2011 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 25-30, 2011. San Francisco, CA, USA

Incremental Learning for Ego Noise Estimation of a Robot

Gökhan Ince, Kazuhiro Nakadai, Tobias Rodemann, Jun-ichi Imura, Keisuke Nakamura, and Hirofumi Nakajima

Abstract— **Using pre-recorded templates to estimate and suppress the ego noise of a robot is advantageous because this method is able to cope with the non-stationarity of this particular type of noise. However, standard template-based estimation requires human intervention in the offline training sessions, storage of large amounts of data and does not adapt to the dynamical changes in the environmental conditions. In this paper we investigate the feasibility of an incremental template learning system to tackle these drawbacks. Incremental learning enables the system to acquire new templates on the fly and update the older ones appropriately. Whilst allowing the system to continually increase its knowledge and enhancing its estimation performance, this learning scheme also reduces the size of the database. We evaluate the performance of the proposed noise estimation method in terms of its estimation accuracy, quality of speech signals enhanced by spectral subtraction method, and size of database. The experimental results show that our system compared to conventional single-channel noise estimation methods achieves better performance in attaining signal quality and improving word correct rates.**

I. Introduction

The prediction of ego noise, a type of noise generated by the fans, hardware and motors of a robot, plays a significant role in suppressing this noise and achieving good performance in various applications like Automatic Speech Recognition (ASR) [1] and Sound Source Localization (SSL) [2] while the robot is in motion. Fundamentally, the overall ego noise of a robot depends on the contribution of each noise signal stemming from different motors and the static fan noise. Therefore the problem gets even tougher, the larger number of motors are employed for a motion, which means that the noise is even more severe for a moving robot with many Degrees of Freedom (DoF). Besides, in a standard task involving robot motions, acoustic properties of the motor noise such as the power and frequency distribution in the spectrum, as well as the locations and number of the active motors dynamically change at each time instance. Sawada *et al.* [3] uses semi-blind signal separation to obtain ego noise estimates by attaching additional noise sensors, e.g. Non-Audible Murmur (NAM) microphones inside the robot, but it requires hardware modifications on the robot. Conventional noise estimation techniques [4]-[6] fail in estimating the non-stationary ego noise because they are neither able to discriminate ego-motion noise from non-stationary speech signals, nor fast enough to track the rapid changes in ego noise. In contrast to stationary noise estimation methods, template estimation is well-suited to capture the dynamic nature of the motion data represented by a sequence of observations. Based on these observations, it is possible to associate either a motion command [7] or the discrete time series data representing the angular state of motors [8],[9] with another discrete time series data representing the ego noise spectra and predict an arbitrary sequence of associated data. The so-called Template-based Estimation (TE), provides a framework for a more effective ego noise estimation that relies on templates representing the instantaneous noise.

For example, Nishimura *et al.* [7] estimated the ego noise using only motion commands as representative labels for the corresponding ego noise spectrum. A query in the repertoire of labeled motion commands was used to select the appropriate ego noise template from a database consisting of templates time-aligned and averaged manually. Ito *et al.* [8] developed a new approach of frame-by-frame based prediction with an Artificial Neural Network (ANN). The trained network had to predict the noise spectrum by using a discrete time series data of angular velocities of the joints. However, they concentrated on a small robot with limited number of DoF. For a huge dataset of motion repertoire, ANN will have a slow training speed and online adaptation is difficult to achieve. Instead, Ince *et al.* [9] proposed the usage of a template database due to its efficiency and enhanced the accuracy of the templates further by incorporating more information related to the joints, such as angular positions, velocities and accelerations. The strengths of all TE methods presented so far, unlike the conventional stationary noise estimators are that they are not dependent on Signal-To-Noise Ratio (SNR), not prone to Voice Activity Detection (VAD) errors and adaptation latency to the actual noise is theoretically zero.

The typical problem tightly coupled to offline training for TE is that they are not well suited to real-time, real-world applications due to insufficiently accurate data in ego noise templates of unknown motions (i.e., missing templates) or the differences in environmental conditions from the conditions in the training session. Creating databases manually is

Gökhan Ince, Kazuhiro Nakadai, and Keisuke Nakamura are with Honda Research Institute Japan Co., Ltd. 8-1 Honcho, Wako-shi, Saitama 351-0188, Japan {gokhan.ince, nakadai, keisuke@jp.honda-ri.com}

Hirofumi Nakajima is with Honda Research Institute Japan Co., Ltd. (currently with Kogakuin University) nakajima@cc.kogakuin.ac.jp

Tobias Rodemann is with Honda Research Institute Europe GmbH, Carl-Legien Strasse 30, 63073 Offenbach, Germany tobias.rodemann@honda-ri.de

Gökhan Ince, Kazuhiro Nakadai, and Jun-ichi Imura are with Dept. of Mechanical and Environmental Informatics, Tokyo Institute of Technology 2-12-1-W8-1, O-okayama, Meguro-ku, Tokyo, 152-8552, Japan imura@mei.titech.ac.jp

978-1-61284-455-8/11/$26.00 ©2011 IEEE

also a tedious work. Furthermore, in case of long training sessions, the data can grow rapidly and expand enormously unless there is an explicit template discarding or update algorithm. To sum up, there are two major drawbacks of template estimation methods: 1) constantly growing database of templates, and 2) incapacity of coping with changing environmental noise in real world.

To our knowledge this paper is the first attempt to create an autonomous and adaptive ego noise estimation technique preventing the expansion of the size of the template database. In this paper, we design an incremental learning mechanism for learning/updating/discarding ego noise templates, which also allows us to tackle the curse of dimensionality problem caused by the large number of DoF of a robot. Whereas 1) is the primary concern of this paper, 2) can be solved with our proposed method only suboptimally. Therefore, to enhance our system against 2), we incorporate an extended noise estimation framework as proposed in [10], which integrates TE with a Histogram-based Recursive Level Estimation (HRLE)-based stationary noise estimator. We examine the capabilities and performances of HRLE [6], TE [9] and the unified noise estimator [10] with a special focus on the influence of the incremental learning. Our main goals will be (1) to improve the results obtained with performance criteria such as Normalized Noise Estimation Error (NNEE), SNR and Log-Spectral Distortion (LSD), (2) to increase the robustness of other speech processing applications to noise (e.g. ASR) and (3) to reduce the size of the database.

II. Ego Noise Reduction System

In this section we first outline the basic architecture of the ego noise reduction system, and then focus on the estimation block based on template estimation. Fig. 1 shows the general configuration for single-channel noise reduction. Suppose an input signal $y(t)$ of time sample t is given such as

$$y(t) = x(t) + n(t), \tag{1}$$

where $x(t)$ is a target signal and $n(t)$ is the noise signal with Ego Noise (EN) and BackGround Noise (BGN). The complex input spectrum $Y(k,l)$ of frequency bin k and time frame l is obtained from

$$Y(k,l) = \sum_{t=0}^{t=W-1} y(t+lM)w(t)\exp\{j(2\pi/W)tk\}, \tag{2}$$

where W is the window length, M is the shift length and $w(t)$ is the window function. The gain calculation process calculates the optimum gain $G(k,l)$ that yields the final estimated target spectrum as

$$\hat{X}(k,l) = G(k,l)Y(k,l). \tag{3}$$

The equation for computing $G(k,l)$ is derived from the reduction method, e.g., for spectral subtraction (SS) [11]:

$$G_{SS}(k,l) = \sqrt{\max\left[\frac{|Y(k,l)|^2 - \lambda_{tot}(k,l)}{|Y(k,l)|^2}, \beta\right]}, \tag{4}$$

where $\lambda_{tot}(k,l)$ shows the estimated spectrum of both stationary and non-stationary noise, *max* shows the maximum value calculation and β is the flooring parameter.

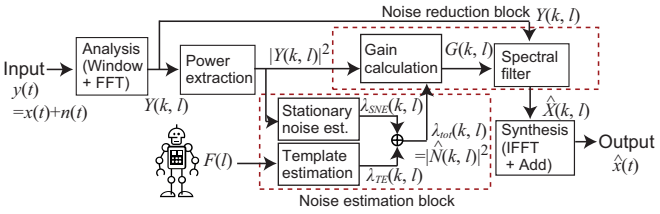

Fig. 1. General configuration for single-channel noise reduction

A. Noise Estimation Block

Template-based Estimation [9] utilizes encoders attached to the motors of the robot, which measure the angular position of each joint. During the motion of the robot, actual position ($\theta(l)$) information regarding each motor is acquired regularly. Additionally, using the difference between consecutive sensor outputs, velocities ($\dot{\theta}(l)$) and accelerations ($\ddot{\theta}(l)$) are calculated. Considering that J joints are active, $3J$ attributes are generated. Each feature is normalized to [0 1] so that all features have the same contribution on the prediction. The resulting feature vector has the form of $F(l) = [\theta_1(l), \dot{\theta}_1(l), \ddot{\theta}_1(l), \theta_2(l), \dot{\theta}_2(l), \ddot{\theta}_2(l), \ldots, \theta_J(l), \dot{\theta}_J(l), \ddot{\theta}_J(l)]$.

Conventional TE methods [7]-[9] lack the ability of filtering the BGN and therefore cannot perform adaptation to overall noise (EN+BGN) in an environment with changing noise conditions. They can only reproduce the templates that exist in the database, hence they can only reflect the noise conditions in the training session. Since EN is mixed with BGN in a realistic situation, we assume the overall noise (EN+BGN) consists of both stationary $N_s(k,l)$ and non-stationary $N_n(k,l)$ portions such as

$$N(k,l) = N_s(k,l) + N_n(k,l). \tag{5}$$

We propose to use stationary and non-stationary noise estimation methods in series as in Fig. 2 in the template generation (database creation) phase so that one feature vector is assigned only to the non-stationary motor noise spectral vector $|N_n(l)|^2$ and used to label the instantaneous noise fragment [10]. We call this data block $T(l) = [F(l) : |N_n(l)|^2]$ a *parameterized template*. In the noise estimation phase, however, a unified framework for noise estimation consisting of two parallel and independent processes as in Fig. 1 is used. While recursive averaging (i.e., HRLE) takes care of $N_s(k,l)$, the background and stationary portion of ego noise (i.e. fan/hardware noise), TE tackles the remaining non-stationary noise portion of motor noise $N_n(k,l)$. The power input spectrum calculated as $|Y(k,l)|^2$ is used to estimate power spectrum of stationary portion of the noise, $\lambda_{SNE}(k,l)$ [6]. The total noise power, $\lambda_{tot}(k,l) = \lambda_{SNE}(k,l) + \lambda_{TE}(k,l)$, is eventually used to compute the gains as in Eq. 4 and extract the refined signal as in Eq. 3.

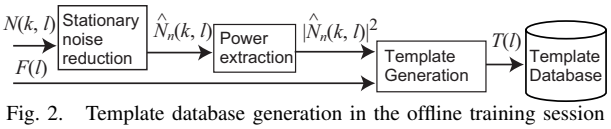

Fig. 2. Template database generation in the offline training session

III. Incremental Template Learning

In Sec. III-A we explain the basic classification model incorporated into the core of template learning and estimation

algorithm. Sec. III-B proposes methods to enhance the classification model and finally the template learning algorithm is discussed in Sec. III-C.

A. Classification Model

Suppose we have a robot with 30 DoF. We gather templates in every frame (i.e. 10 milliseconds), which contain feature vectors $F(l)$ consisting of 90 features and spectral vectors $|\hat{N}_n(l)|^2$ consisting of 128 spectrotemporal values, all represented in floating point values. It is easy to imagine that in several minutes huge streams of continuous data will be stored that must be processed, learned and updated in an online fashion. For high dimensional learning tasks, the performance of the machine learning algorithm plays a crucial role. One alternative is using ANNs with sigmoidal activation functions, which learn slowly in high dimensional spaces and are vulnerable to unlearning of relevant knowledge when trained on new data points [12]. Instead, we prefer to use a non-parametric, instance-based classification technique like the Nearest Neighbor (NN) algorithm because it is easy to implement, does not need any a priori knowledge about the data and the output of the NN algorithm can be interpreted as an a posteriori probability of the input pattern being the estimated pattern [13]. The last point is especially important because it provides us a measure of performance allowing to update existing templates in our incremental learning algorithm based on the relative template confidence levels.

For a given database $\boldsymbol{F}$ of template feature vector in $3J$-dimensional feature space and a query feature vector $Q(l)$, we find the closest feature vector in $\boldsymbol{F}$ to $Q(l)$. The distance is measured by the Euclidean distance between two feature vectors $Q(l) = (Q_1(l), Q_2(l), ..., Q_{3J}(l))$ and $F(l) = (F_1(l), F_2(l), ..., F_{3J}(l))$, where $F(l)$ is an element $\boldsymbol{F}$.

$$d(Q(l), F(l)) = ||Q(l) - F(l)|| = \sqrt{\sum_{j=1}^{3J} (Q_j(l) - F_j(l))^2} \quad (6)$$

The spectral vector $|N_n(l)|^2$ stored in the template $T(l)$ with $F(l)$ having the shortest distance to $Q(l)$ is selected as the ego noise estimate $\lambda(l)$.

B. Extensions of the Basic Classification Model

In order to improve the robustness of template prediction, we made the following modifications on the classification method.

1) Inverse distance weighted average (IDWA): Instead of using 1-NN, we assign confidence-based weights ω^k to K nearest templates $F^1(l) \dots F^K(l)$ by giving the highest weight to the closest neighbor, and compute the final spectral vector $\bar{\lambda}_{TE}(l)$ from their Inverse Distance Weighted Average (IDWA) of all nearest candidates such as in Eq. 7.

$$\bar{\lambda}_{TE}(l) = \sum_{n=1}^{K} \underbrace{\left(\frac{\frac{1}{d(Q(l), F^n(l))}}{\sum_{m=1}^{K} \frac{1}{d(Q(l), F^m(l))}} \right)}_{\omega^n} \cdot \lambda_{TE}^n(l) \quad (7)$$

IDWA assumes that templates closer to the query point are more representative than templates further away. The denominator term is used for normalization of the ω^k such that $\sum_{k=1}^{K} \omega^k = 1$.

2) K-dimensional trees: To increase the speed of K-NN, we suggest to utilize tree structures, such as K-dimensional trees (KD-trees) [14]. So, the search is conducted more efficiently by using the tree properties that quickly eliminate large portions of the search space. Because the computation cost is also related to the code implementation quality, the reader is advised to address to reliable machine learning software[1], which was able to provide real-time computation for our system implementation.

C. Learning Algorithm

Incremental learning is essential for adaptive generation of the template database because it makes use of previously learned knowledge about the templates to speed up the learning. It makes the noise estimation module more robust because errors in the training set can be corrected *during operation* and it enables the system to adapt to partially-known or dynamic environments. Therefore, it is expected that the performance will gradually improve in time.

In the proposed system, the task of the learning system is to autonomously extract and learn templates. The system checks whether the acquired audio signal is mixed with a speech signal based on the decision of VAD and discards it if it is not only ego noise. This continuous VAD loop determines the onset and offset times of the template update interval. During this interval, the system also decides if each observed template is a known template or a new template to be learned. The observed template is searched in the trained database and its similarity with other templates in the database is computed using the same distance metric as in Eq.6. Based on the comparison of $d_{min}(l)$, the smallest distance $d(Q(l), F(l))$ in $\boldsymbol{F}$, with a given fixed distance threshold, T, the current template is either used to update the old template or it is inserted into the database as a new template. When the similarity is low, the template is treated as missing template and inserted into $\boldsymbol{F}$; otherwise the adaptive update mechanism is active, which computes the weighted sum of the old and current template by laying the focus more on recently-acquired templates and less on earlier observations. The contribution of past templates are reduced by introducing a forgetting factor η with $0 \leq \eta \leq 1$, which helps to provide a moderate balance between adaptivity (learning quality) and stability (robustness against errors). The former is achieved by using lower η, whereas higher η causes stability. The pseudo-code of the incremental learning algorithm is shown below.

One key aspect of this incremental learning algorithm is the rebuilding of the KD-tree due to practical concerns. Insertion-based incremental construction of a KD-tree for a long time is problematic, because the tree becomes unbalanced eventually. The re-balancing task is tedious and

[1] e.g., http://www.cs.umd.edu/~mount/ANN/

repeating it at each insertion is also costly. Therefore, we rebuild the data structure in constant time intervals determined by τ. Until the next rebuilding phase, new templates to be inserted are stored in a temporary buffer.

while (state(VAD)=NON-SPEECH) **do**
  if $d_{min}(Q(l),F(l)) \geq T$ **then**
    $[F^{new} : |N_n^{new}|^2] \leftarrow [F(l) : |N_n(l)|^2]$
  else
    $[F^{old} : |N_n^{upd}|^2] \leftarrow [F^{old} : \eta|N_n^{old}|^2 + (1-\eta)|N_n(l)|^2]$
  end if
  if (timer=τ) **then**
    Rebuild the tree and reset the timer
  end if
end while

IV. Evaluation

In this section, we assess the estimation and suppression capabilities of (1) conventional TE [9] method whose templates represent both stationary and non-stationary noise, and (2) proposed method in Sec. II-A whose templates represent only motor noise by applying them to the noise signals consisting of ego noise and environmental background noise. Since the current implementation does not have a noise-robust VAD, we intentionally by-pass (exclude) it in the training session by recording only ego noise. One set of noise data (200 seconds long) for training and three sets of noise data (100 seconds long) for testing are collected during a continuous head motion of 2 Degree of Freedoms (DoF) and arm motion of 4 DoF (see Fig. 3), which generates 18 features. The recording environment is a room with the dimensions of 4.0 m×7.0 m×3.0 m with a reverberation time (RT_{20}) of 0.2 sec. The performance of all methods are compared under 4 different SNR conditions for the same signal segments as in Fig. 4. Condition (1)-(2): Noise energy is fixed, speech signals are amplified to yield $SNR_{(1)} = 3dB$ and $SNR_{(2)} = -3dB$; Condition (3)-(4): Gaussian white noise is added to (2) to represent changing conditions of static BGN (e.g. entering into a new room or turning on the air conditioner) with $SNR_{(3)} = -3.1dB$ and $SNR_{(4)} = -3.2dB$. The parameters of the HRLE are selected in a way that they are optimal for stationary noise estimation [6]. The optimal value of $\eta = 0.9$ for incremental learning, which pursues stability rather than adaptivity is found empirically. For a database size such as in our experiments, real-time conditions are properly provided with $\tau = 5$ frames. A minor spectral floor $\beta = 0.1$ is used in the SS stage.

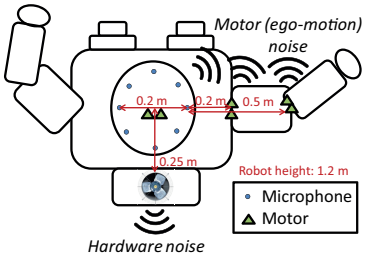

Fig. 3. Experimental setup

A. Evaluation Criteria

1) Normalized Noise Estimation Error (NNEE): NNEE computes the error of the noise estimate normalized by the energy of the actual noise using the following formula:

$$\bar{\varepsilon} = \frac{1}{L}\sum_{l=1} 10 \cdot \log_{10}\left(\frac{\sum_{k=0}^{M} ||N(k,l)|^2 - |\hat{N}(k,l)|^2|)}{\sum_{k=0}^{M} ||N(k,l)|^2|}\right), \quad (8)$$

where L is the number of frames.

2) Segmental SNR: The average of the SNR values is calculated for segments of audio data such as:

$$SNR = \frac{1}{L}\sum_{l=1}^{L} 10 \cdot \log_{10}\left(\frac{\sum_t x^2(t)}{\sum_t (x(t)-\hat{x}(t))^2}\right). \quad (9)$$

3) Log-Spectral Distortion [15]: This evaluation measure computes the reconstruction error of the clean speech by comparing the enhanced speech signal $\hat{X}(k,l)$ with the original speech $X(k,l)$ in the log domain as follows:

$$LSD = \frac{1}{L}\sum_{l=1}^{L}\left(\frac{1}{K}\sum_{k=1}^{K}\left[\mathscr{L}X(k,l) - \mathscr{L}\hat{X}(k,l)\right]^2\right)^{1/2}, \quad (10)$$

where $\mathscr{L}X(k,l) \triangleq \max\{20\log_{10}|X(k,l)|, \delta\}$ is the log spectrum confined to about 50 dB dynamic range, hence $\delta = \max_{k,l}\{20\log_{10}|X(k,l)|\} - 50$.

4) Automatic Speech Recognition: The noise signals are mixed with clean speech utterances used in a typical human-robot interaction dialog and recorded by us. This Japanese word dataset includes 236 words for 4 female and 4 male speakers. We used a clean acoustic model trained with Japanese Newspaper Article Sentences (JNAS) corpus, 60-hour of speech data spoken by 306 male and female speakers, hence the speech recognition is a word and speaker-open test. We used 13 static Mel-Scale Log Spectrum (MSLS) features, 13 delta MSLS features and 1 delta power feature. Speech recognition results are given as average Word Correct Rates (WCR) of instances from the noisy test set.

B. Results

1) Learning performance: To assess the learning capability of our system with respect to threshold T, we evaluated the estimation error (NNEE) in incremental steps, i.e. after repeating the same motion N times ($1 \leq N \leq 20$). Because we used bounded ([0 1]) features, the values of T are also bounded to $[0\ \sqrt{3J}]$. Fig. 5 demonstrates the tendency of reduced error (ε) with respect to the increased repetitions. The settings denoted as "$T \to 0$" indicates that there is a continuous insertion of every incoming template into the database like in conventional template estimation methods [8], [9] and "$T \to \infty$" indicates that there is only one single (mean) template updated during all repetitions. They both yield the baseline performance. Because HRLE is a stationary noise estimator, it cannot deal with the non-stationary ego noise and shows also a poor performance. The error decreases when T is sufficiently low. We also observe that there is a negative correlation between the number of templates stored and the value of T (see Fig. 6). Therefore,

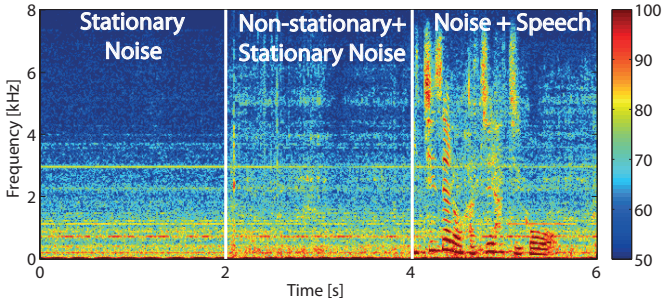

Fig. 4. Noisy spectrogram

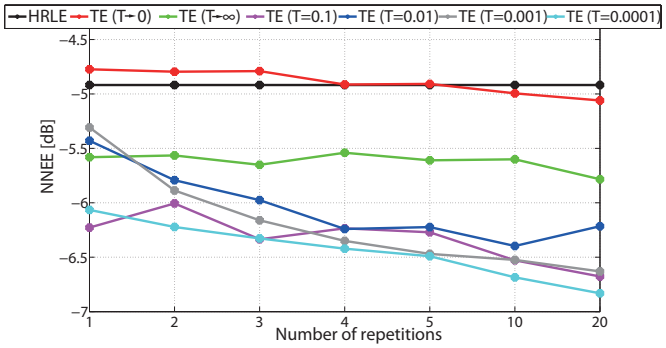

Fig. 5. Estimation error in relation with the number of iterations

the system designer must consider the trade-off between the size of the template database (i.e., data explosion) and distance threshold. Since "$T = 0.0001$" among others yielded the smallest error in our experiments, it is selected as the optimal T for the incremental learning and we continue to evaluate the final estimation and suppression results based on the templates obtained with this value after 20 repetitions. Because the number of templates almost saturate after several repetitions, the tree reordering will not pose a problem.

TABLE I

EGO NOISE ESTIMATION PERFORMANCE FOR ALL METHODS

SNR [dB]	$\bar{\varepsilon}$ for given interval	HRLE	TE $T \to 0$	TE $T \to \infty$	TE $T = 0.0001$
(1) 3.0	N_s	-5.81	-5.08	-1.74	**-6.77**
	$N_s + N_n$	-4.61	-4.89	-4.44	**-6.59**
	$N + Speech$	-4.92	-5.06	-4.78	**-6.84**
(2) -3.0	N_s	-5.81	-5.08	-1.74	**-6.77**
	$N_s + N_n$	-4.61	-4.89	-4.44	**-6.59**
	$N + Speech$	-4.84	-5.06	-4.78	**-6.84**
(3) -3.1	N_s	**-7.96**	-4.95	-4.00	-6.69
	$N_s + N_n$	-6.30	-4.95	-5.28	**-6.71**
	$N + Speech$	-6.71	-5.11	-5.57	**-6.99**
(4) -3.2	N_s	**-8.87**	-4.52	-5.04	-5.96
	$N_s + N_n$	**-7.1**	-4.76	-5.56	-6.41
	$N + Speech$	**-7.42**	-4.92	-5.83	-6.69

2) Performance of conventional TE with incremental learning: It is also important to investigate the performance of these methods in the presence of speech such as depicted in Fig. 4. Final estimates after the 20^{th} iteration can be seen in Fig. 7 and Fig. 8 representing TE with $T \to 0$ and TE with $T = 0.0001$, respectively. The smoothness of the spectrum in Fig. 8 reflects the more accurate estimation results as given in Tab. I. In conditions (1) and (2), TE ($T = 0.0001$) performed better than other methods for any given time interval, surprisingly even in the case of stationary noise, where HRLE is more suitable to apply. The reason is that

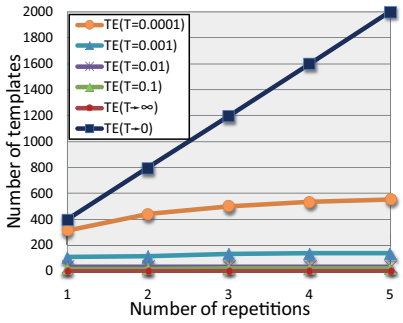

Fig. 6. Number of templates in relation with the number of iterations

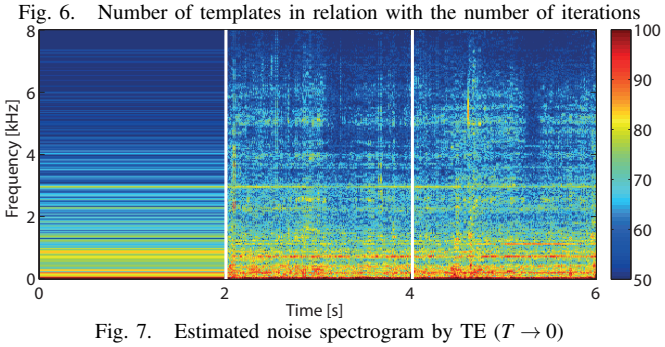

Fig. 7. Estimated noise spectrogram by TE ($T \to 0$)

the BGN conditions in (1) and (2) are the same with the database. However, unfamiliar BGN conditions such as in (3) and (4) degrade the accuracy of the TE because the portion of the stationary noise, compared to the ego noise, becomes more dominant in the overall noise energy. We can clearly see the tendency of HRLE outperforming TE ($T = 0.0001$) gradually in the conditions of slightly increased BGN power. This justifies the usage of the concatenated processing of HRLE and TE (referred as HRLE+TE) as proposed in Fig. 1 to compensate the deteriorating performance of TE. An alternative way to deal with the changing acoustic conditions is letting the templates adapt to those conditions entirely using the incremental update mechanism, but it may take a long time and the improvements are in rather small steps during the adaptation process with a high η, which makes this option less feasible.

We also analyze the distribution of the estimation error (NNEE) over the frequency bins (see Fig. 9). One important advantage of TE is that the error is almost evenly distributed to all frequencies. In contrast, the spectral distortion of the noise estimate provided by semi-blind source separation based on internal NAM microphones [3] is rather large at low frequencies (see also Fig. 5 in [3]), which are known to heavily contain acoustic features of speech.

3) Performance of proposed noise estimation framework with incremental learning: Finally, we evaluate the noise reduction performance by using the system depicted in Fig. 1. As Tab. II demonstrates, TE-based SS with incremental learning achieves the smallest LSD and largest WCRs among all methods for the trained conditions (1) and (2). Besides, HRLE+TE with incremental learning attains the second-best results to TE, which can allow us to make a small compromise between the best performance and adaptivity of the noise estimation system. Conditions (3) and (4) show the results for the simulation of changing ambient noise. We observe that the higher the portion of the background

TABLE II

EGO NOISE REDUCTION PERFORMANCE FOR ALL METHODS

Estimation Method	$SNR_{(1)} = 3dB$			$SNR_{(2)} = -3dB$			$SNR_{(3)} = -3.1dB$			$SNR_{(4)} = -3.2dB$		
	SNR	*LSD*	*WCR*	*SNR*	*LSD*	*WCR*	*SNR*	*LSD*	*WCR*	*SNR*	*LSD*	*WCR*
NP	3.00	9.7	78	-3.0	11.2	28.3	-3.1	12.6	25.5	-3.2	14	22.8
HRLE	3.96	8.94	84.1	-1.2	10.2	47.2	-0.75	10.0	42.1	-0.76	9.93	38.2
Standard TE ($T \to 0$)	5.49	8.51	87.4	2.05	8.73	58.7	1.97	9.75	51.7	1.77	11.1	44.5
Standard TE ($T \to \infty$)	4.92	8.20	85.5	2.18	8.31	64.1	2.42	8.90	59.8	1.90	10.09	52.6
Standard TE ($T = 0.0001$)	5.24	**8.03**	**89.3**	2.43	**8.18**	**69.9**	1.97	9.02	64.1	2.23	10.38	55.8
HRLE+TE ($T \to 0$)	**6.02**	8.66	86.2	**2.74**	8.88	54.8	**2.63**	9.03	52.0	**2.77**	9.24	46.5
HRLE+TE ($T \to \infty$)	5.01	8.36	84.6	2.47	8.45	62.7	2.59	8.69	59.6	2.02	9.38	55.2
HRLE+TE ($T = 0.0001$)	5.46	8.20	88.8	2.62	8.31	68.6	2.61	**8.66**	**64.6**	2.45	**9.23**	**59.9**

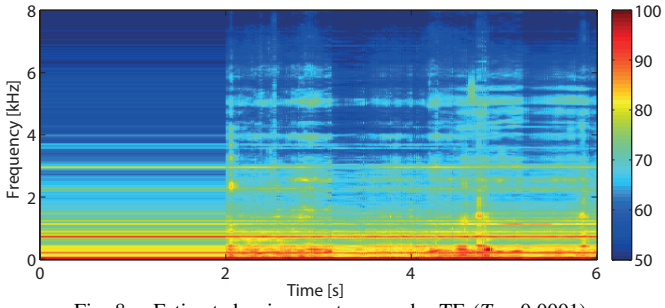

Fig. 8. Estimated noise spectrogram by TE ($T = 0.0001$)

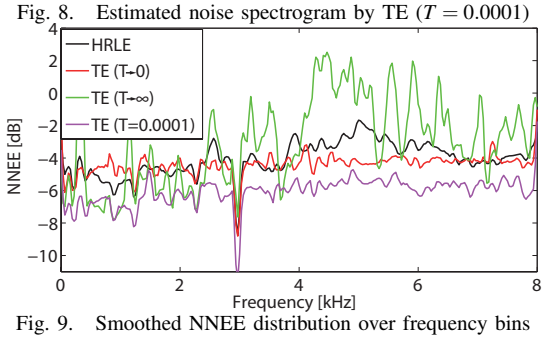

Fig. 9. Smoothed NNEE distribution over frequency bins

noise, the more effective HRLE method gets. Under these conditions, HRLE contributes more to cancelling the overall noise by eliminating the background noise. Hence in its combination with TE, TE deals only with the non-stationary part of the overall noise regarding the ego-motion noise. This kind of configuration increases the robustness of the noise suppression system and makes it independent of any change in the environmental noise condition. In terms of SNR, HRLE+TE with incremental learning is only outperformed by HRLE+TE with $T \to 0$.

V. CONCLUSION

In this paper we proposed an automated learning mechanism for an adaptive ego noise estimation framework consisting of a stationary noise estimation (HRLE) and non-stationary noise estimation (TE) in series. We assessed the learning, estimation and suppression performance of this template-based single-channel noise reduction method in the presence of background noise and ego-motion noise. The proposed method exhibits high robustness even against changing conditions of the environment. We showed that the incremental learning contributes to precise estimation of overall noise and a high ASR accuracy under various SNR conditions. Future plans include a performance evaluation in real world for a dancing and beat tracking robot. Furthermore, as the number of observed templates becomes huge, the robot must have an effective way of storing the acquired database for easier retrieval and organization. In this case, a more advanced database storage technology than KD-trees might be needed. A second direction in our future work focuses on the integration of VAD to make our method fully online. To resolve problems related to low SNR, we plan to apply robust audiovisual VAD mechanisms.

REFERENCES

[1] G. Ince, K. Nakadai, T. Rodemann, Y. Hasegawa, H. Tsujino, and J. Imura, "A Hybrid Framework for Ego Noise Cancellation of a Robot", *Proc. of the IEEE/RSJ International Conference on Robotics and Automation (ICRA)*, pp.3623-3628, 2010.

[2] G. Ince, K. Nakamura, F. Asano, H. Nakajima and K. Nakadai, "Assessment of General Applicability of Ego Noise Estimation - Application to Automatic Speech Recognition and Sound Source Localization-", to appear in *Proc. of the IEEE/RSJ International Conference on Robotics and Automation (ICRA)*, 2011.

[3] H. Sawada, J. Even, H. Saruwatari, K. Shikano, T. Takatani, "Improvement of Speech Recognition Performance for Spoken-Oriented Robot Dialog System Using End-fire Array", *Proc. of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, pp.970-975, 2010.

[4] R. Martin, "Spectral Subtraction Based on Minimum Statistics", *Proc. Eur. Signal Processing*, 1182-1185, 1994.

[5] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments", *Signal Processing*, vol 81, pp.2403-2481, 2001.

[6] H. Nakajima, G. Ince, K. Nakadai and Y. Hasegawa, "An Easily-configurable Robot Audition System using Histogram-based Recursive Level Estimation", *Proc. of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, pp.958-963, 2010.

[7] Y. Nishimura, M. Nakano, K. Nakadai, H. Tsujino and M. Ishizuka, "Speech Recognition for a Robot under its Motor Noises by Selective Application of Missing Feature Theory and MLLR", *ISCA Tutorial and Research Workshop on Statistical And Perceptual Audition*, 2006.

[8] A. Ito, T. Kanayama, M. Suzuki, S. Makino, "Internal Noise Suppression for Speech Recognition by Small Robots", *Interspeech 2005*, pp.2685-2688, 2005.

[9] G. Ince, K. Nakadai, T. Rodemann, Y. Hasegawa, H. Tsujino, and J. Imura "Ego Noise Suppression of a Robot Using Template Subtraction", *Proc. of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, pp.199-204, 2009.

[10] G. Ince, H. Nakajima, K. Nakamura, T. Rodemann, K. Nakadai and J. Imura "Assesment of Single-channel Noise Estimation Methods for Ego Noise", to appear in *Proc. of IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, 2011.

[11] J. Deller, *Discrete-Time Processing of Speech Signals*, IEEE Press, 2000.

[12] S. Schaal and C. G. Atkeson, "Constructive incremental learning from only local information", *Neural Computation*, vol. 10, 2047-2084, 1998.

[13] R. Duda and P. Hart *Pattern Classification and Scene Analysis*, Wiley, New York, 1979.

[14] J. L. Bentley, "K-d trees for semidynamic point sets", *Proc. of ACM Symposium on Computational Geometry*, pp.187-197, 1990.

[15] J. Benesty, M. M. Sondhi, Y. Huang, *Speech Processing*, Springer-Verlag, 2008.