How Do I Sound Like? Forward Models for Robot Ego-Noise Prediction

Antonio Pico[1], Guido Schillaci[1], Verena V. Hafner[1] and Bruno Lara[2]

***Abstract*—How do robots sound like? Robot ego-noise, that is the sound produced by a robot while moving around, is an important factor that can affect the way an artificial agent perceives the environment and interacts with it. In robot audition, for example, ego-noise is usually addressed due to its effects on the quality of the auditory input signal, as it can severely impact the performance of processes such as speech recognition. Nonetheless, robot ego-noise can carry out very useful information about the robot embodiment or about the external environment. In this study, we present a mechanism for learning and for predicting the auditory consequences of self-generated movements on a custom robotic platform. We show two experiments based on a computational model capable of performing forward predictions. First, we demonstrate that the system can classify motor behaviours by comparing the noise they produce with that of simulated actions. Thus, we show that, by using similar processes, the robot can detect unexpected environmental conditions, such as changes in the inclination of the surface it is walking on.**

I. Introduction

Robot audition is a recent research field that addresses the capability of listening in artificial agents. Its aim is to improve the auditory capabilities of robots so that they can better interact with humans and with their surroundings [1]. One of the biggest challenges in robot audition is the presence of ego-noise, that is the noise that the robot *itself* generates while moving around. Humans and animals generate auditory noise when interacting with the environment. For instance, our footsteps produce noise, which varies according to the surface we are walking on. We produce noise when we type on a keyboard or when we put an object on a table, or even when we just breath. Similarly, robots generate noise when they move around, for example due to the friction of their motors and actuators. Ego-noise can be present also when the robot is not moving at all. For instance, noise is produced by the cooling fan of an onboard processor, especially when the fan is close enough to the internal microphones of the robot.

Ego-noise can be an important issue in robot audition due to its effect on the auditory input signals captured from the microphones. For instance, ego-noise can severely decrease the performance of a speech recogniser or of a sound source localiser. Several researchers spent efforts on noise, and robot ego-noise, suppression. The general approach is to create a model of the noise that is subtracted from the noisy signal. Such a model is usually built on recordings of the noise that the system produces in an idle state. Robots, however, can produce different noises while moving around. First attempts to solve this problem have been proposed. For example, Ince et al. [2] implemented large noise template databases that were used for ego-noise prediction and subtraction. However, building up a model of all the possible noises that a robot can produce in all the possible environments is very challenging.

Robot ego-noise has not only a negative impact in robot audition. In fact, it can carry out useful information about the movements that the robot is executing and about some of the characteristics of the external environment. In this paper, we present a mechanism for learning and for predicting ego-noise on a custom robotic platform that we designed and built in our department (see section II-A for more details about the platform). In particular, we adopted forward models [3], [4] as a computational tool for encoding the dynamics of the motor system of the robot and the effect of self-produced movements on the perceived ego-noise. Forward models incorporate knowledge about sensory outcomes of self-generated actions. In the context of this work, they can provide an artificial agent with the capability to predict the ego-noise that would be produced by an *intended* motor action. In a preliminary experiment, we showed similar capabilities implemented on a humanoid robot [5]. In this work, we exploit the predictive capabilities of forward models in two experimental setups (see section II-C for more details). First, we present a classification experiment in which we show how the auditory predictions provided by a set of trained forward models can be used for determining the velocity profile a sensorimotor input belongs to. Next, we show how the auditory predictions provided by a forward model trained with data gathered using a self-exploration behaviour, namely random motor babbling, can be used to detect changes in the external environment. As an example, we demonstrate how the robot can detect a change in the inclination of the surface where it is moving on. Random motor babbling is inspired by a behaviour exhibited by infants during early developmental stages, through which they explore their bodily capabilities and acquire coordination skills [6], [7].

The results of the experiments are presented in section III. Finally, we present the conclusions of this study in section IV.

[1]Antonio Pico, Guido Schillaci and Verena V. Hafner are affiliated with the Adaptive Systems Group, Humboldt-Universität zu Berlin, Germany. E-mails: {pivilla, guido.schillaci, hafner}@informatik.hu-berlin.de

[2]Bruno Lara is affiliated with the Cognitive Robotics Group, Center for Science Research Universidad Autónoma del Estado de Morelos, Cuernavaca, Mexico. E-mail: bruno.lara@uaem.mx

II. Methodology

A. Robotic Platform

We developed a two-wheeled mobile robot for our experiments (Figure 1). This was made with the aim of having low-level control over the robot's hardware and software, which was crucial to obtain a precise synchronization between motor commands and sensory data.

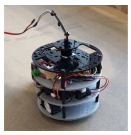

Fig. 1. Mobile robot with two wheels and motors used in both experiments.

The platform consists of a stack of four 3D-printed circular plates with a diameter of 13 cm and a total height of 11 cm. The wheels are powered by two DC gearmotors placed in a differential configuration, each of them equipped with a quadrature magnetic encoder for speed measurements. The sensory inputs also include a microphone for audio recording and seven infrared distance sensors for obstacle avoidance.

1) Hardware configuration: Figure 2 shows a diagram of the hardware configuration of the mobile robot, which consists of the following modules:

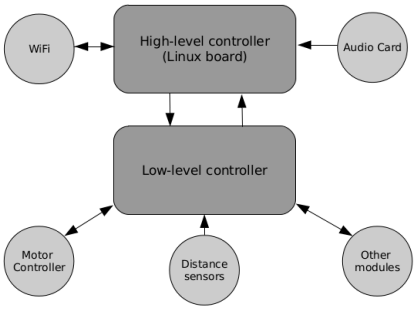

Fig. 2. Hardware configuration of the mobile robot.

- *High-level controller.* This is the main module in which all high-level tasks, such as prediction processes and audio pre-processing, can be run. It synchronizes the output and input signals from the other modules in order to match the sensory data with the actions executed by the robot. This module is also connected to a desktop computer through WiFi in order to save the collected data for further processing (in the experiments presented here, data and predictions have been only processed offline). The controller is based on an odroid[1] single-board computer equipped with a 1.7 GHz ARM Quad-Core processor and 2 GB of RAM.
- *Low-level controller.* The low-level controller interacts directly with the sensors serving as a bridge between the high-level controller and them. In other words, it is in charge of managing different communication buses and protocols (spi, i2c, uart, parallel) of the modules and of streaming the data to the high-level controller. Additionally, it can also run control tasks, in the case a high-level controller is not needed. The low-level controller uses an ARM microcontroller running a real time operating system called FreeRTOS.
- *Audio module.* This module is in charge of the audio capturing. It consists of a USB mini audio card and a microphone, which are connected to the high-level controller.
- *Motor controller.* It contains the electronic circuits needed to drive the DC motors. An Atmega AVR 8-bit microcontroller is used to control the motor drivers and to receive the signals from the magnetic encoders.
- *Distance sensors.* Seven infrared distance sensors (Sharp GP2D120) are connected to the low level controller. They can detect objects in a range from four to thirty centimeters. In the experiments presented here, only three of these sensors were used.

2) Software: The main board (high-level controller) runs an Arch Linux distribution [2]. The software development was made under the Robot Operating System[3] (ROS) which is a collection of software frameworks for robots. Additionaly, Fast Artificial Neural Network Library[4] (Fann2) and Essentia[5] libraries were used for internal models design and audio processing, respectively. The low-level controller runs FreeRTOS[6], a real time operating system for embedded devices that allows the microcontroller to run multiple threads sharing the processor resources.

B. Ego-noise representation

We represented the ego-noise generated by the robot motors by using Mel-frequency cepstral coefficients (MFCCs), which are inspired by human auditory perception. Roughly, they represent the envelope of the quantized spectrum of the audio signal. MFCCs are features widely used for speech recognition [8] as well as for music classification. In this work, MFCCs are obtained by performing the following steps:

- Frame the audio signal (single-channel, sampling rate: 44.1 kHz) into 50ms frames using a Hamming window.
- Calculate the discrete Fourier transform of the frame and obtain the power of the spectrum.
- Apply the Mel filterbank to the power of the spectrum. We have used a filterbank characterised by 40 triangular filters and a frequency range from 30Hz to 16 kHz.
- Compute the logarithm of the 40 filterbank energies.
- Compute the discrete cosine transform of the resulting data, discard the first coefficient and keep the next 20 coefficients as the MFCC features.

[1] http://www.hardkernel.com/main/main.php

[2] https://www.archlinux.org/

[3] http://www.ros.org/

[4] http://leenissen.dk/fann/wp/

[5] http://essentia.upf.edu/

[6] http://www.freertos.org/

C. Experimental Setup

We carried out two experiments. First, we present a classification experiment in which we show how the auditory predictions provided by a set of trained forward models can be used for determining the velocity profile a sensorimotor input belongs to. Next, we show how the auditory predictions provided by a forward model trained with data gathered using a self-exploration behaviour, namely random motor babbling, can be used to detect changes in the external environment, for example a change in the inclination of the surface where the robot is moving on.

In both experiments, the forward models were implemented with a Multi-Layer Perceptron (MLP), shown in Figure 3 using the following structure:
Inputs:

- *Speed Sensors (8 variables):* represents the current speed of each wheel of the robot. Each speed is encoded by the pulse counts generated by the magnetic encoders, ranging between 0 and 120 pulses per time window. At time t, a sensory state $S(t)$ is encoded by two variables. In the experiments presented here, the forward models were fed with four sensory states as inputs: $S(t)$, $S(t-1)$, $S(t-2)$ and $S(t-3)$, that is with a vector of 8 elements [7].
- *Motor Command (8 variables):* represents the motor command applied to each wheel (can be set to ten different speeds: 0-9). At time t, a motor command $M(t)$ is thus encoded by two variables. As for the sensory state, the forward models were fed with four motor commands as inputs, that is with a vector of 8 elements.

Outputs:

- *Auditory response (20 variables).* The resulting auditory consequence is represented as a vector of 20 Mel-frequency Cepstral Coefficients at the time instant $t+1$.

The results reported here are using networks with 30 hidden nodes for the first experiment and 40 hidden nodes for the second as these were the ones performing better. The networks were trained using resilient backpropagation.

Fig. 3. An illustration of the implemented forward model.

1) Ego-noise classification: In this experiment, we trained eleven forward models, using sensorimotor data gathered from the execution of robot movements characterised by eleven different velocity profiles. Figure 4 shows the different profiles that describe the pattern of commands sent every 55 milliseconds to the robot motors. Both motors received the same commands but in the opposite direction, resulting in the robot turning around itself[8]. For each of the velocity profiles, 5000 sensorimotor samples were gathered (recorded in 275 seconds). From each of these datasets, 3000 samples were used for training, 1000 samples for testing and 1000 samples for validating the model. Another set of 10000 samples per velocity profile was recorded for carrying out the classification experiment.

Figure 5 illustrates the classification process. It consisted in feeding the eleven internal models with the data samples of each velocity profile and obtaining the prediction errors by calculating the Euclidean distance between the predicted and the observed auditory outcomes (MFCCs). The input sample was classified as belonging to the model that produced the smallest prediction error.

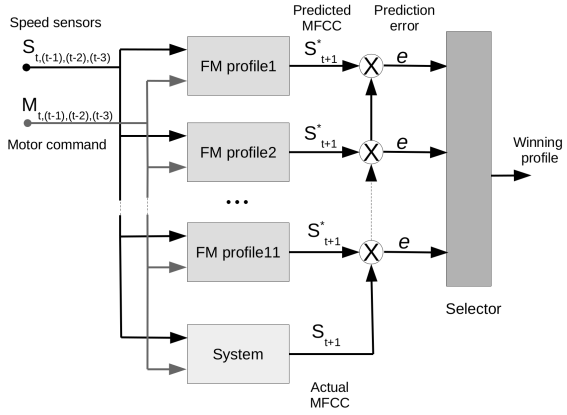

Fig. 5. Diagram of the classification process.

2) Motor babbling and ego-noise prediction: In this experiment, a forward model was trained using sensorimotor data gathered by executing a self-exploration behaviour on the robot, namely random motor babbling. The robot explored its motor capabilities in a rectangular arena (1.8m x 1m) made of a flat wooden floor and cardboard walls (see Figure **??**).

The behaviour was implemented by executing movements characterised by two random parameters: the motor command (speed ranging from 0 to 9) and its duration (number of time steps the same command is sent, ranging from 1 to 5). The same motor command was applied to both motors, so that the robot could only move forward. For avoiding collision with the walls, the robot was programmed to turn whenever the distance sensors detected an obstacle closer than a given a threshold. The sensorimotor data collected while applying the collision avoidance behaviour was discarded. We gathered 40000 samples, 24000 of which were used for training, 8000 for testing and 8000 for validation of the model.

The forward model trained with this data was tested in two environments: the original setup, where the conditions of the arena matched those of the training setup (Figure **??**), and

[7]The aim of using 4 previous time steps is to provide the model with a short memory on the characteristics of the input data. Using the amount of data points that would capture the whole profile characteristics would amount to a different approach worth investigating.

[8]The robot body had four contact points with the ground: two wheels and two castor balls, used to keep the inclination of the robot horizontal. To have the models categorize the noise coming from the motors we placed a sheet of paper on the table which cancels the sounds of the castor wheels.

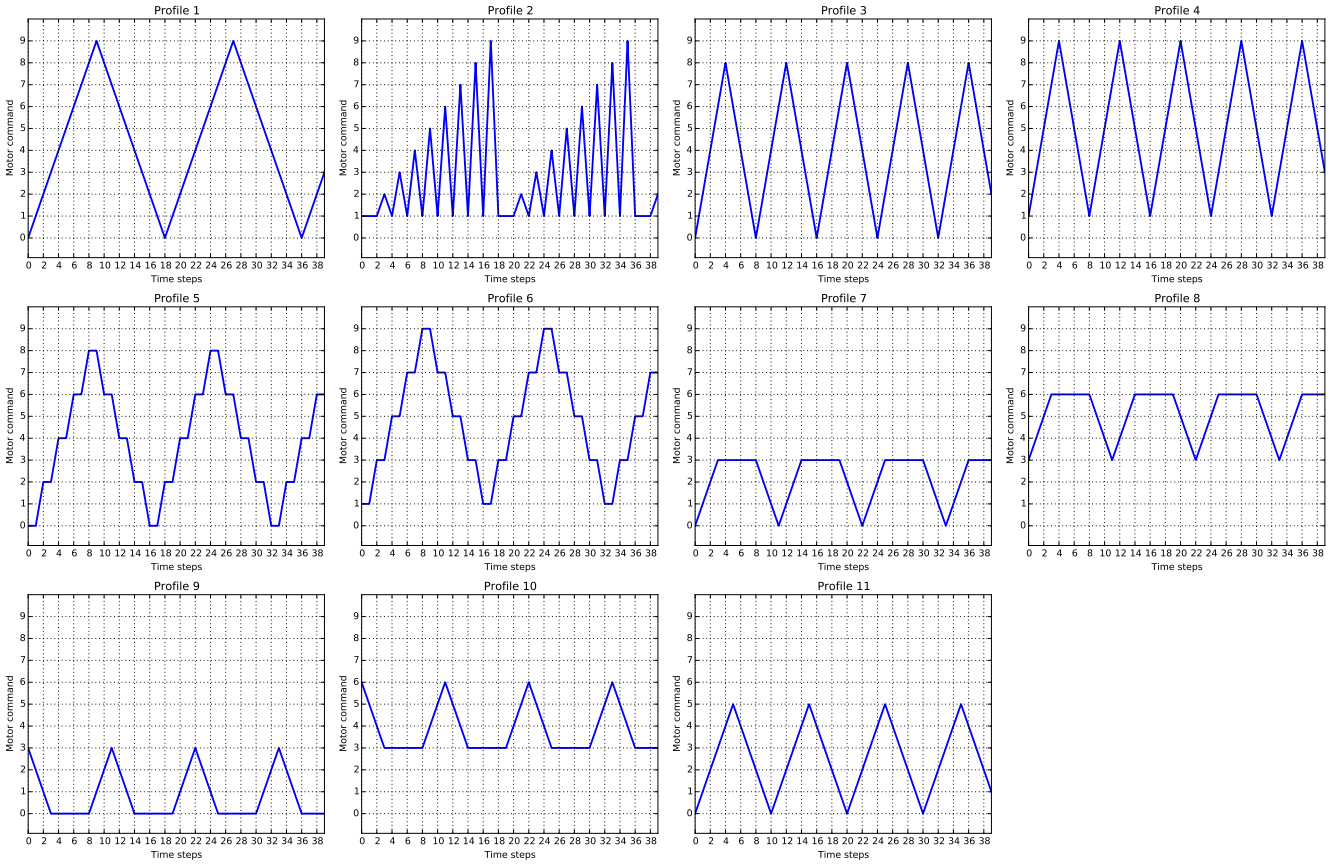

Fig. 4. Plots of the eleven velocity profiles used to train the forward models.

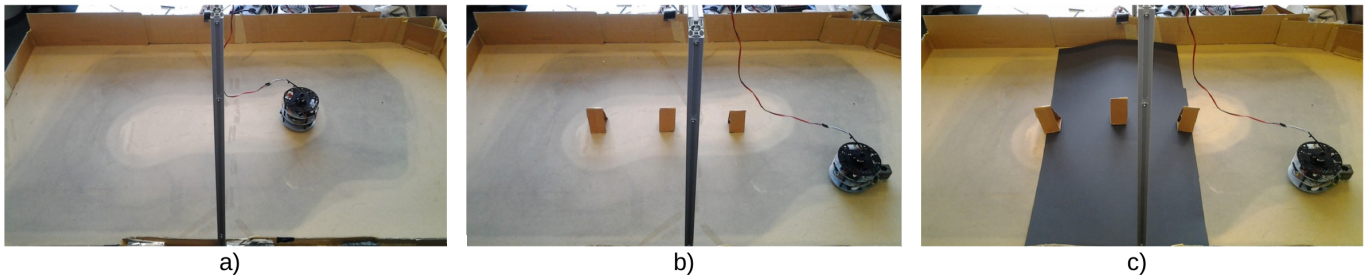

Fig. 6. a) arena used for running the self-exploration behaviour. b) arena with three markers. c) arena with a ramp and three markers. The upward and downward slopes have an inclination of ca. 16.7 degrees. The horizontal surface at the top of the slope has a length of 8 cm.

a modified one, where a ramp was placed in the middle of the arena (see Figure 6). During the test, the robot moved from one side of the arena to the opposite at a constant speed, while calculating prediction errors using the model trained as described above. We repeated the test with four different speeds (6, 7, 8 and 9) and, for each speed, in both the original (flat) and the modified (ramp) environments. Each of these runs was repeated ten times.

In order to detect whenever the robot moved over the ramp, we placed three markers: one at the beginning, one on the top and one at the end of the ramp (see Figure 6). The markers were placed at the same locations also in the flat environment[9].

III. Results

A. *Ego-noise classification*

Table I reports the results of the classification experiment in the form of a confusion matrix. For each profile, we used a test sensorimotor trajectory consisting of 10000 samples. Each sample stored the proprioceptive and motor inputs to be fed to the MLP, and the auditory output for the comparison to the one predicted by the model, as described in Section II-C1 and as

[9]A distance sensor placed at the right side of the robot was used for detecting the locations of the markers and for simplifying the segmentation of the sensorimotor streams for further analysis.

		Predicted (%)										
		Profile 1	Profile 2	Profile 3	Profile 4	Profile 5	Profile 6	Profile 7	Profile 8	Profile 9	Profile 10	Profile 11
Actual class	Profile 1	**62.01**	0.48	0.36	0.39	4.71	13.45	7.21	1.42	0.79	1.81	7.36
	Profile 2	0.35	**97.47**	0.04	0.30	0.43	0.19	0.08	0.41	0.20	0.21	0.32
	Profile 3	0.00	0.33	**92.18**	1.63	0.48	0.34	0.26	2.70	0.27	1.80	0.01
	Profile 4	0.97	0.61	0.75	**90.52**	0.73	1.02	0.59	3.58	0.07	1.10	0.06
	Profile 5	3.22	0.57	0.70	1.33	**87.19**	2.33	0.39	1.81	0.53	1.11	0.82
	Profile 6	3.43	0.90	0.35	0.93	0.50	**91.16**	0.37	1.28	0.03	0.73	0.32
	Profile 7	3.73	0.83	0.00	0.23	0.90	0.38	**77.73**	0.01	0.77	9.48	5.94
	Profile 8	1.87	0.16	1.18	0.62	0.54	0.93	0.01	**89.76**	0.02	4.89	0.02
	Profile 9	0.42	0.60	0.38	0.44	0.99	0.15	4.63	0.00	**92.20**	0.02	0.17
	Profile 10	1.36	0.49	1.32	0.08	0.32	1.06	13.32	10.10	0.03	**71.85**	0.06
	Profile 11	20.85	0.63	0.35	0.33	2.85	1.53	16.40	2.36	1.33	3.95	**49.41**

TABLE I
CONFUSION MATRIX SHOWING THE CLASSIFICATION PERFORMANCE

illustrated in Figure 3. For each of the eleven test trajectories, we fed the corresponding proprioceptive and motor inputs to the eleven trained forward models. This resulted in 10000 auditory predictions per class. Each auditory prediction was compared to the ego-noise observation at the corresponding time step, as described in Figure 5. A prediction error was, thus, calculated by subtracting the predicted ego-noise from the observed one, per each sample. The input sample was therefore classified as belonging to the model (or velocity profile) that produced the smallest prediction error.

As depicted in Table I, five models (Profiles: 2, 3, 4, 6 and 9) correctly classified auditory observations for more than 90% of the times; two models (Profiles: 5 and 8) did it for more than 85% of of the times; two models (Profiles: 7 and 10) did it for more than 70% of the times. The classification performance was worse for Profiles 1 (62.01% correct classification rate) and 11 (49.41% correct classification rate). Moreover, Profile 11 was classified in 20.85% of the times as Profile 1. We believe that this is due to the fact that the two profiles, as illustrated in Figure 4, look too similar to each other: the first half of one period of Profile 11 is, in fact, exactly the same as the first 5 time steps of Profile 1; similarly, the second half of one period of Profile 11 is the same as the last 5 time steps of Profile 1. Profile 11 has been classified as Profile 7 in 16.40% of the times. As it can be seen from Figure 4, the beginning and the end of one period of the two profiles are very close to each other. Moreover, Profile 1 has been classified 4.71% of the times as Profile 5 and 13.45% of the times as Profile 6, probably due to the similar shape of the three trajectories. Similarities can be found also between Profiles 7 and 10, whose classification performance did not overcome 80%. Profile 7 has been classified in 9.48% of the times as Profile 10. Profile 10 has been classified in 13.32% of the times as Profile 7. In one period, in fact, the two profiles have the same sensory and motor input from time step 3 to 8.

Overall, apart from the cases commented above which are probably due to similarities in the velocity profiles, the results of the classification are satisfactory. This demonstrates that the forward models properly encode the mappings between motor commands and the auditory consequences of these movements.

B. Motor babbling and ego-noise prediction

The aim of the second experiment was to demonstrate that the predictive capabilities provided by forward models can be used for detecting unexpected changes, either in the robot embodiment or in the external environment. As described in Section II-C2, in this experiment we did not pre-code motor primitives as in the previous one, rather we ran a random motor babbling behaviour for autonomous exploration (see Section II-C2 for details). The forward model trained with the resulting data was used to calculate ego-noise predictions. As described in the previous section, we executed different tests, where the robot behaviour was characterised by four constant velocities executed in two different environments. We executed 10 runs for each of these tests. Each recorded trajectory has been segmented into 5 parts (see figure 7) using the locations of the markers detected from the distance sensor of the robot. Figure 7 shows the statistics of the prediction errors for each type. In particular, each box plot shows the averages and other statistics of the prediction errors in each of the 5 segments of a trajectory. As illustrated by the plots and as confirmed by a t-test analysis we carried out, there was a statistically significant difference between the two conditions (robot moving on a flat surface and on a ramp) for all the four velocities. For example, when the robot was moving forward with velocity 6 (top plots in Figure 7), there was a significant difference in the prediction error from the moment when the robot started moving on the ramp (see the difference between the *flat* and *ramp* box plots from Segment B to Segment E). The same outcome has been observed when applying different speeds (7, 8, 9). Interestingly, for all the speeds tested, the prediction errors in the last segment of the trajectory (E) under the ramp condition were higher than under the flat condition, probably due to the inertia of the robot after descending the ramp.

These results demonstrate that, through the ego-noise simulation processes provided by the forward model, the robot can detect changes in the environment, such as inclinations of the surface it is moving on.

IV. CONCLUSIONS

This work addressed one of the most unexplored topics in developmental robotics: learning robot ego-noise, that is the

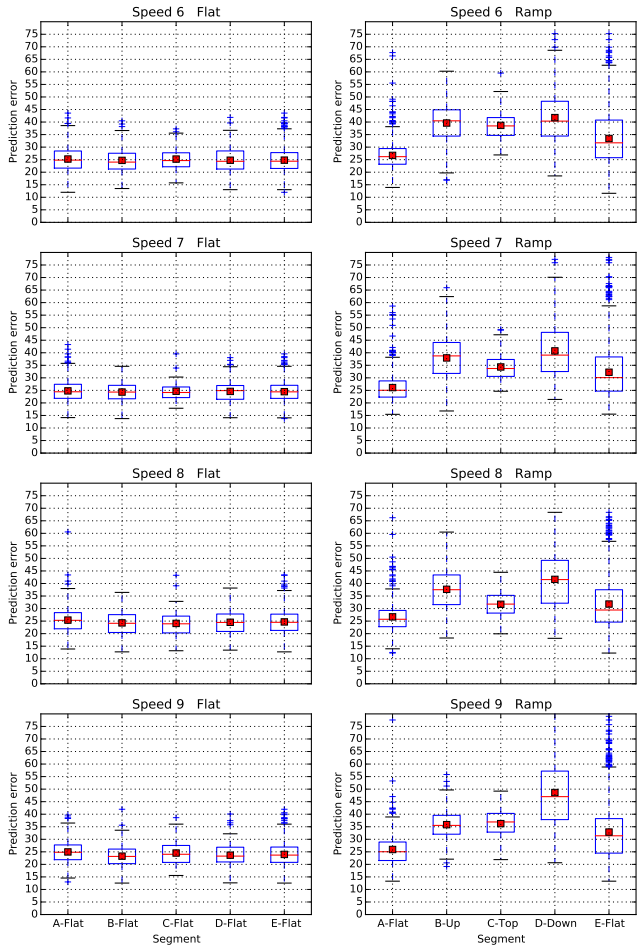

Fig. 7. Box plots showing the statistics of the prediction errors of the second experiment.

		Segment				
		A	***B***	***C***	***D***	***E***
Speed 6	***T-stat***	-4.39	-35.55	-18.87	-23.01	-32.85
	P-value	$< .05$	$< .05$	$< .05$	$< .05$	$< .05$
	Cohen's d	0.26	2.2	2.62	2.16	1.07
Speed 7	***T-stat***	-3.9	-29.51	-14.16	-18.63	-27.95
	P-value	$< .05$	$< .05$	$< .05$	$< .05$	$< .05$
	Cohen's d	0.25	2.02	2.05	1.95	1
Speed 8	***T-stat***	-3.25	-25.14	-8.69	-17.58	-23.05
	P-value	$< .05$	$< .05$	$< .05$	$< .05$	$< .05$
	Cohen's d	0.22	1.95	1.44	1.89	0.89
Speed 9	***T-stat***	-2.34	-26.83	-12.54	-22.37	-24.6
	P-value	$< .05$	$< .05$	$< .05$	$< .05$	$< .05$
	Cohen's d	0.17	2.37	2.16	2.51	1.02

TABLE II
RESULTS OF THE T-TESTS.

noise that a robot produces while moving around. We presented a mechanism for learning and predicting ego-noise on a custom robotic platform. In particular, we adopted forward models as a computational tool for encoding the dynamics of the motor system of the robot and the effect of self-produced movements on the perceived ego-noise. We showed how the predictive capabilities provided by forward models can be used for anticipating the noise produced by intended movements. This capability has been demonstrated in two experimental setups: a classification experiment, where the system classified robot behaviours based on comparison between the produced noise and the noise that would be produced by intended actions; an experiment where we showed how a robot knowledgeable of its ego-noise can detect unexpected environmental conditions, such as changes in the inclination of the surface it is moving on.

We believe that the approach is scalable to more complex behaviours. At this level, these models can be seen as the unit of analysis. Further, more complex behaviors (e.g trajectories) can use these predictions as a base which we intend to investigate further. We have investigated this scaling up in other work [9]. It is worth noting that the current work aimed at investigating the possibilities and potential use of MFCC as a tool for coding noise and the use of forward models for this type of prediction. The work is inspired by the MOSAIC architecture which also uses pairs of internal models. However, we want to investigate further approaches where each model is coding multiple behaviours and producing the corresponding sensory predictions.

ACKNOWLEDGMENT

The research leading to these results has partially received funding from the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 609465 (EARS (Embodied Audition for RobotS) Project) as well as from the Mexican National Science and Technology Council (CONACyT) and the German Service of Academic Exchange (DAAD).

REFERENCES

[1] H. G. Okuno, K. Nakadai, and H.-D. Kim, *Robotics Research: The 14th International Symposium ISRR*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ch. Robot Audition: Missing Feature Theory Approach and Active Audition, pp. 227–244.

[2] G. Ince, K. Nakadai, T. Rodemann, Y. Hasegawa, H. Tsujino, and J. Imura, "Ego noise suppression of a robot using template subtraction," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2009)*, Oct 2009, pp. 199–204.

[3] D. M. Wolpert, Z. Ghahramani, and J. R. Flanagan, "Perspectives and problems in motor learning," *Trends in Cognitive Sciences*, vol. 5, no. 11, pp. 487 – 494, 2001.

[4] G. Schillaci, V. V. Hafner, and B. Lara, "Exploration behaviours, body representations and simulations processes for the development of cognition in artificial agents." *Frontiers in Robotics and AI*, vol. 3, no. 39, 2016.

[5] G. Schillaci, C. N. Ritter, V. V. Hafner, and B. Lara, "Body representations for robot ego-noise modelling and prediction. towards the development of a sense of agency in artificial agents," *Proc. of the 15th Int. Conf. on the Synthesis and Simulation of Living Systems (ALIFE XV). Cancun. Mexico*, pp. 390–397, 2016.

[6] P. Rochat, "Self-perception and action in infancy," *Experimental brain research*, vol. 123, no. 1-2, pp. 102–109, 1998.

[7] S. Zoia, L. Blason, G. D'Ottavio, M. Bulgheroni, E. Pezzetta, A. Scabar, and U. Castiello, "Evidence of early development of action planning in the human foetus: a kinematic study," *Experimental Brain Research*, vol. 176, no. 2, pp. 217–226, 2007.

[8] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques," *Journal of Computing*, vol. 2, no. 3, pp. 138–143, 2010.

[9] E. Escobar Juárez, G. Schillaci, J. Hermosillo Valadez, and B. Lara Guzman, "A self-organized internal models architecture for coding sensory-motor schemes," *Frontiers in Robotics and AI*, vol. 3, no. 22, 2016.