\mathbf{X}^l[k] = \left[\begin{array}{ccc} X^l_0[k] & \ldots & X^l_{M-1}[k] \end{array} \right]^T \tag{1}$$

Localization is performed by a beamformer that scans the energy over a 2562-point spherical grid with a one meter radius. Equation (2) shows that the direction θ of a potential source matches the grid point with the maximum energy.

$$\theta = \arg\max_d \sum_{m_1=0}^{M-2} \sum_{m_2=m_1+1}^{M-1} R^l_{m_1,m_2}(delay_{m_1,m_2}(d)) \tag{2}$$

The expression $delay_{m_1,m_2}(d)$ stands for the time delay of arrival (TDOA) between microphones m_1 and m_2 for a sound source located at the point d on the grid. All delays are computed once and stored in memory. The cross-correlation $R^l_{m_1,m_2}(\tau)$ between the microphones signals m_1 and m_2 at frame l is given by the expression in (3), where the weighting factor $\zeta^l_m[k]$ reduces the contribution of noisy frequency bins affected by stationnary noise and reverberation. The operator $(\cdot)^*$ stands for the complex conjugate.

$$R^l_{m_1,m_2}(\tau) = \sum_{k=0}^{N-1} \frac{\zeta^l_{m_1}[k] X^l_{m_1}[k] \zeta^l_{m_2}[k] X^l_{m_2}[k]^*}{|(X_{mic})^l_{m_1}[k]||(X_{mic})^l_{m_2}[k]|} e^{\left(\frac{j2\pi k\tau}{N}\right)} \tag{3}$$

Once a potential source is found, the cross-correlation terms are set to zero over a delay range τ_r, as shown in (4). This removes the contribution of this source to the total energy of each point on the grid such that the next potential source can be found using (2). The localization step returns four potential sources as this process is set to repeat four times, for simultaneous tracking of sound sources.

$$\begin{array}{l} R^l_{m_1,m_2}(\tau) = 0, \\ \forall\ m_1, m_2, \tau : m_1 \neq m_2, |\tau - delay_{m_1,m_2}(\theta)| \leq \tau_r \end{array} \tag{4}$$

Each potential source either corresponds to a new source, a false detection or a source already tracked. Moreover, a sound source contains energy gaps in time because the speaker is not active during silence periods. For this reason, a particle filter made of 500 particles is assigned to each tracked source in order to model the source behavior as a slowly moving object that releases energy unevenly in time.

Separation of sound sources is done using information provided by the Localization & Tracking module. Sound source separation is based on the GSS technique, which minimizes cross-talk when multiple sources are active and maximizes directivity. A separation vector $(\mathbf{X_{test}^{sep}})^l[k]$ with Z dimensions (where Z is the number of sources currently tracked) is defined, as shown in (5).

$$(\mathbf{X_{test}^{sep}})^l[k] = \left[\begin{array}{ccc} (X_{test}^{sep})_0^l[k] & \dots & (X_{test}^{sep})_{Z-1}^l[k] \end{array} \right]^T \quad (5)$$

Equation (6) shows that the separated sources are obtained from the multiplication of the microphone matrix and the separation matrix, denoted by $\mathbf{W}^l[k]$.

$$(\mathbf{X_{test}^{sep}})^l[k] = \mathbf{W}^l[k]\mathbf{X}^l[k] \quad (6)$$

The separation matrix is updated recursively by discrete steps of μ according to two constraints, $\frac{\partial J_1(\mathbf{W}^l[k])}{\partial(\mathbf{W}^l)^*[k]}$ and $\frac{\partial J_2(\mathbf{W}^l[k])}{\partial(\mathbf{W}^l)^*[k]}$, a regularisation term λ and an energy normalization term $\alpha^l[k]$, as shown in (7). The first constraint minimizes the cross-correlation between the separated sources and the second one forces a unity gain for each separated source and no gain for interfering sources. These expressions are derived in [1].

$$\begin{aligned} \mathbf{W}^{(l+1)}[k] = \quad & (1-\lambda\mu)\mathbf{W}^l[k] - \\ & \mu\left[\alpha^l[k]\frac{\partial J_1(\mathbf{W}^l[k])}{\partial(\mathbf{W}^l)^*[k]} + \frac{\partial J_2(\mathbf{W}^l[k])}{\partial(\mathbf{W}^l)^*[k]}\right] \end{aligned} \quad (7)$$

When a new source z is tracked, a row is added to the separation matrix and its terms match the delay-and-sum beamformer. The separation is then improved with time as the separation matrix is updated recursively.

Finally, post-filtering is performed to enhance the separated signals. The post-filter module makes use of an attenuation factor $G_z^l[k]$ shown in (8), which takes into account stationnary noise and reverberation, as explained in [2].

$$(X_{test}^{post})_z^l[k] = G_z^l[k](X_{test}^{sep})_z^l[k] \quad (8)$$

B. WISS, A Speaker Identification System

WISS is divided in two stages: training and identification. The speaker models are generated in the training stage and they are compared to speech segments in the identification stage.

1) Training: Training is performed in a quiet environment with a close microphone. Features are extracted and then a model is generated.

a) Features: A pre-emphasis step is performed to increase the weight of high frequencies, which are less corrupted by pink noise. The emphasis factor $H_{emph}[k]$ is given by (9), where α_{emph} is the emphasis parameter.

$$H_{emph}[k] = 1 + (\alpha_{emph})^2 - 2\alpha_{emph}\cos(2\pi k/N) \quad (9)$$

The weighted spectrum $(\{X_{train}^{all}\}_{emph})_u^l[k]$ is given in (10), where $(X_{train}^{all})_u^l[k]$ stands for the power spectrum of the microphone signal at frame l for the speaker u.

$$(\{X_{train}^{all}\}_{emph})_u^l[k] = H_{emph}[k](X_{train}^{all})_u^l[k] \quad (10)$$

This spectrum is then multiplied by a filterbank of $F = 24$ filters, as shown in (11). The expression $b_f[k]$ stands for the gain of each filter f at each bin k. The range of frequencies goes from 0 Hz to 15500 Hz according to the Bark scale [18].

$$(\{X_{train}^{all}\}_{mel})_u^l[f] = \sum_{k=0}^{N-1} b_f[k](\{X_{train}^{all}\}_{emph})_u^l[k] \quad (11)$$

The logarithmic amplitude $(c_{train}^{all})_u^l[f]$ is computed to model the behavior of the human ear as shown in (12). The constant ε_{log} is added to avoid a math error if $(\{X_{train}^{all}\}_{mel})_u^l[f]$ is zero.

$$(c_{train}^{all})_u^l[f] = \ln\left[(\{X_{train}^{all}\}_{mel})_u^l[f] + \varepsilon_{log}\right] \quad (12)$$

Generated features include silence and speech frames. A voice activity detector similar to the one used in [1] is used and L_{train}^{speech} active frames are kept (named $(c_{train}^{speech})_u^l[f]$). Features are not converted to the cepstral domain as it is usually the case with MFCCs because masks can only be used in the spectral domain. Equation (13) shows how the CMN technique is used in the spectral domain to compensate for channel distortion to obtain the training features $(c_{train}^{cmn})_u^l[f]$.

$$(c_{train}^{cmn})_u^l[f] = (c_{train}^{speech})_u^l[f] - \frac{1}{L_{train}^{speech}} \sum_{l=0}^{L_{train}^{speech}-1} (c_{train}^{speech})_u^l[f] \quad (13)$$

b) Models generation: A codebook of V centroids is generated with the VQ technique to model the distribution of the features obtained in (13). The fast k-means algorithm is used to generate these centroids denoted by the expression $\kappa_u^v[f]$, where v stands for the centroid index [10].

2) Identification: Features, masks and noises estimation are first computed. The models are then updated to match these conditions and scoring is finally performed.

a) Features, Masks & Noises estimation: In a noisy environment, the speech signal is corrupted by both additive ($B_r[f]$) and convolutive ($H_r[f]$) noises. It is assumed that the speech signal and noises are statistically independent and homogeneous for each filter f. The variable r stands for the test segment index.

The separated and post-filtered spectra, $(X_{test}^{sep})_r^l[k]$ and $(X_{test}^{post})_r^l[k]$, are weighted with the same pre-emphasis factor proposed in (9) in order to obtain the new spectra $(\{X_{test}^{sep}\}_{emph})_r^l[k]$ and $(\{X_{test}^{post}\}_{emph})_r^l[k]$. These spectra are then multiplied by the filterbank used for training in (11), which leads to the expressions $(\{X_{test}^{sep}\}_{mel})_r^l[f]$ and $(\{X_{test}^{post}\}_{mel})_r^l[f]$.

To identify the noiseless dimensions of each feature, a hard instantaneous mask $(m_{inst})_r^l[f]$ is generated in (15) from the ratio $(ratio_{inst})_r^l[f]$ of the post-filtered and separated spectra obtained in (14).

$$(ratio_{inst})_r^l[f] = \frac{(\{X_{test}^{post}\}_{mel})_r^l[f]}{(\{X_{test}^{sep}\}_{mel})_r^l[f]} \quad (14)$$

$$(m_{inst})_r^l[f] = \begin{cases} 0 & (ratio_{inst})_r^l[f] < T_{inst} \\ 1 & (ratio_{inst})_r^l[f] \geq T_{inst} \end{cases} \quad (15)$$

In some cases, most dimensions of a feature are corrupted by noise (e.g., in silence periods), such that the full vector should not be used at all in the recognition process. A vertical mask $(m_{vert})_r^l$ is therefore needed to filter out these noisy features. Equation (16) shows how this hard mask is derived from the instantaneous mask.

$$(m_{vert})_r^l = \begin{cases} 0 & \left(\sum_{f=0}^{F-1} (m_{inst})_r^l[f]\right) < T_{vert} \\ 1 & \left(\sum_{f=0}^{F-1} (m_{inst})_r^l[f]\right) \geq T_{vert} \end{cases} \quad (16)$$

The logarithmic amplitude is computed to obtain $(c_{test}^{all})_r^l[f]$ from $(\{X_{test}^{sep}\}_{mel})_r^l[f]$ the same way this is done in (12).

Active frames (when $(m_{vert})_r^l$ is non-zero) are renamed $(c_{test}^{speech})_r^l[f]$, for a total of L_{test}^{speech} frames. For these frames, the convolutive noise usually dominates the additive noise. The corrupted speech with convolutive noise can therefore be estimated by averaging these frames, as shown in (17).

$$(\hat{X}_{conv})_r[f] = \frac{1}{L_{test}^{speech}} \sum_{l=0}^{L_{test}^{speech}-1} (c_{test}^{speech})_r^l[f] \quad (17)$$

On the other hand, additive noise is mainly observed in silence periods, more specifically in frequency bands where speech is missing. This noise $\hat{B}_r[f]$ can thus be estimated with a weighted average that relies on the instantaneous masks, as shown in (18).

$$\hat{B}_r[f] = \frac{\sum_{l=0}^{(L_{test}-1)} (c_{test}^{all})_r^l[f](1-(m_{inst})_r^l[f])}{\sum_{l=0}^{(L_{test}-1)} (1-(m_{inst})_r^l[f])} \quad (18)$$

Moreover, simulations showed that the expression in (19) is a fair estimate of the convolutive noise $\hat{H}_r[f]$, provided that the latter dominates the additive noise.

$$\hat{H}_r[f] = \ln\left\{\exp\left((\hat{X}_{conv})_r[f]\right) - \exp\left(\hat{B}_r[f]\right)\right\} \quad (19)$$

Convolutive noise might not outstand additive noise for all frequency bands, especially when pink noise is observed. A horizontal mask $(m_{hori})_r[f]$ is thus generated in (20) to address this problem.

$$(m_{hori})_r[f] = \begin{cases} 0 & \{(\hat{X}_{conv})_r[f] - \hat{B}_r[f]\} < 0 \\ 1 & \{(\hat{X}_{conv})_r[f] - \hat{B}_r[f]\} \geq 0 \end{cases} \quad (20)$$

The mean energy for each feature is removed in (21) in order to keep only the spectral shape, denoted by $(c_{test}^{ac})_r^l[f]$. The horizontal mask is used here to disregard noisy bands, which do not convey much speech information.

$$(c_{test}^{ac})_r^l[f] = (c_{test}^{all})_r^l[f] - \sum_{f=0}^{F-1} (c_{test}^{all})_r^l[f](m_{hori})_r^l[f] \quad (21)$$

Finally, global masks $(m_{all})_r^l[f]$ are obtained from the product of the instantaneous, vertical and horizontal masks, as shown in (22).

$$(m_{all})_r^l[f] = (m_{vert})_r^l(m_{hori})_r[f](m_{inst})_r^l[f] \quad (22)$$

b) Updated models: Models are then dynamically updated to match the environment conditions. Equation (23) shows that the noisy centroids $(\kappa_{noisy})_u^v[f]$ are obtained by adding additive and convolutive noises previously obtained in the linear-spectral and log-spectral domains respectively.

$$(\kappa_{noisy})_u^v[f] = \ln\left\{\exp\left(\kappa_u^v[f] + \hat{H}_r[f]\right) + \exp\left(\hat{B}_r[f]\right)\right\} \quad (23)$$

The normalized centroids $(\kappa_{ac})_u^v[f]$ are obtained in (24) by removing the weighted mean energy.

$$(\kappa_{ac})_u^v[f] = (\kappa_{noisy})_u^v[f] - \sum_{f=0}^{F-1} (\kappa_{noisy})_u^v[f](m_{hori})_r[f] \quad (24)$$

c) Scoring: The general expression for the Euclidean distance weighted with masks is defined in (25).

$$\text{dist}(\kappa, c, m) = \sqrt{\sum_{f=0}^{F-1} [m[f](\kappa[f] - c[f])^2]} \quad (25)$$

The score for each model u is obtained from the summation of the minimum Euclidean distance between all noisy centroids $(\kappa_{ac})_u^v$ and each test feature $(c_{test}^{ac})_r^l$, with the mask $(m_{all})_r^l$ used for weighting, as shown in (26).

$$score_u^r = \sum_{l=0}^{L_{test}-1} \min_{v=0,\ldots,(V-1)} \text{dist}((\kappa_{ac})_u^v, (c_{test}^{ac})_r^l, (m_{all})_r^l) \quad (26)$$

Equation (27) shows that the identified speaker corresponds to the model with the smallest difference between the model and the features.

$$(id_{exp})_r = \arg\min_u \{score_u^r\} \quad (27)$$

Finally, a confidence value $conf_r$ is evaluated using (28) for the identified speaker. This value makes use of a sigmoid function and depends on the difference $\Delta score$ between both smallest scores. The expressions α_s and β_s set respectively the decision level and the rate at which the confidence increases as this level is exceeded.

$$conf_r = (1 + \exp[-(\Delta score - \alpha_s)/\beta_s])^{-1} \quad (28)$$

III. RESULTS

Experiments are conducted to evaluate the recognition rates as a function of the SNR. For the trials, ManyEars is coded in C language while WISS is implemented using Matlab. They both run on a Intel Core i7 processor clocked at 2.93 GHz. In this setup, WISS uses only 10% of the CPU cycles.

Experiments are carried out in a room of 10 m × 10 m × 2.5 m with normal reverberation and some audible background noise generated by fans and other electronics. This noise is pink as most of its energy lies in the low frequencies. The eight microphones cubic array (0.32 m × 0.32 m × 0.32 m) is positioned in the middle of the room at 0.6 m above the floor. A loud speaker is used as a sound source located at ten different positions around the array and 0.6 m above the floor, as shown in Figure 2. The parameters of ManyEars and WISS set experimentally to optimize the performances are shown in Table I.

TABLE I
EXPERIMENTAL PARAMETERS

ManyEars		**WISS**						
λ	μ	α_{emph}	ε_{log}	V	T_{inst}	T_{vert}	α_s	β_s
0.5	0.002	0.95	10^{-10}	256	0.05	6	0.02	0.01

The speech segments of 11 female and 9 male speakers from the TSP Speech Database [19] are used for this experiment. Sequences of 60 seconds recorded in a quiet environment are used for training the model of each speaker. Six utterances of 10 seconds are then played sequentially for each speaker at five different gains (labeled from 1 to 5) to characterize the performances according to different SNRs. More than 16 hours of tests are thus recorded (10 positions × 20 speakers × 6 utterances × 5 gains × 10 secs) and analyzed later on.

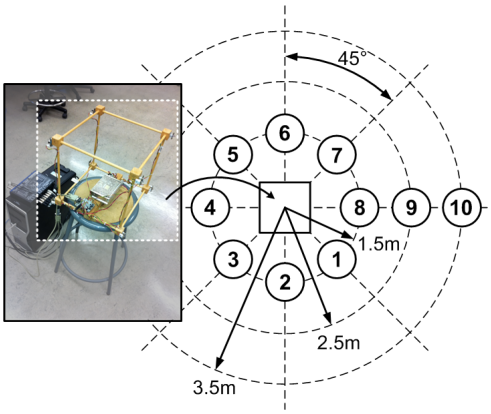

Fig. 2. Experimental setup

Table II presents the averaged SNR of the speech energy over the average of the noise energy during silence periods. As expected, similar SNR are observed for positions 1 to 8 for the same sound levels, when the distance is constant and the angle changes, and the SNR decreases as the speaker moves to positions 9 and 10 (as the distance increases).

The weighting rate of good identifications is defined as the ratio of the sum of the confidence values that correspond to a good speaker identifications over the sum of all confidence values, as shown in (29). The expression $(id_{theo})_r$ stands for the real speaker identity. On the other hand, the unweighted rate of good identifications is the sum of correct identifications over the sum of all identifications.

TABLE II
SIGNAL-TO-NOISE RATIOS (IN DB) AS A FUNCTION OF SOURCE POSITIONS AND SOUND LEVELS

Position	Level 1	Level 2	Level 3	Level 4	Level 5
1	16.92	12.20	7.94	4.41	2.07
2	17.33	12.57	8.23	4.61	2.25
3	16.56	11.86	7.72	4.27	1.82
4	15.56	10.91	6.68	3.58	1.50
5	15.31	10.77	6.60	3.52	1.61
6	15.75	11.21	6.99	3.66	1.63
7	17.29	12.54	8.24	4.60	2.24
8	16.59	11.84	7.75	4.18	1.69
9	12.48	8.13	4.50	2.18	0.83
10	10.22	6.14	3.11	1.30	0.42

$$rate^{good}_{weight} = \frac{\sum_{\forall r:(id_{exp})_r=(id_{theo})_r} conf_r}{\sum_{\forall r} conf_r} \tag{29}$$

Figure 3 shows the weighted rates of good identifications without masks (e.g. $(m_{all})^l_r[f] = 1 \; \forall \; r,l,f$) and without PMC. In the latter case, the features $(c^{ac}_{test})^l_r[f]$ are normalized with CMN as shown in (13), the mean energy of each of the features $(c^{speech}_{train})^l_u[f]$ is removed as in (24) prior to training the models, and models are not updated, which leads to $(\kappa_{ac})^v_u[f] = (\kappa_{noisy})^v_u[f] \; \forall \; u,v,f$. The unweighted rates are also presented for level 5 and show that this method improves the identification rates in addition to sharpen the weighted rates. Performances without masks but with PMC are shown in Figure 4 and rates with masks and PMC are shown in Figure 5. When masks and PMC are used, best performances are achieved at a distance of 1.5 m at level 1. Performances are similar for positions 1 to 8, no matter what is the source angle (the small difference is due to different microphone gains). In this configuration, the SNR is 16dB on average and the weighted rate of good identifications is 96% on average. At level 5, for the same positions, the error rate drops to 84%, as the SNR drops to 2dB on average. On the other hand, worst performances are observed at a distance of 3.5 m, which corresponds to position 10. For this scenario, the SNR drops to 0.42 dB and thus the rate decreases to 70%. In a few cases, the performances at position 10 outperform those at position 9 since the low SNR introduces some randomness in the results. Figure 3, Figure 4 and Figure 5 suggest that using masks and PMC increase the rate of good identifications up by 52% in some cases. However, when the SNR is high (e.g., at level 1), PMC improves the performances by 10% whereas masks do not increase the recognition rates, as expected.

IV. CONCLUSION

This paper presents WISS, a new speaker identification system designed to be coupled to ManyEars. The proposed system is well prepared for speaker identification on mobile robots because training is performed once and models are updated online to the noise level perceived in the operating environment. Moreover, the use of masks and PMC in WISS improves the performances when dealing with pink noise.

Based on the good performances observed in our trials, the next step consists in designing a C-library to run WISS in

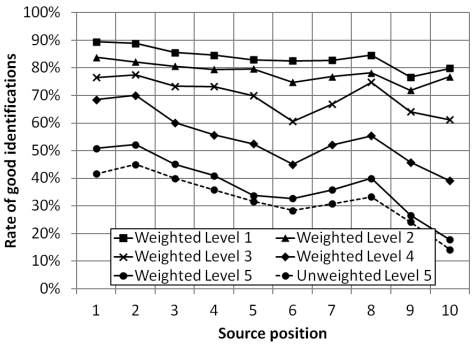

Fig. 3. Rate of good identifications (No mask, No PMC)

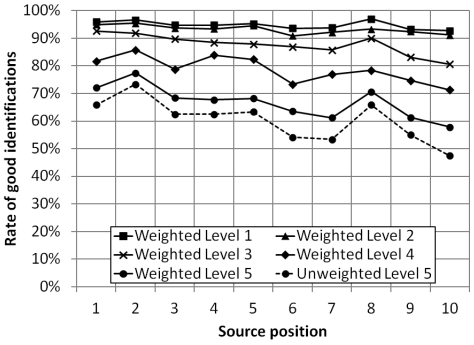

Fig. 4. Rate of good identifications (No mask, With PMC)

real-time on a mobile robot. Experiments will be conducted to measure performances with multiple simultaneous sound sources. Up to now, testing has been verified with utterances of 10 seconds. Dealing with shorter utterances (2 or 3 seconds) will be investigated as they will occur in real life dialog. Once these validations are completed, we intend to combine WISS with more sophisticated processes, such as speaker verification (with the speaker indicating its identity before performing the verification), speaker tracking (using localization and tracking data from ManyEars) and multi-modal people identification (using visual and audio features). The confidence value introduced will be used for fusion with face recognition engines and other bio-identification systems.

REFERENCES

[1] J.-M. Valin, S. Yamamoto, J. Rouat, F. Michaud, K. Nakadai, and H. G. Okuno, "Robust recognition of simultaneous speech by a mobile robot," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 742 – 752, 2007.

[2] J.-M. Valin, J. Rouat, and F. Michaud, "Enhanced robot audition based on microphone array source separation with post-filter," vol. 3, 2004, pp. 2123 – 2128.

[3] S. Yamamoto, K. Nakadai, J.-M. Valin, J. Rouat, F. Michaud, K. Komatani, T. Ogata, and H. Okuno, "Making a robot recognize three simultaneous sentences in real-time," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2005, pp. 4040 – 4045.

[4] S. Yamamoto, K. Nakadai, M. Nakano, H. Tsujino, J.-M. Valin, K. Komatani, T. Ogata, and H. G. Okuno, "Real-time robot audition system that recognizes simultaneous speech in the real world," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 5333 – 5338.

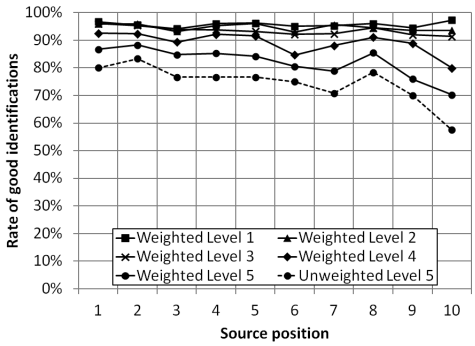

Fig. 5. Rate of good identifications (With masks, With PMC)

[5] J.-M. Valin, F. Michaud, and J. Rouat, "Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering," *Robotics and Autonomous Systems*, vol. 55, no. 3, pp. 216 – 228, 2007.

[6] C. Zhang, Z. Zhang, and D. Florencio, "Maximum likelihood sound source localization for multiple directional microphones," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, 2007, pp. I125 – I128.

[7] M. Ji, S. Kim, H. Kim, and H.-S. Yoon, "Text-independent speaker identification using soft channel selection in home robot environments," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 1, pp. 140 – 144, 2008.

[8] K.-D. Ban, K.-C. Kwak, H.-S. Yoon, and Y.-K. Chung, "Fusion technique for user identification using camera and microphone in the intelligent service robots," *Proceedings of the International Symposium on Consumer Electronics*, 2007.

[9] F. Michaud, C. Cote, D. Letourneau, Y. Brosseau, J.-M. Valin, E. Beaudry, C. Raievsky, A. Ponchon, P. Moisan, P. Lepage, Y. Morin, F. Gagnon, P. Giguere, M.-A. Roux, S. Caron, P. Frenette, and F. Kabanza, "Spartacus attending the 2005 AAAI conference," *Autonomous Robots*, vol. 22, no. 4, pp. 369 – 383, 2007.

[10] C. Elkan, "Using the triangle inequality to accelerate k-means," *Proceedings of the Twentieth International Conference on Machine Learning*, vol. 1, pp. 147 – 153, 2003.

[11] D. Reynolds and R. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72 – 83, 1995.

[12] D. Reynolds, W. Andrews, J. Campbell, J. Navratil, B. Peskin, A. Adami, Q. Jin, D. Klusacek, J. Abramson, R. Mihaescu, J. Godfrey, D. Jones, and B. Xiang, "The SuperSID project: Exploiting high-level information for high-accuracy speaker recognition," in *Acoustics, Speech, and Signal Processing*, vol. 4, 2003, pp. 784 – 787.

[13] C.-W. Hsu and L.-S. Lee, "Higher order cepstral moment normalization for improved robust speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 2, pp. 205 – 220, 2009.

[14] W. Mwema and E. Mwangi, "A spectral subtraction method for noise reduction in speech signals," in *Proceedings of the Fourth IEEE Africon Conference*, vol. 1, 1996, pp. 382 – 385.

[15] W. Hong and P. Jin'gui, "Modified MFCCs for robust speaker recognition," in *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems*, vol. 1, 2010, pp. 276 – 279.

[16] L. P. Wong and M. J. Russell, "Speaker verification under additive noise conditions with non-stationary SNR using PMC," in *The Speaker Recognition Workshop*, 2001.

[17] Y. Gong, "A method of joint compensation of additive and convolutive distortions for speaker-independent speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 975 – 983, 2005.

[18] T. Gulzow, A. Engelsberg, and U. Heute, "Comparison of a discrete wavelet transformation and a nonuniform polyphase filterbank applied to spectral-subtraction speech enhancement," *Signal Processing*, vol. 64, no. 1, pp. 5 – 19, 1998.

[19] P. Kabal. (2002) Tsp speech database. [Online]. Available: http://www-mmsp.ece.mcgill.ca/Documents/Data/index.html