Proceedings of the 2007 IEEE/RSJ International
Conference on Intelligent Robots and Systems
San Diego, CA, USA, Oct 29 - Nov 2, 2007

HUMANOID ROBOT NOISE SUPPRESSION BY PARTICLE FILTERS FOR IMPROVED AUTOMATIC SPEECH RECOGNITION ACCURACY

Florian Kraft and Matthias Wölfel

Institut für Theoretische Informatik, Universität Karlsruhe (TH)
Am Fasanengarten 5, 76131 Karlsruhe, Germany
fkraft@ira.uka.de, wolfel@ira.uka.de

ABSTRACT

Automatic speech recognition on a humanoid robot is exposed to numerous known noises produced by the robot's own motion system and background noises such as fans. Those noises interfere with target speech by an unknown transfer function at high distortion levels, since some noise sources might be closer to the robot's microphones than the target speech sources. In this paper we show how to remedy those distortions by a speech feature enhancement technique based on the recently proposed particle filters. A significant increase of recognition accuracy could be reached at different distances for both engine and background noises.

Index Terms— speech feature enhancement, particle filter, humanoid robots, automatic speech recognition

Fig. 1. Armar III in the kitchen lab

1. INTRODUCTION

We address noise robustness for far distant speech recognition in the context of a humanoid robot's own noise production. The research project SFB588 ”Humanoid Robots - Learning and Cooperating Multimodal Robots” [1] aims at the development of a household robot (Fig.1) which is intended to support humans at home. One part of the project is natural human robot interaction with verbal communication. The current working environment of the humanoid robot *Armar* [2, 3] is a kitchen. Besides concurrent kitchen sound events and human speech Armar's auditory system has to deal with its own noise production corrupting the human speech signal. Noises from the robot could be generated by fans or engines and its synthetic voice. This work concentrates on the compensation of distortions due to engine and background noises rather than synthetic speech, in which case the source signal is known. We therefore use an approach which can model noises descended from incompletely known dynamic noise sources, namely sequential Monte Carlo methods. They have been recently introduced as a method for speech feature enhancement for speech recognition [4, 5], which can deal with non-stationary noises. Particle filters, as a special case of sequential Monte Carlo methods, are getting more popular for the task of dynamic noise compensation: the automatic speech recognition performance for several noise types like machine-guns, music, traffic or garbage collection vehicles could be improved by applying particle filters [4, 5, 6]. Faubel and Wölfel have demonstrated an additional gain in accuracy by coupling the particle filter with the automatic speech recognition [7]. In this paper we investigate particle filter based speech feature enhancement technique using sampling importance resampling (SIR) in human robot interaction at different distances between the two. Since the robot's own noises usually have less distance to the integrated far distance microphones compared to a human speaker, a natural human robot communication suffers from high distortion levels. Particle filters seem to be a suited solution for this problem, since the dynamic noises of the motion system interfere additively and the engine timings are known, which allows to switch between different noise models for particle filter initialization.

2. PARTICLE FILTER BASED SPEECH FEATURE ENHANCEMENT

Speech feature enhancement can be formulated as to track the clean speech spectrum $\mathbf{x}_k$ with the observation history $\mathbf{y}_{1:k} = \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$ using the probabilistic relationship $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. To find the optimal solution (with respect to

1-4244-0912-8/07/$25.00 ©2007 IEEE.

the mean square error criterion) to yield the minimum mean square error estimate consists of finding the conditional mean $\mathrm{E}[\mathbf{x}_{1:k}|\mathbf{y}_{1:k}]$ [8]. Assuming that $(\mathbf{x}_k)_{k\in\mathbb{N}}$ is a Markov process and that the current observation is only dependent on the current state. A recently proposed solution in speech processing to this problem, without constrains such as linearity or gaussianity of the system is the particle filter [9]. An illustration of the particle filter with importance sampling and resampling is given in Fig. 2.

2.1. The Particle Filter

The used particle filter follows the approach by Singh et al. [5] which aims, in contrast to common approaches, to track the noise spectra corrupted by speech. Clean speech spectra is later on derived by an interference step. This particle filter for speech feature enhancement can be outlined as follows:

1. *Sampling*
   At time zero ($k = 0$) noise hypotheses (particles) $\mathbf{n}_0^{(j)}$ ($j = 1, ..., N$) are drawn from the *prior noise density* $p(\mathbf{n}_0)$. For $k > 0$, $\mathbf{n}_k^{(j)}$ is sampled from the *noise transition probability* $p(\mathbf{n}_k|\mathbf{n}_{k-1}^{(j)})$ for $j = 1, ..., N$.

2. *Calculating the normalized importance weights*
   The importance weight (likelihood) of each noise hypothesis $\mathbf{n}_k^{(j)}$ is evaluated by $p(\mathbf{y}_k|\mathbf{n}_k^{(j)})$. The normalized importance weights are calculated as

$$\tilde{w}_k^{(j)} = \frac{p(\mathbf{y}_k|\mathbf{n}_k^{(j)})}{\sum_{m=1}^{N} p(\mathbf{y}_k|\mathbf{n}_k^{(m)})}$$

3. *Inferring clean speech*
   Clean speech is inferred as explained in 2.4 for two different strategies by using the discrete Monte Carlo representation of the continuous filtering density

$$p(\mathbf{n}_k|\mathbf{y}_{1:k}) = \sum_{j=1}^{N} \tilde{w}_k^{(j)} \delta_{\mathbf{n}_k^{(j)}}(\mathbf{n}_k) \qquad (1)$$

   The term $\delta_{\mathbf{n}_k^{(j)}}$ denotes a translated Dirac delta function.

4. *Importance resampling*
   The normalized weights are used to resample among the noise hypotheses $\mathbf{n}_k^{(j)}$ ($j = 1, ..., N$). This can be regarded as a pruning step where likely hypotheses are multiplied, unlikely ones are removed from the population.

Those steps are repeated with $k \mapsto (k + 1)$ until all timeframes are processed.

2.2. Modeling noise and its evolution

To initialize the noise hypothesis one can sample from the prior noise density $p(\mathbf{n}_0)$. The prior noise density can be modeled as a Gaussian mixture model and trained on known or expected noise types a priori or on silence regions of the current speech observation.

In [6] the evolution of (log Mel) noise spectra is modeled as a 1st-order autoregressive process

$$\mathbf{n}_{k+1} = A \cdot \mathbf{n}_k + \varepsilon_k$$

where A is the transition matrix that is learned for a specific type of noise and $\mathbf{n}_k$ denotes the noise spectrum at time k. The ε_k terms are considered to be i.i.d. zero mean Gaussian, i.e. $\varepsilon_k \sim \mathcal{N}(0, \mathbf{\Sigma}_{\mathrm{noise}})$, where the covariance matrix $\mathbf{\Sigma}_{\mathrm{noise}}$ is assumed to be diagonal. Using this model the noise transition probability $p(\mathbf{n}_{k+1}|\mathbf{n}_k)$ can be written as

$$p(\mathbf{n}_{k+1}|\mathbf{n}_k) = \mathcal{N}(\mathbf{n}_{k+1}; A \cdot \mathbf{n}_k, \mathbf{\Sigma}_{\mathrm{noise}}) \qquad (2)$$

Since [6] et al. found that higher model orders than 1st-order are not leading to a significant improved performance, we use 1st-order predictors.

2.3. Modeling clean speech and evaluate likelihoods

Modeling clean speech (log Mel) spectra $\mathbf{x}_k$ as a Gaussian mixture distribution $\mathbf{p}(\mathbf{x})$ learned for all of speech, the relationship

$$\mathbf{x}_k \approx \log(e^{\mathbf{y}_k} - e^{\mathbf{n}_k}) = \mathbf{y}_k + \log(1 - e^{\mathbf{n}_k - \mathbf{y}_k}) \qquad (3)$$

between corrupted speech spectra $\mathbf{y}_k$, $\mathbf{n}_k$ and $\mathbf{x}_k$ (all in the log Mel domain), the likelihood $l(\mathbf{n}_k^{(j)}; \mathbf{y}_k) = p(\mathbf{y}_k|\mathbf{n}_k^{(j)})$ of a noise hypothesis $\mathbf{n}_k^{(j)}$ can be evaluated as

$$p(\mathbf{y}_k|\mathbf{n}_k^{(j)}) = \frac{p_x(\mathbf{y}_k + \log(1 - e^{\mathbf{n}_k^{(j)} - \mathbf{y}_k}))}{\prod_{i=1}^{d} \left|1 - e^{n_{k,i}^{(j)} - \mathbf{y}_{k,i}}\right|}, \qquad (4)$$

where $\mathbf{d}$ represents the dimension of the noise spectral vector. Note that this equation can only be evaluated if no spectral bin $\mathbf{n}_k^{(j)}$ exceeds $\mathbf{y}_k$, otherwise $p(\mathbf{y}_k|\mathbf{n}_k) = 0$.

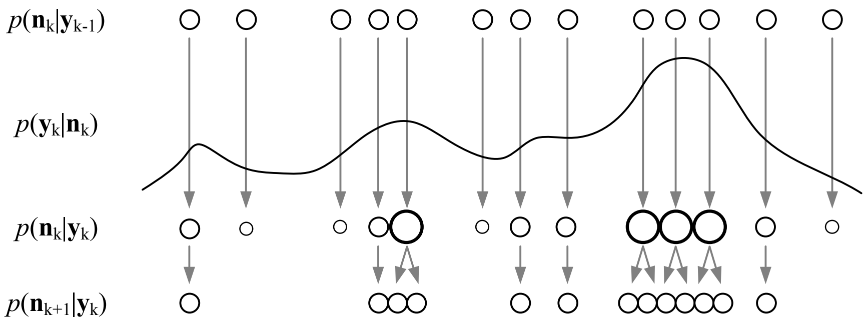
2.4. Inferring Clean Speech

The solution to the particle filter problem, to get enhanced speech features, consists in computing the conditional mean

$$\mathrm{E}[\mathbf{x}_k|\mathbf{y}_{1:k}] = \int \mathbf{x}_k \cdot p(\mathbf{x}_k|\mathbf{y}_{1:k}) dx \qquad (5)$$

where the noise $\mathbf{n}_k$ is introduced as a hidden variable

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \int p(\mathbf{x}_k, \mathbf{n}_k|\mathbf{y}_{1:k}) d\mathbf{n}_k$$



$p(\mathbf{n}_\mathrm{k}|\mathbf{y}_\mathrm{k\text{-}1})$
$p(\mathbf{y}_\mathrm{k}|\mathbf{n}_\mathrm{k})$
$p(\mathbf{n}_\mathrm{k}|\mathbf{y}_\mathrm{k})$
$p(\mathbf{n}_\mathrm{k+1}|\mathbf{y}_\mathrm{k})$

Fig. 2. Illustration of a particle filter with importance sampling and resampling.

Further, using $p(\mathbf{x}_k, \mathbf{n}_k|\mathbf{y}_{1:k}) = p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{n}_k) \cdot p(\mathbf{n}_k|\mathbf{y}_{1:k})$ and changing the order of integration we obtain

$$\mathrm{E}[\mathbf{x}_k|\mathbf{y}_{1:k}] = \int \underbrace{\int \mathbf{x}_k \cdot p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{n}_k) d\mathbf{x}_k}_{=\mathbf{h}_k(\mathbf{n}_k)} p(\mathbf{n}_k|\mathbf{y}_{1:k}) d\mathbf{n} \tag{6}$$

Since this is equivalent to $E_{p(\mathbf{n}_k|\mathbf{y}_{1:k})}[h_k(\mathbf{n}_k)|\mathbf{y}_{1:k}]$, the weighted empirical density (1) can be used to approximate (6) by Monte Carlo integration [10]:

$$\mathrm{E}[\mathbf{x}_k|\mathbf{y}_{1:k}] \quad \approx \quad \sum_{j=1}^{N} \tilde{w}_k^{(j)} h_k(\mathbf{n}_k^{(j)}) \tag{7}$$

To finally be able to calculate the conditional mean we have to evaluate for

$$\mathbf{h}_k(\mathbf{n}_k) = \int \mathbf{x}_k \cdot p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{n}_k) d\mathbf{x}_k \tag{8}$$

with

$$\mathbf{x}_k = \mathbf{y}_k + \log(1 - e^{\mathbf{n}_k - \mathbf{y}_k})$$

Two approaches to solve for $\mathbf{h}$ have been proposed in the literature which we will summarize in the next sections:

The vector Taylor series approximation

Following Raj *et al.* [6] we can approximate $log(1 + e^{\mathbf{n}_k - \mathbf{x}_k})$ by a 0th order Taylor series expansion around every Gaussian mean $\mu_g \; \forall \; g$.

$$\begin{aligned} \mathbf{h}_k(\mathbf{n}_k) &= \int \mathbf{x}_k \sum_{g=1}^{G} p(\mathbf{x}_k|g, \mathbf{y}_k, \mathbf{n}_k) p(g|\mathbf{y}_k, \mathbf{n}_k) \; d\mathbf{x}_k \\ &= \sum_{g=1}^{G} p(g|\mathbf{y}_k, \mathbf{n}_k) \int \mathbf{x}_k p(\mathbf{x}_k|g, \mathbf{y}_k, \mathbf{n}_k) \; d\mathbf{x}_k \end{aligned}$$

where, under the assumption that clean speech and noise is stochastically independent, we can write

$$p(g|\mathbf{y}_k, \mathbf{n}_k) = \frac{p(g, \mathbf{y}_k|\mathbf{n}_k)}{p(\mathbf{y}_k|\mathbf{n}_k)} = \frac{c_k \cdot p(\mathbf{y}_k|\mathbf{n}_k, g)}{p(\mathbf{y}_k|\mathbf{n}_k)}$$

with $p(g|\mathbf{y}_k) = p(g) = c_g$.

The noise shifts the means of the Gaussians in the logarithmic domain to

$$\mu_k' = \mu_k + \underbrace{log(1 + e^{\mathbf{n}_k - \mu_k})}_{=\Delta_{\mu_k, \mathbf{n}_k}}. \tag{9}$$

which can also be considered as a shift of the corrupted spectrum in the opposite direction to obtain the clean spectrum:

$$\mathbf{x}_k = \mathbf{y}_k - \Delta_{\mu_g, \mathbf{n}_k}.$$

With

$$p(\mathbf{x}_k|g, \mathbf{y}_k, \mathbf{n}_k) = \delta_{\mathbf{y}_k - \Delta_{\mu_g, \mathbf{n}_k}}(\mathbf{x}_k)$$

we can finally solve for

$$\begin{aligned} \mathbf{h}_k^{\mathrm{vts}}(\mathbf{n}_k) &= \sum_{g=1}^{G} p(g|\mathbf{y}_k, \mathbf{n}_k) \int \mathbf{x} \delta_{\mathbf{y}_k - \Delta_{\mu_g, \mathbf{n}_k}}(\mathbf{x}_k) \; d\mathbf{x}_k \\ &= \sum_{g=1}^{G} p(k|\mathbf{y}_k, \mathbf{n}_k) \left(\mathbf{y}_k - \Delta_{\mu_g, \mathbf{n}_k}\right) \\ &= \mathbf{y}_k - \sum_{g=1}^{G} p(g|\mathbf{y}_k, \mathbf{n}_k) \Delta_{\mu_g, \mathbf{n}_k} \end{aligned} \tag{10}$$

The direct approach

Faubel and Wölfel proposed to directly use the relationship between $\mathbf{x}_k$, $\mathbf{n}_k$ and $\mathbf{y}_k$ [11] from (3). This makes the probability density $p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{n}_k)$ deterministic, since $\mathbf{x}_k$ is completely determined if $\mathbf{y}_k$ and $\mathbf{n}_k$ are given:

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{n}_k) = \delta_{\mathbf{y}_k + log(1 - e^{\mathbf{n}_k - \mathbf{y}_k})}(\mathbf{x}_k).$$

Substitution in (8) yields

$$\begin{aligned} \mathbf{h}_k^{\mathrm{direct}}(\mathbf{n}_k) &= \int \mathbf{x}_k \cdot \delta_{\mathbf{y}_k + log(1 - e^{\mathbf{n}_k - \mathbf{y}_k})}(\mathbf{x}_k) d\mathbf{x}_k \\ &= \mathbf{y}_k + log(1 - e^{\mathbf{n}_k - \mathbf{y}_k}). \end{aligned} \tag{11}$$

3. DATA DESCRIPTION AND BASELINE SYSTEM

The following subsection describes the collection of three data sets: one set containing far distant speech without noise constituting our test set, and two noise sets (engine and background noise) which we both split into training data used for the particle filter models and testing data used for distortion of the speech test set. Thereafter we describe the front-end with feature spaces used, the acoustic and the language model. Finally the particle filter model training is explained.

3.1. Recording Setup and Data Sets

We recorded 100 continously spoken sentences from each of three non-native speakers using a close-talking microphone at 48 kHz. The utterances contain typical instructions and questions for a household robot in the kitchen domain (e.g. ”Make a cup of tea with milk and sugar.” or ”What can I cook with tomatoes and onions?”). For a fair comparison of automatic speech recognition (ASR) performance we played back all recorded utterances in a real kitchen environment using the Fostex Personal Monitor 6301B with three distances (60 cm, 120 cm, 180 cm) between the robot microphones (Sony ECM C115) which are located at the left and right side of the head and the loudspeaker. Besides these distant speech recordings, we captured four types of engine noises caused by the robot's motion system (movement of head, hands, arms and platform) and background noise (especially by the robot's fan) during a real robot dialog interaction, again using the robot's microphones in the same environment. Since the noise type *hand movement* turned out to be somewhat silent, we ignored this subset. The resulting engine noise type set contained 139 instances and was split into a training (104 recordings) and a test set (35 recordings). The background noise consisted mainly of a fan on the robot's platform which is far more static than the engine noises. Therefore 35 seconds for particle filter prior model training and 10 seconds for test data distortion are reasonable amounts. All data have been recorded at 48 kHz and later downsampled to 16 kHz.

3.2. Acoustic Pre-Processing

To extract robust speech features, every 10 ms, we have replaced the traditional power spectrum by a warped *minimum variance distortionless response* (MVDR) spectral envelope [12] of model order 30. In contrast to traditional approaches no filterbank was used as the warped MVDR envelope already provides those properties, namely smoothing and frequency warping. The 129 spectral features have been truncated to 20 cepstral coefficients after cosinus transformation. Those features have been transfered back to logarithmic mel spectral domain by an inverted cosine transformation to span the 20 dimensional feature space in which the particle filter is applied. After feature enhancement the 20 dimensional vector is again converted into the cepstral domain. After mean and variance normalization the cepstral features where stacked (7 adjacent left and right frames) and truncated to the final feature dimension 42 by multiplying with the optimal feature space matrix (the linear discriminant analysis matrix multiplied with the global semi-tight covariance transformation matrix [13]).

3.3. Acoustic and Language Model

The acoustic model (AM) contains 19,460 distributions over 4,127 models, with a maximum of 64 Gaussians per model trained on close talking meeting and lecture data. The dictionary contains 62,421 pronunciation variants over a vocabulary of 51,733 words. Further we used a 4-gram in-domain language model (LM).

3.4. Particle Filter Model Training

To initialize the noise model for the particle filter sampling step a GMM was used as prior noise model for each particle filter. The likelihood evaluation of a noise hypothesis on the corrupted speech log Mel spectra enters the clean speech GMM into equation (4). This model was trained on the same data (meeting and lecture data) as used to train the AM, but in this case in the spectral log Mel feature space in which the particle filter operates. The noise and speech GMMs have been trained by split-and-merge training.

To model time evolution of a particular noise type, a linear prediction transition matrix was estimated for each noise type. Since the robot is likely to move different engines at once, we aim at modeling several different engine noises at a time. To allow this in the evolution model, waveforms from the engine noise training set were mixed such that all three engine noise types were present at a time. The mixed waveform was then used in the 1st-order autoregressive estimation process.

4. EXPERIMENTS

Before analyzing the ASR performances for improved word error rates by feature enhancement using particle filters, we give a short description of the three test sets labelled as *clean, background noise* and *engine noise* in Table 1.

4.1. Creation of Test Sets

As shown in Table 1 we experimented with three test sets. All sets consist of the same far distant recordings: the *clean* case represents the far distant recordings played back from the loudspeaker which were captured by the robot's microphones. In the other two cases *background noise* or *engine noise* were added to the *clean*-case. Note that although we additively mixed the noise to the far distant speech, both speech and noise were recorded by the same microphones and environment.

test data	PF type	PF noise trained on	word error rate in %					
distance			60 cm		120 cm		180 cm	
pass			1st	2nd	1st	2nd	1st	2nd
clean	no	—	9.8	4.7	20.6	7.7	24.9	9.3
	vts	background noise	9.5	4.3	18.0	*8.0*	18.7	6.6
	direct	background noise	8.1	4.1	16.2	*7.8*	18.9	7.9
background noise	no	—	9.9	4.7	21.2	8.2	28.2	10.4
	vts	background noise	9.3	*5.2*	20.0	7.6	23.4	8.6
	direct	background noise	9.0	4.1	16.8	6.8	21.9	8.6
engine noise	no	—	22.6	10.0	57.1	32.3	78.0	52.4
	vts	engine noise	13.6	8.8	46.9	26.9	67.5	50.0
	direct	engine noise	16.8	8.0	44.6	25.1	67.1	48.5

Table 1. ASR performances in WER for the particle filter types *no, direct approach, vts approach* using different training and test data combinations at three robot-speaker distances

The mixing was done such that we determined the signal-to noise ratio (SNR) between the speech signal and the noise signal in the time domain, both captured by the robot's microphones. The distance dependent SNRs were used accordingly as distortion levels for speech signal corruption by additive mixing. For the *engine noises* the SNRs were about 15 dB at 60 cm, 9dB at 120 cm and 6 dB at 180 cm. The *background noises* for additive mixing with the far distant speech test data were taken from the same recordings as the engine noise recordings used for testing. Therefore the SNRs for the background noise were higher according to the background proportion of the engine recordings.

For both test data cases in which speech was distorted by background noise or by engine noise, all utterances were mixed with the noises over the whole time span. Even though this affects the WERs in an unrealistic way for the engine noises, because the robot won't move all its' engines all the time during a speech dialog, we can get a sense of how the noises affect those parts of speech utterances when they are present.

4.2. Results and Discussion

For each test set we show a reference system without a particle filter (PF type: no) which we compare to particle filter approaches with *vector Taylor series* (PF type: vts) approximation and with the *direct* deterministic solution (PF type: direct). The results are given in *word error rates* (WER) in Table 1 for first and second pass recognition runs, where the second pass is adapted by *maximum likelihood linear regression* (MLLR) [14] and constrained MLLR on the appropriate hypotheses of the first passes.

Adding background noise to the clean far distant speech baseline affects the WERs less negative than adding with engine noises, which are quite dynamic and have a higher power level. Comparing the presence of engine noise with the silence case at 180 cm distance, without speech feature enhancement, we observe a performance drop on the 2nd pass from 52.4% to 9.3% WER. Compared to this, a difference of 10.4% to 9.3% in the same condition with background noise instead of engine noise is less drastic.

Besides a common degradation of performance in WER with increasing microphone-speaker distance, we observe a significant improvement trend (except of three italic marked cases) for each distance and each test set for the 2nd passes of both particle filter types *direct* and *vts* when comparing with the reference systems. The *direct* approach shows one outlier with a not significant absolute WER difference of 0.1% to the reference system. For the *direct*-type this negative difference appears only on the clean[1] test data, where an improvement by the particle filter was not necessarily expected. Nevertheless in all other clean test data cases both particle filter types reached improvements over the baseline system, which consists of a plain recording without further noise mixing. The other two remaining degradation cases mentioned before happened for the *vts*-type with an absolute WER degradation of 0.3% for clean and 0.5% for background noise added test data at 60 cm.

In all other cases both particle filter types could achieve benefits up to 7.2% absolute WER difference. Note that the reported improvements are for the adapted 2nd pass for a typical robot-speaker dialog distance (120cm) with severe engine noise distortion for disjoint training and testing sets.

Finally we compare the different approaches to infer the clean speech, namely the *vts* and *direct* approach. In seven of nine first pass cases the particle filter type *direct* performed better than the *vector Taylor series* approach. This trend was confirmed by the adapted recognition runs: besides one outlier out of nine cases the *direct* approach achieved better re-

[1] The particle filter training for the clean test data case was performed on the fly by using background noise from non-speech regions of the clean test data recording. In both other test data cases we split particle filter training data and noise data, used for mixing, in disjoint sets. This offline-training strategy is motivated by the timing knowledge of the robot's engines.

sults. This reveals the superiority of the deterministic solution.

5. CONCLUSIONS

We successfully applied particle filtering for feature enhancement in a humanoid robot's far distant speech recognition system in a kitchen environment. A significant performance improvement in terms of WER could be shown for compensation of human speech distortion by background and engine noises at different speaker distances for both particle filter types (direct and vts). For a typical robot speaker distance we could show an absolute reduction of 7.2% WER for a simultaneous distortion by three engine-types at about 9 dB SNR. In contrary to our expections even undistorted speech could be recognized at about same and even better WERs when applying particle filters trained on speech pauses. The comparison of the vts-method with the direct-method revealed a superiority of the deterministic solution.

In the future we plan to embed engine specific particle filters and their mixtures which can be switched according to the known timing of various robot engines. Furthermore, the integration of knowledge from a kitchen sound event classification system [15] should improve speech recognition distorted by noises not tied to the robot's own noise production.

6. ACKNOWLEDGEMENTS

This research is partially funded by the German Research Foundation (DFG) under Sonderforschungsbereich SFB 588: "Humanoid Robots - Learning and Cooperating Multimodal Robots".

7. REFERENCES

[1] "The SFB 588 website," www.sfb588.uni-karlsruhe.de.

[2] T. Asfour, K. Berns, and R. Dillmann, "The humanoid robot armar: Design and control," *IEEE-RAS International Conference on Humanoid Robots*, 2000.

[3] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "Armar-III: An integrated humanoid plattfrom for sensory-motor control," *IEEE-RAS International Conference on Humanoid Robots*, 2006.

[4] K. Yao and S. Nakamura, "Sequential noise compensation by sequential monte carlo methods," *Adv. Neural Inform. Process. Syst.*, vol. 14, Sep. 2002.

[5] R. Singh and B. Raj, "Tracking noise via dynamical systems with a continuum of states," *Proc. of ICASSP*, 2003.

[6] B. Raj, R. Singh, and R. Stern, "On tracking noise with linear dynamical system models," *Proc. of ICASSP*, 2004.

[7] F. Faubel and M. Wölfel, "Coupling particle filters with automatic speech recognition for speech feature enhancement," *Proc. of Interspeech*, Sep. 2006.

[8] S. Julier and Uhlmann J.K., "A general method for approximating nonlinear transformations of probability distributions," Nov. 1996.

[9] N.J. Gordon, D.J. Salmond, and A.F.M. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEEE Proceedings on Radar and Signal Processing*, vol. 140, pp. 107–113, Sep. 1993.

[10] C.P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer Texts in Statistics. Springer, second edition, 2004.

[11] F. Faubel and M. Wölfel, "Overcoming the vector tailor series approximation in speech feature enhancement — a particle filter approach," *Proc. of ICASSP*, 2007.

[12] M. Wölfel and J.W. McDonough, "Minimum variance distortionless response spectral estimation, review and refinements," *IEEE Signal Processing Magazine*, vol. 22, no. 5, pp. 117–126, Sept. 2005.

[13] M.J.F. Gales, "Semi-tied covariance matrices for hidden markov models," *IEEE Transactions Speech and Audio Processing*, vol. 7, pp. 272–281, 1999.

[14] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language*, pp. 171–185, 1995.

[15] F. Kraft, R. Malkin, T. Schaaf, and A. Waibel, "Temporal ICA for classification of acoustic events in a kitchen environment," *Proc. of Interspeech*, 2005.