Autonomous Sensorimotor Learning for Sound Source Localization by a Humanoid Robot

Quan V. Nguyen[1], Laurent Girin[1,2], Gérard Bailly[1], Frédéric Elisei[1], and Duc Canh Nguyen[1]

***Abstract*— We consider the problem of learning to localize a speech source using a humanoid robot equipped with a binaural hearing system. We aim to map binaural audio features into the relative angle between the robot's head direction and the target source direction based on a sensorimotor training framework. To this end, we make the following contributions: (i) a procedure to automatically collect and label audio and motor data for sensorimotor training; (ii) the use of a convolutional neural network (CNN) trained with white noise signal and ground truth relative source direction. Experimental evaluation with speech signals shows that the CNN can localize the speech source even without an explicit algorithm for dealing with missing spectral features.**

I. Introduction

For a social robot dedicated to human-robot interaction, the ability to perceive and analyze auditory scenes is an important function. This includes sound source localization (SSL) which consists in inferring the position of the emitting source(s) relatively to the sensors based on acoustic characteristics of the perceived sounds. SSL is used to direct the robot's attention to a target source. It is also related to further processing such as source separation or speech recognition.

For humanoid robots equipped with a binaural hearing system, the two main acoustic cues for SSL are the (frequency-wise) interaural level difference (ILD) and interaural phase difference (IPD) between the signals arriving at the two ears [1], [2]. To map such features into source location, some methods are based on (the inversion of) a physical model of sound propagation from source to sensors [3], [4], [5].[3] A major problem for these methods is that binaural features depend not only on the source location but also on the geometry of the robot and its sensors, *e.g.* the head related transfer functions (HRTF), and on the surrounding environment, *e.g.* room reverberation. Most of these techniques limit to 1D localization, *i.e.* azimuth estimation.

To overcome the above limitations, machine-learning-based supervised SSL methods have been proposed. The main idea of those methods is to learn a statistical mapping –i.e. a regression– from acoustic features to source location using controlled training data with known source position. Basically, the mapping models are either derived from generative models (typically density mixture models) or consist of artificial deep neural networks (DNNs). As for SSL based on generative models, one can cite [3], [7], [8]. A good example that we reuse in the present study is the probabilistic regression called Gaussian locally-linear mapping (GLLiM) that was proposed in [9] and applied to SSL in [10], [11] using audio-visual training data. Approaches using DNNs include [12], [13], [14], [15], [16].

Both approaches require a large amount of training data for the model to correctly capture the complex relationship between source location and acoustic features in real-world scenarios. Yet, real-world SSL training data are difficult to collect in a large amount. For example, in [11], data are collected by manually moving a loudspeaker within the field-of-view of the robot camera. Because of this difficulty, SSL techniques are often trained and tested with simulated data, which may be oversimplistic compared to real-world data, even if good room impulse response (RIR) simulators exist. Therefore, a method to automatically collect a large amount of training data for robust SSL is highly desirable.

The contributions of this paper are twofold. (i) First, we propose, implemented and tested a protocol to make a humanoid robot –in the present case the iCub robot– automatically record and label data for training and testing SSL. We developed a device, a loudspeaker embedded in a marker cardboard fixated at the end of a rod. With this device fixed on its arm, the robot can autonomously move the loudspeaker around its body, record audio data and visually detect the ground truth source location so that it can later explore and learn the resulting auditory-motor relations. (ii) Second, we build and train a DNN, here a convolutional neural network (CNN), to estimate from binaural features the motor commands that will drive the robot head toward the target speech source. Following [11], the CNN model is trained with white noise data and then evaluated with speech data. GLLiM is used as a baseline. Surprisingly, even without an explicit algorithm for dealing with missing features in the speech spectral data, as is the case for GLLiM, the proposed CNN model shows competitive performance.

The remainder of the paper is organized as follows. Section II presents the protocol for the iCub to automatically collect and label data for SSL training and testing. The CNN architecture and the input/output data used for our SSL experiments are described in Section III. Speech source localization results are presented in Section III-D. Section IV concludes the paper.

[1]Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-lab, 38000 Grenoble, France * Institute of Engineering Univ. Grenoble Alpes

[2]INRIA Grenoble Rhône-Alpes, Montbonnot, France

firstname.lastname@gipsa-lab.grenoble-inp.fr

[3]In the present study we mostly consider single-source localization. Localization of multiple (simultaneously emitting) sources usually requires to rely on the so-called source sparsity in the time-frequency domain and to group the measured acoustic features into source clusters [6].

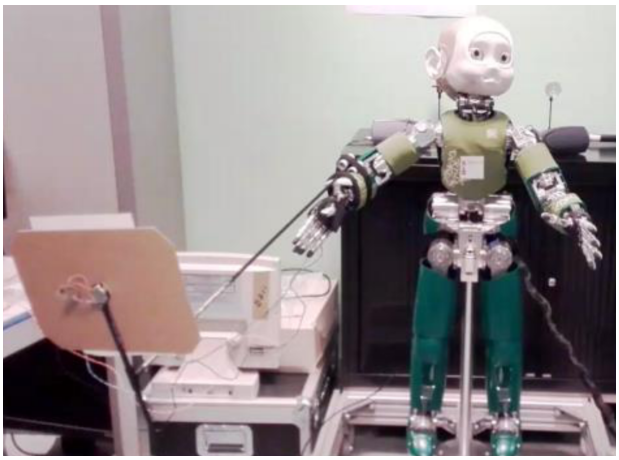

Fig. 1. The iCub humanoid robot with the recording device fixed on its right arm. The device consists of a rod ended with a loudspeaker embedded into a visual marker for video tracking.

II. Automatic data collecting and labeling for SSL

In this section, we describe the protocol we designed and implemented to make the iCub robot able to collect and label audio and source location data automatically.

A. Recording device

Before recording, we attached a rod to the right arm of the iCub robot and mounted a (light) loudspeaker at the end of the rod, as illustrated in Fig. 1. The rod was about 70 cm long. The loudspeaker was placed at the center of a visual marker so that the robot can detect the loudspeaker location using the cameras placed in its eyes and a video tracking system. The visual tracker uses the Aruco Library, which is fast and accurate [17], [18]. Note that an Aruco pattern with dark areas in the center was used as the marker, so that the dark-colored loudspeaker does not impede the marker detection. This visual tracking provides us the ground truth of the source location, as detailed in the next subsection.

Note that, independently of the present study, our iCub robot is specifically equipped for robot audition and SSL. Its ears have high-quality microphones (Soundman OKMII) and 3D printed pinnae obtained from a human mould (see Fig. 1). This increases the impact of the sound source orientation on the spectrum of the recorded sound, i.e. a less flat HTRF.

B. Recording procedure

We now explain the general principle/strategy for the data recording (quantitative details are given later).

First, the robot moves the rod/loudspeaker to a given position in the space in front of him by changing its arm configuration. Second, the robot uses the visual tracking to direct its head towards the center of the visual marker, i.e. the loudspeaker. More precisely, the iCub's head has three degrees of freedom which are pan, yaw, and tilt rotations. In this work we fixed the yaw angle to zero, and only vary the pan angle α and the tilt angle β. The robot's eyes being in neutral position (the robot looks straight in front of him), the robot moves his head so that the visual marker tied to the loudspeaker and detected by the visual tracker is being placed at the center of the image recorded by the robot. In short, the robot now looks straight to the loudspeaker. The source ground-truth location is thus defined by the resulting pan and tilt angles, denoted α_{target} and β_{target}, respectively. These values are stored.

Then the robot moves its head on a grid of head orientations (see quantitative details below). For each head orientation $\{\alpha_{\text{current}}, \beta_{\text{current}}\}$ it proceeds to the following binaural sound recording. The loudspeaker emits a 1-s white-noise signal, pauses for 1 s, and then emits a variable-length utterance randomly selected from the TIMIT dataset [19]. After finishing these two-item recordings (one with white-noise signal and one with speech signal) at one head orientation, it moves to the next head orientation on the sampling grid to perform the next recording.

Once the recordings are made for all the head orientations on the grid, the robot moves his arm to put the loudspeaker in another position, then it repeats the above steps. This is done for a large set of loudspeaker positions and head orientations (see below).

C. Resulting dataset

The robot proceeded to the above procedure for a total of 56 source positions corresponding to 56 right arm configurations, occupying a large span on a quasi-sphere in front of the robot. These source positions are distributed evenly on the quasi-sphere. The far-left or far-right source positions are at approximately $\pm 45^\circ$ from the "neutral" head orientation (when the robot looks straight ahead). The upper and lower source positions are at approximately $\pm 30^\circ$. The source-to-head distance is about 1.2 m.

For each source position, the robot varies its head orientation within a grid of 21 pan angles ranging from -40° to 40° with 4°-step, and 7 tilt angles ranging from -20° to 14°: $\beta = \{-20^\circ, -14^\circ, -8^\circ, -2^\circ, 3^\circ, 9^\circ, 14^\circ\}$.

At the end of the recording session, we have a total of $N = 8{,}232$ binaural recordings for white noise signal and the same number of recordings for speech signal (along with the corresponding $8{,}232$ uplets $\{\alpha_{\text{current}}, \beta_{\text{current}}, \alpha_{\text{target}}, \beta_{\text{target}}\}$). These stereo signals were recorded at $44,1$ kHz and then downsampled to 16 kHz for SSL processing. The audio dataset sums up to $7{,}600$ Mbytes.

III. SSL experiments with automatically recorded data

In this section, we describe the SSL experiments we conducted with the above-described data. We first present the input and output of the mapping, then we present the CNN architecture, its training/testing, and the SSL results.

A. Input: Binaural feature vector

From the two-channel audio recordings, we extract binaural feature vectors which contain information about the source direction. These vectors are obtained by first applying the short-time Fourier transform (STFT) to the 16 kHz

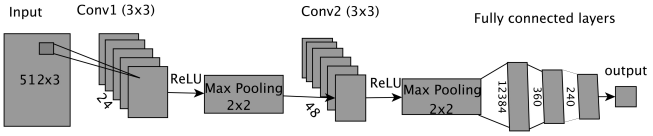

Fig. 2. Proposed CNN architecture for SSL.

microphone signals with a 64-ms Hann window and 56-ms overlap, yielding $T = 125$ windows per second. Each time window indexed by t contains 1,024 samples, leading to $F = 512$ positive-frequency complex Fourier coefficients s^{l}_{ft} and s^{r}_{ft} (for the left and right channel respectively), covering 0 Hz–8 kHz. We consider two binaural cues $\boldsymbol{\gamma} = \{\gamma_{ft}\}_{f=1,t=1}^{F,T}$ and $\boldsymbol{\phi} = \{\phi_{ft}\}_{f=1,t=1}^{F,T}$, where:

$$\begin{aligned} \gamma_{ft} &= 20 \log |s^{\mathrm{r}}_{ft}/s^{\mathrm{l}}_{ft}| \in \mathbb{R}, \\ \phi_{ft} &= \exp(j \arg(s^{\mathrm{r}}_{ft}/s^{\mathrm{l}}_{ft})) \in \mathbb{C} \equiv \mathbb{R}^2. \end{aligned} \tag{1}$$

$\boldsymbol{\gamma}$ is the classical ILD spectrogram, and $\boldsymbol{\phi}$ is the complex exponential representation of the IPD spectrogram, chosen in order to avoid problems related to phase circularity. Each input data $\mathbf{x}$ to the CNN is an $F \times 3$ vector obtained by the concatenation of $\boldsymbol{\gamma}$ and the real and imaginary parts of $\boldsymbol{\phi}$, and temporal averaging over the sound duration:

$$\mathbf{x} = \frac{1}{T} \sum_{t=1}^{T} [\boldsymbol{\gamma}_{:t}; \mathrm{real}(\boldsymbol{\phi}_{:t}); \mathrm{imag}(\boldsymbol{\phi}_{:t})] \in \mathbb{R}^{F \times 3}. \tag{2}$$

B. Output: Relative source-to-head orientation vector

Binaural features are generally assumed to mostly depend on the relative position of the source w.r.t. the sensors. Indeed, if the source moves in one direction and the robot head moves accordingly, i.e. follows the source, the effect of the head will remain the same. Binaural feature variation will only be produced by the change w.r.t. to robot torso and room reverberation. Therefore, the output vector is defined as the angle difference between the target head orientation and the current head orientation of the robot, denoted $\mathbf{y} = [\Delta_\alpha = \alpha_{\mathrm{target}} - \alpha_{\mathrm{current}}, \Delta_\beta = \beta_{\mathrm{target}} - \beta_{\mathrm{current}}]$. At training time all angles are available. At test time, the mapping provides the estimated angle difference vector $\widehat{\mathbf{y}} = [\widehat{\Delta}_\alpha, \widehat{\Delta}_\beta]$, the motor control unit provides the current head orientation, and the estimated target (source) direction is given by: $[\widehat{\alpha}_{\mathrm{target}} = \alpha_{\mathrm{current}} + \widehat{\Delta}_\alpha, \widehat{\beta}_{\mathrm{target}} = \beta_{\mathrm{current}} + \widehat{\Delta}_\beta]$. Used as a motor command, this output vector will guide the robot head to look straight to the sound source.

C. Network architecture and training

The architecture of our CNN is depicted in Fig. 2. It has two convolution layers and three fully connected layers. After each convolution layer, we apply a Rectified Linear Unit (ReLU) activation function and max pooling operation. We apply batch normalization after the two convolution layers. For the fully connected layers, we use two hidden layers and one output layer. Each sample input is of size $F \times 3$. We use the white noise + source direction data for training (and the speech + source direction data for testing, see below). The complete training (and testing) set is thus a tensor of size $8{,}232 \times F \times 3$. For this kind of input, we use 2D convolution for the two convolution layers. In the first convolution layer, we use a 2D convolution with 3 input channels, 24 output channels, and 3×3 convolution. The max pooling operation uses a 2×2 window. In the second convolution layer, the 2D convolution has 24 input channels, 48 output channels, and 3×3 convolution. Here also, the max pooling operation uses a 2×2 window. In the fully connected layers, we have 12,384 neurons in the first fully connected layer, 360 neurons in the first hidden layer, 240 neurons in the second hidden layer, and 2 neurons in the output layer corresponding to Δ_α and Δ_β. The learning rate during training is set to 0.001. We use Mean Square Error for the loss function (MSEloss) and Adam optimizer for optimization [20]. The training batch size is 3 and training finishes after 77 epochs.

D. Evaluation and results

The CNN was tested with the 8,232 recorded speech signals. We compare this model with the GLLiM algorithm [9], [10] used as a baseline method on the same data. As briefly stated in the introduction, GLLiM explicitly considers the missing data in speech spectral vectors by adding a time-frequency domain binaural mask (activity matrix) as an additional input. To have the best performance of the GLLiM model, we used 32 Gaussian components in the mixture.

We computed the root mean squared error (RMSE) between the estimated target angle values $\hat{\alpha}_{\mathrm{target}}$ (resp. $\hat{\beta}_{\mathrm{target}}$) and their corresponding ground truth values α_{target} (resp. β_{target}) in degree. With the CNN model, the RMSE for $\hat{\alpha}_{\mathrm{target}}$ and $\hat{\beta}_{\mathrm{target}}$, averaged over all source locations and all head orientations, both equal to $5.6°$. Such an error of the order of $5°$ is consistent with the performances reported in the SSL literature, hence the data that were automatically recorded by the robot are exploitable for SSL. For GLLiM, the RMSE is $3.4°$ for $\hat{\alpha}_{\mathrm{target}}$ and $7.5°$ for $\hat{\beta}_{\mathrm{target}}$. Therefore, the CNN has a slightly lower average accuracy in pan estimation and a slightly higher accuracy in tilt estimation, compared with GLLiM.

To better characterize the difference between the CNN and GLLiM, we computed the RMSE for $\hat{\alpha}_{\mathrm{target}}$ and $\hat{\beta}_{\mathrm{target}}$, still averaged over all source directions, but for each robot head orientation $\{\alpha_{\mathrm{current}}, \beta_{\mathrm{current}}\}$. The results are plotted as "heatmaps" in Fig. 3. For the source pan angle $\hat{\alpha}_{\mathrm{target}}$, we can see that the RMSE is distributed as a function of the robot head pan and tilt angles in a slightly smoother manner (and with lower values) for GLLiM compared to the CNN. In contrast, for the source tilt angle $\hat{\beta}_{\mathrm{target}}$, we can see that the RMSE obtained with GLLiM becomes large (up to $17°$) at the extrema of head orientation, especially for the lower tilt angles, i.e. when the robot is looking down, whereas the RMSE is more regularly distributed (and with lower values) for the CNN model.

IV. Conclusions

We presented a protocol to automatically collect and label SSL training/testing data with a humanoid robot, here an

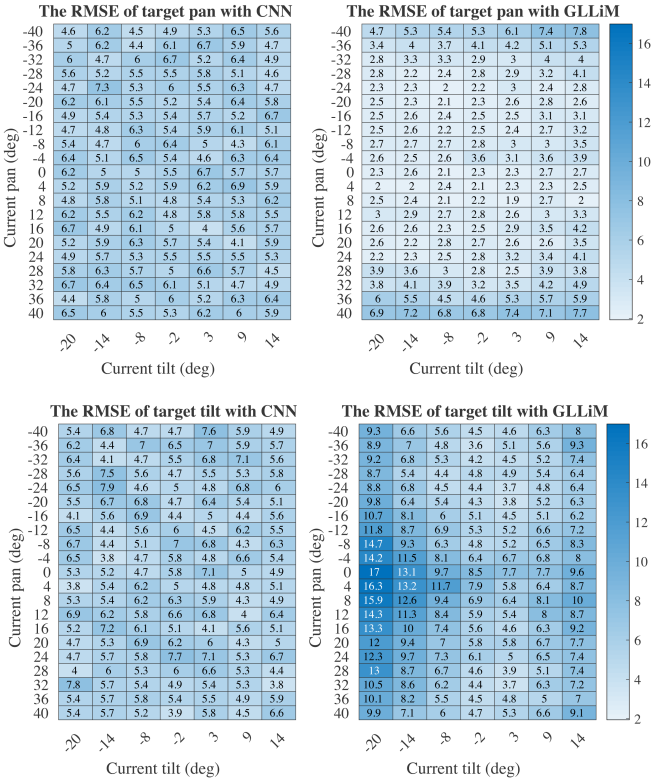

Fig. 3. RMSE of target pan (top) and tilt (bottom) angles (in degree), averaged over all source locations, as a function of the robot head orientations, using the CNN model (left) and using the GLLiM model (right).

iCub robot. A CNN was build and trained to map binaural features into the relative source-to-robot head angles. The average SSL accuracy obtained with the CNN in both pan and tilt is $5.6°$. Compared to the baseline method GLLiM, the CNN model has a lower average performance for pan estimation and a better average performance for tilt estimation, and an overall more consistent performance across different robot head orientations. However, the keypoint here is not the difference between CNN and GLLiM, but rather the fact that **the robot can automatically proceed to the complete chain going from collecting multimodal data to learning from these data.** The resulting mapping model can be used in Human-robot interaction to make the robot turn its head and look to its interlocutor.

In future works, we will increase the recorded dataset by varying the distance from the source to the robot head, e.g. using a telescopic rod. We will also extend the range of source-to-head angles using both left and right arms and more arm configurations. As for the CNN, we will test if adding a time-frequency mask information characterizing source activity as an additional input, as done with GLLiM can improve the CNN performance. We may also extend the model for localizing multiple sound sources.

ACKNOWLEDGMENT

The author would like to thank Grégoire Mugnier for his preliminary work on the robot grabbing system, and Xiaofei Li for fruitful discussion on SSL.

REFERENCES

[1] M. Raspaud, H. Viste, and G. Evangelista, "Binaural source localization by joint estimation of ild and itd," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, pp. 68–77, 2010.

[2] K. Youssef, S. Argentieri, and J.-L. Zarader, "Towards a systematic study of binaural cues," in *IEEE IROS*, 2012, pp. 1004–1009.

[3] V. Willert, J. Eggert, J. Adamy, R. Stahl, and E. Korner, "A probabilistic model for binaural sound localization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 5, pp. 982–994, 2006.

[4] F. Keyrouz, "Advanced binaural sound localization in 3-D for humanoid robots," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 9, pp. 2098–2107, 2014.

[5] X. Li, L. Girin, R. Horaud, and S. Gannot, "Estimation of the direct-path relative transfer function for supervised sound-source localization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2171–2186, 2016.

[6] ——, "Multiple-speaker localization based on direct-path features and likelihood maximization with spatial sparsity regularization," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 10, pp. 1997–2012, 2017.

[7] J. Hornstein, M. Lopes, J. Santos-Victor, and F. Lacerda, "Sound localization for humanoid robots-building audio-motor maps based on the hrtf," in *IEEE IROS*, 2006, pp. 1170–1176.

[8] T. May, S. Van de Par, and A. Kohlrausch, "A probabilistic model for robust localization based on a binaural auditory front-end," *IEEE Transactions on audio, speech, and language processing*, vol. 19, no. 1, pp. 1–13, 2011.

[9] A. Deleforge, F. Forbes, and R. Horaud, "High-dimensional regression with Gaussian mixtures and partially-latent response variables," *Statistics and Computing*, vol. 25, no. 5, pp. 893–911, 2015.

[10] A. Deleforge, V. Drouard, L. Girin, and R. Horaud, "Mapping sounds on images using binaural spectrograms," in *European Signal Processing Conference (EUSIPCO)*, 2014, pp. 2470–2474.

[11] A. Deleforge, R. Horaud, Y. Y. Schechner, and L. Girin, "Co-localization of audio sources in images using binaural features and locally-linear regression," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 4, pp. 718–731, Apr. 2015.

[12] M. S. Datum, F. Palmieri, and A. Moiseff, "An artificial neural network for sound localization using binaural cues," *The Journal of the Acoustical Society of America*, vol. 100, no. 1, pp. 372–383, 1996.

[13] L. Wang, T.-K. Hon, J. D. Reiss, and A. Cavallaro, "An iterative approach to source counting and localization using two distant microphones," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 6, pp. 1079–1093, 2016.

[14] R. Takeda and K. Komatani, "Unsupervised adaptation of deep neural networks for sound source localization using entropy minimization," in *IEEE ICASSP*, 2017, pp. 2217–2221.

[15] S. Adavanne, A. Politis, and T. Virtanen, "Direction of arrival estimation for multiple sound sources using convolutional recurrent neural network," *arXiv preprint:1710.10059*, 2017.

[16] N. Ma, T. May, and G. J. Brown, "Exploiting deep neural networks and head movements for robust binaural localization of multiple sources in reverberant environments," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 12, pp. 2444–2453, 2017.

[17] S. Garrido-Jurado, R. Munoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014.

[18] F. J. Romero-Ramirez, R. Munoz-Salinas, and R. Medina-Carnicer, "Speeded up detection of squared fiducial markers," *Image and Vision Computing*, vol. 76, pp. 38 – 47, 2018.

[19] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "TIMIT acoustic-phonetic continous speech corpus LDC93S1," *Linguistic Data Consortium*, 1993.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.