2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Self-Supervised Incremental Learning for Sound Source Localization in Complex Indoor Environment

Hangxin Liu[1⋆] Zeyu Zhang[1⋆] Yixin Zhu[1,2] Song-Chun Zhu[1,2]

***Abstract*— This paper presents an incremental learning framework for mobile robots localizing the human sound source using a microphone array in a complex indoor environment consisting of multiple rooms. In contrast to conventional approaches that leverage direction-of-arrival (DOA) estimation, the framework allows a robot to accumulate training data and improve the performance of the prediction model over time using an incremental learning scheme. Specifically, we use implicit acoustic features obtained from an auto-encoder together with the geometry features from the map for training. A self-supervision process is developed such that the model ranks the priority of rooms to explore and assigns the ground truth label to the collected data, updating the learned model on-the-fly. The framework does not require pre-collected data and can be directly applied to real-world scenarios without any human supervisions or interventions. In experiments, we demonstrate that the prediction accuracy reaches 67% using about 20 training samples and eventually achieves 90% accuracy within 120 samples, surpassing prior classification-based methods with explicit GCC-PHAT features.**

I. Introduction

The Sound Source Localization (SSL) problem in robotics [1], [2] tackles the issue of obtaining the position of the sound source by determining its direction and distance using audio signals. Typical setup involves using a microphone array [3] or binaural microphones [4] to collect multi-channel acoustic signals for calculating direction-of-arrival (DOA) or spectral cues from the raw audio signals. Such information is further processed to estimate the sound source position.

However, the majority of the field in SSL is currently restricted to localizing sound source inside a single room [3], [4], [5], [6], [7], or in simple non-line-of-sight (NLOS) scenarios, *i.e.*, behind a corner [8], [9], or blocked by objects [10]. Such setup is insufficient for a domestic robot or service robot to react rapidly from users across multiple rooms, hindering the practical uses of SSL in large-scale.

Take a typical multi-room setup (see Figure 1) as an example, where the mobile robot (highlighted with black bounding box) stations in the hallway. The explicit acoustic features (*e.g.*, time-difference-of-arrival (TDOA) or inter-microphone intensity difference (IID)) are incapable of providing adequate information, especially for the non-field-of-view (NFOV) region in the far distance, *e.g.*, the user (highlighted with a blue skeleton) in one of the three rooms. Moreover, the acoustic signal is polluted due to the high noise-to-signal ratio, reverberation, *etc*. As a result, any explicit features extracted from the polluted signals become deficient in such large-scale, unstructured, and noisy setup, demanding more modern approaches to incorporate the features of both the sound source and the environment.

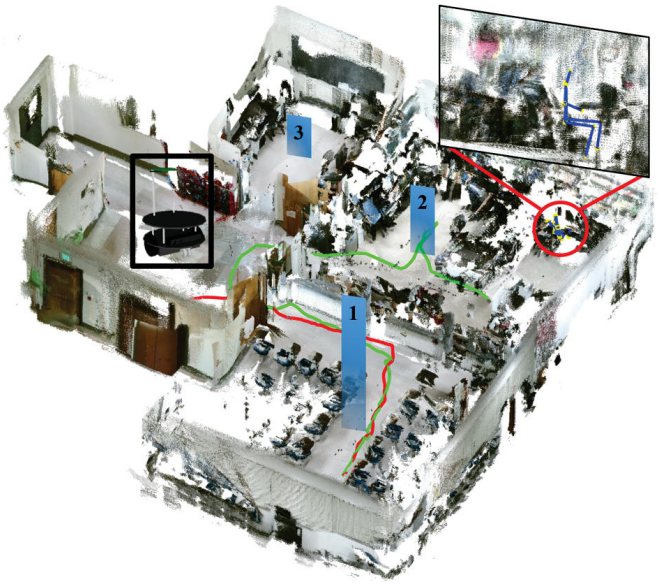


Fig. 1: A typical indoor environment consisting of multiple rooms. Given a verbal command from a user, the proposed incremental learning framework ranks the priority of the rooms to be explored, indicated by the height of the blue bars. In this example, the robot initially explores the wrong room following the red path, which serves as a negative sample. Following the ranking order, it continues to explore the second room with the green path. A detection of the user leads to a positive labeled sample of the training data. All the positive and negative data is labeled on-the-fly to adapt to new users in unknown complex indoor environments, and is accumulated to refine the current model to improve future prediction accuracy.

The recent advancements of Deep Neural Networks (DNNs) [11] allow machine learning methods reach a remarkable level in some specific tasks, even arguably better than human, *e.g.*, control [12], [13], grasping [14], [15], object recognition [16], [17], learning from demonstration [18]. It is proven to be an effective way to extract implicit features that are robust against noises and interference. Although DNNs-based methods have been applied to SSL problems and demonstrated decent performance [19], [20], [21], [22], [7], prior methods suffer from two major issues that prevent them from being applied in larger scale: (i) difficult to collect a vast amount of training data, and (ii) too cumbersome to adapt the trained model to recognize the acoustic signals from untrained sources in unknown indoor environments. Such drawbacks result in poor performance, prohibiting the practical uses in complex, large-scale indoor environments.

To handle these difficulties, we propose a three-step self-supervised incremental learning framework for mobile robot's SSL in indoor environments, summarized in Figure 2:

⋆ H. Liu, Z. Zhang contributed equally to this work.
[1] UCLA Center for Vision, Cognition, Learning, and Autonomy (VCLA) at Statistics Department. Emails: {hx.liu, zeyuzhang, yixin.zhu}@ucla.edu, sczhu@stat.ucla.edu.
[2] International Center for AI and Robot Autonomy (CARA)

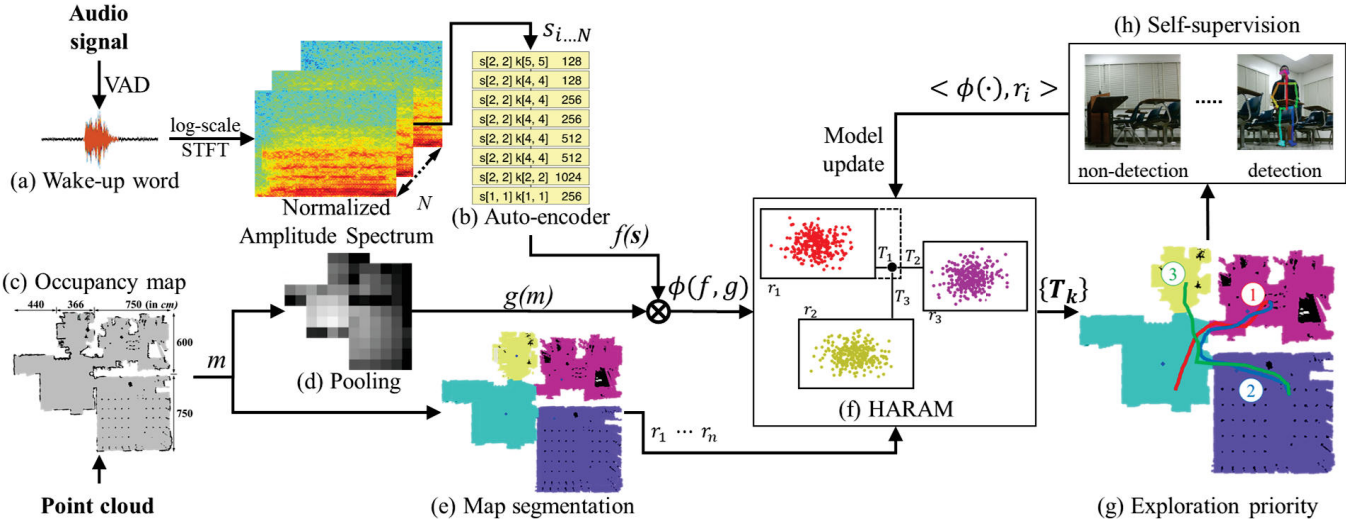


Fig. 2: The proposed approach using a self-supervised incremental learning scheme. (a) The multi-channel signals from the user's wake-up word are picked up by VAD. Each signal is transferred to the amplitude spectrum and normalized to $[0, 1]$, from which (b) an auto-encoder is trained to extract implicit features. Each block represents a 2D convolution with stride s[·, ·], kernel size k[·, ·] and the number of channels. In addition, (c) an occupancy map obtained from the reconstructed point cloud is down-sampled by pooling (d). (b)(d) together form the feature for learning. (e) Individual rooms are segmented from the point cloud. (f) The HARAM model is adopted to predict the priority rank of rooms the robot should visit. (g) The robot self-supervises the learning by exploring the rooms. (h) The exploration will be labeled as the positive sample if the robot detects the user, which will update the HARAM model incrementally.

1) **Localization model**. We apply a room segmentation algorithm to obtain candidate regions (*e.g.*, rooms) from an occupancy map. A prediction model ranks the regions by the likelihood of location of the sound source (*e.g.*, labels of the rooms) from high to low.
2) **Incremental learning**. In contrast to batch learning methods, we use an incremental learning scheme that allows the system to accumulate the training data over time and refine the prediction model once a new sample arrives. Hence, no pre-collected data is required.
3) **Self-supervised data labeling via active exploration**. We design a self-supervised process to label each new sample received on-the-fly. Specifically, the robot explores each room following the predicted ranking order. The room will be labeled negative if no sound source detected; otherwise positive.

In summary, we argue that the proposed method is by far the closest setting to real-world scenarios compared to the prior work. Such a method can be directly applied to indoor mobile robots equipped with acoustic sensors (*e.g.*, a microphone array) for SSL, alleviating the needs of human supervisions or intervention after the deployment.

A. Related Work

In the field of **SSL**, prior work mainly adopts a wide range of signal processing methods [1], [2], which usually calculate the DOA and perform a distance estimation to localize the sound source. Some typical algorithms include beamforming, Generalized Cross-Correlation with Phase Transform (GCC-PHAT), Multiple Signal Classification (MUSIC), *etc*. Masking is also applied to improve the performance [23], [24]. They are, however, limited to the single room scenario.

SSL for the NFOV target has been attempted. For instance, [8], [25] incorporates optical and acoustic observations to enhance the estimation of the sound source using a pre-built acoustic observation database. Leveraging environment geometry cues and the DOA, [9] combines diffraction and reflection directions to localize the target around a corner in an anechoic chamber. Similarly, [10] tracks a moving sound source in an open room using direct and reflection acoustic rays, where the NFOV was created by a wall. However, these methods would have difficulties in multi-room setups with untrained sound sources inside unknown environments.

DNNs are widely used in natural language processing and speech recognition [26], which are orthogonal to the SSL problem. There are recent efforts in using DNNs for SSL, only limited to estimating the DOA [19], [20], [21], [22], [7]. They are also limited to the single room scenario, requiring training data of sound sources inside every new environment.

Active sensing that changes acoustic sensors' configuration has also been studied in SSL. Using the binaural microphone with pinnae setting, the platform can change its pinnae configuration [27] actively based on the data received to improve performance. However, it lacks the capability of exploration in the environment. By contrast, mobile robots with sound source mapping actively navigate in large space to localize sound sources [5], [6]. [28] also utilized a mobile robot to collect ground truth acoustic data. However, they do not leverage the observed new data to improve the model.

B. Contribution

This paper makes the following three contributions:

1) We introduce an incremental learning scheme for SSL in the indoor setting with multiple rooms that allows the system to accumulate the training data on-the-fly.
2) We incorporate a self-supervision method that combines with the robot's active exploration. Once a sample is received, the robot will explore the rooms based on the rank, thereby labeling the sample in a self-supervised fashion according to the detection of the sound source.
3) We provide a Robot Operating System (ROS) package that integrates all modules of the proposed framework, including the acoustic signal processing, room segmentation, and the learning and inference, which allows a robot to perform SSL task without any human supervisions or interventions. The code will be made publicly available.

C. Overview

The remainder of the paper is organized as follows. Section II describes how to extract both acoustic signal features and environmental geometry features from the received raw signals. Based on the extracted features, Section III details the localization model adopted to predict sound sources in a multi-room setup. In Section IV, we explain how the robot explores multiple rooms in a self-supervised fashion using an incremental learning scheme. Section V showcases the experiment results. We discuss the results and conclude the paper in Section VI.

II. Feature Extraction

This section introduces the feature extraction process. The features consist of both the acoustic features based on the collected signals from the microphone array and the geometry feature extracted from the SLAM results, which encode both the geometry structure of the environment and the robot's current position.

A. Acoustic Features

Voice Activity Detection (VAD): To recognize the acoustic signal from the human's wake-up word and distill the data of interest out of the background noise, we utilize the state-of-the-art Google WebRTC VAD [29] with a frame size 20 ms in duration. Figure 2a shows one example of the detected segment, consisting of multi-channel acoustic signals. Each channel of the detected audio segment is transformed to its amplitude spectrum using the log-scale Short-Time Fourier Transform (STFT) with an FFT size 1024, normalized to $[0,1]$. Figure 3a shows one example of the normalized spectrum with a dimension 255×255.

Signal Low-dimensional Embedding: The dimensions of the normalized spectrum are still large, and the data contains certain levels of noises. To address these issues, we use an auto-encoder to extract a low-dimensional embedding from the spectrum per channel. Figure 2b depicts the encoder structure that contains multiple convolutional layers; each layer is followed by a Leaky-ReLU activation layer and the batch normalization. The decoder is symmetrical to the encoder. Such structure results in a 256-dimensional embedding by minimizing the weighted Mean Square Error (MSE) between the original spectrum and the reconstructed spectrum by the decoder:

$$\mathcal{L}(\theta;\mathbf{s}) = \frac{1}{N}\sum_{i=1}^{N}\ell(s_i,\psi(s_i;\theta)), \tag{1}$$

where s_i denotes the ith original amplitude spectrum, $\psi(s_i;\theta)$ the corresponding reconstructed spectrum, and ℓ the weighted MSE between the two spectrum, where the weights decrease from 10 to 1 linearly as the frequency increase from 0Hz to 6000Hz and above [30]. Such embedding contains implicit features of one spectrum, denoted as $f(s_i)$. Figure 3b shows an example of the reconstructed spectrum with the reconstruction error shown in Figure 3c.

There are three advantages using such an auto-encoder method to encode the acoustic signals: (i) The dimension reduction process reduces the noise contained in the raw signal, such as the background noise and reverberation. (ii) Reducing the dimension shrinks the memory required in the proposed incremental learning framework. (iii) Since the auto-encoder is designed to minimize reconstruction loss, the encoding process still preserves meaningful information in the signal as implicit features.

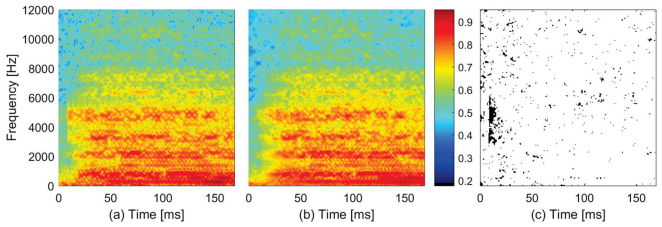


Fig. 3: (a) The original spectrum normalized to $[0,1]$. (b) The reconstructed spectrum using an auto-encoder. (c) The reconstruction error as a binary image, in which the black pixel indicates the relative error larger than 5%.

B. Environment Geometry Features

To obtain richer environmental information through SLAM, we use a Kinect v2 sensor to construct the 3D structure of the environment using RTAB-Map [31]. Figure 1 depicts the reconstructed environment in the form of the registered point cloud, which can be easily converted to a 2D occupancy map. We apply a pooling strategy to down-sample the occupancy map to reduce its dimension. A diffusion is applied based on the robot's position. Figure 2c shows the original map (m) that spans about $15.5m\times13.5m$ is compressed to a 12×11 matrix ($g(m)$), where each element is normalized to $[0,1]$: the closer the element to the robot's current position, the whiter the element is in Figure 2d.

The resulting matrix obtained from map pooling is flatted and concatenated to the embedding vector of the acoustic signals. The resulting vectors, $\boldsymbol{\phi}=[f(s),g(m)]$, accommodate the features extracted from both the acoustic signal and the environment geometry. The produced feature vectors are used for the later incremental learning process.

III. Localization Model by Ranking

We adopt the Hierarchical Adaptive Resonance Associative Map (HARAM) algorithm [32], [33] to rank individual room where the sound source could potentially be from. HARAM is a neural architecture, able to real-time perform supervised learning of pattern pairs (*i.e.*, given a feature vector and the ground truth of the room label) in an incremental manner. The rest of this section briefly describes the HARAM model under our SSL setting; we refer readers to the original papers [32], [33] for in-depth details.

Learning: Formally, given the concatenated features $\boldsymbol{\phi}$ and the list of candidate rooms $\boldsymbol{r}$, the input vector $\boldsymbol{\Phi}$ of HARAM is a $2M$-dimensional vector $\boldsymbol{\Phi}=(\boldsymbol{\phi},\ \boldsymbol{\phi}^c)$, where M is the dimension of the feature $\boldsymbol{\phi}$. The candidate vector $\mathcal{R}$ is a $2R$-dimensional vector $\mathcal{R}=(\boldsymbol{r},\boldsymbol{r}^c)$, where R is the number of rooms in the environment. Complement coding ϕ_i^c and r_i^c are defined as $\phi_i^c\equiv1-\phi_i$ and $r_i^c\equiv1-r_i$, representing both on-responses and off-responses of the input vector. The weight vectors $\boldsymbol{\omega}_k^{\phi}$ and $\boldsymbol{\omega}_k^{r}$, $k=1,\ldots,R$ are initialized to unity, and will be updated incrementally during the learning process. Once receiving a feature $\boldsymbol{\phi}$, the neural activation function for each room T_k is calculated as

$$T_k(\boldsymbol{\Phi},\mathcal{R}) = \gamma\frac{|\boldsymbol{\Phi}\wedge\boldsymbol{\omega}_k^{\phi}|}{\alpha_{\phi}+|\boldsymbol{\omega}_k^{\phi}|}+(1-\gamma)\frac{|\mathcal{R}\wedge\boldsymbol{\omega}_k^{r}|}{\alpha_r+|\boldsymbol{\omega}_k^{r}|}, \tag{2}$$

where $\alpha_{\phi}>0$, $\alpha_r>0$, and $\gamma\in[0,1]$ are the learning parameters set by the cross-validation, $\wedge$ is the fuzzy AND operation defined as $(\boldsymbol{p}\wedge\boldsymbol{q})_i\equiv\min(p_i,q_i)$, and the norm $|\cdot|$ is defined as $|\boldsymbol{p}|\equiv\sum_i p_i$. The system will make choices by selecting the neural activation functions with the largest magnitude

$$T_*=\max\{T_k:k=1,\ldots,R\}. \tag{3}$$

Fig. 4: (Top) Examples of the human pose detection. (Bottom) Non-detection examples.

A matching criterion is defined to confirm the choice of T_* or creating a new neural activation function. The parameters ρ_ϕ and ρ_r are user-defined to measure the minimum accepted similarity and the overall model complexity, respectively

$$\frac{|\boldsymbol{\Phi} \wedge \boldsymbol{\omega}_*^{\phi}|}{|\boldsymbol{\Phi}|} \geqslant \rho_\phi, \frac{|\mathcal{R} \wedge \boldsymbol{\omega}_*^{r}|}{|\mathcal{R}|} \geqslant \rho_r. \tag{4}$$

Specifically, if the above inequalities are violated, a new neural activation function is created to include the new sample, and the corresponding T_* is set to 0. If the above criterion is satisfied, the weight vectors are adjusted incrementally during the learning

$$\begin{cases} \boldsymbol{\omega}_*^{\phi(\text{new})} = \lambda_\phi(\boldsymbol{\Phi} \wedge \boldsymbol{\omega}_*^{\phi(\text{old})}) + (1-\lambda_\phi)\boldsymbol{\omega}_*^{\phi(\text{old})} & \text{(5a)} \\ \boldsymbol{\omega}_*^{r(\text{new})} = \lambda_r(\mathcal{R} \wedge \boldsymbol{\omega}_*^{r(\text{old})}) + (1-\lambda_r)\boldsymbol{\omega}_*^{r(\text{old})} & \text{(5b)} \end{cases}$$

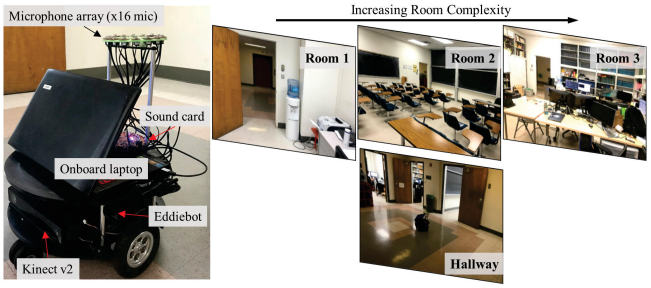
where λ_ϕ and $\lambda_r \in [0,1]$ are the learning rates. Take an example shown in Figure 2f, the hyperbox of cluster r_1 expands (see the dash box) to include the new sample.

Ranking: By sorting $\{T_k\}$ in Equation 3 based on their relative magnitudes, the order of T_k implies the ranking of the candidate rooms based on the current sample received, illustrated in Figure 2f. The hyperbox of each cluster has been constructed based on prior samples. When a new sample (black dot in the center) arrives, the activation function calculates the distance between the received data and each hyperbox. The higher the magnitude. The smaller the distance between the received data and each hyperbox. The smaller the distance is, the higher the priority of a room will be explored with. In this example, since the magnitude $T_1 > T_2 > T_3$ (*i.e.*, $\text{dist}(T_1) < \text{dist}(T_2) < \text{dist}(T_3)$), the robot will explore in the order of room 1, room 2, and room 3.

IV. Self-supervision via Active Exploration

This section describes the self-supervision process built on top of the HARAM algorithm, enabling a mobile robot to acquire the ground truth label of a sample without any human supervisions or interventions.

Room Segmentation: In order to obtain the number of candidate rooms in the environment, the robot is required to segment each room from the entire occupancy map. This step is equivalent to finding the number of labels for the learning process. We utilize the room segmentation algorithm described in [34] (see Figure 2e). Specifically, we use the Distance Transform-based Segmentation: given an 8-bit single channel image obtained from the occupancy map where accessible areas are white and inaccessible black, the algorithm applies different thresholds to merge accessible areas iteratively, and the most valid segments will be chosen.



(a) (b)

Fig. 5: (a) Eddiebot robot setup. A Kinect v2 RGB-D sensor is mounted in the front. A uniform circular microphone array containing 16 microphones is placed on the top. The robot and all the sensors are connected to an on-board laptop that runs the learning algorithm in real-time. (b) A multi-room environment used in experiments. The robot stations in the hallway and the sound sources are in room 1, 2, and 3 with an increasing room complexity.

Exploration: The HARAM model produces the distance (see Equation 2) between the feature of a newly received sample $\boldsymbol{\phi}$ and each of the cluster, *i.e.*, rooms. The lower the distance, the more likely the $\boldsymbol{\phi}$ is from the corresponding room (cluster). Therefore, the rank of rooms is determined by ranking the distance from low to high.

Before the very first sample arrives, the model can only generate uniform predictions. In this case, the robot explores the rooms based on a random guess. After receiving the very first sample, the exploration is based on the ranking described in Section III, and the performance is expected to improve with the increasing number of the sample received.

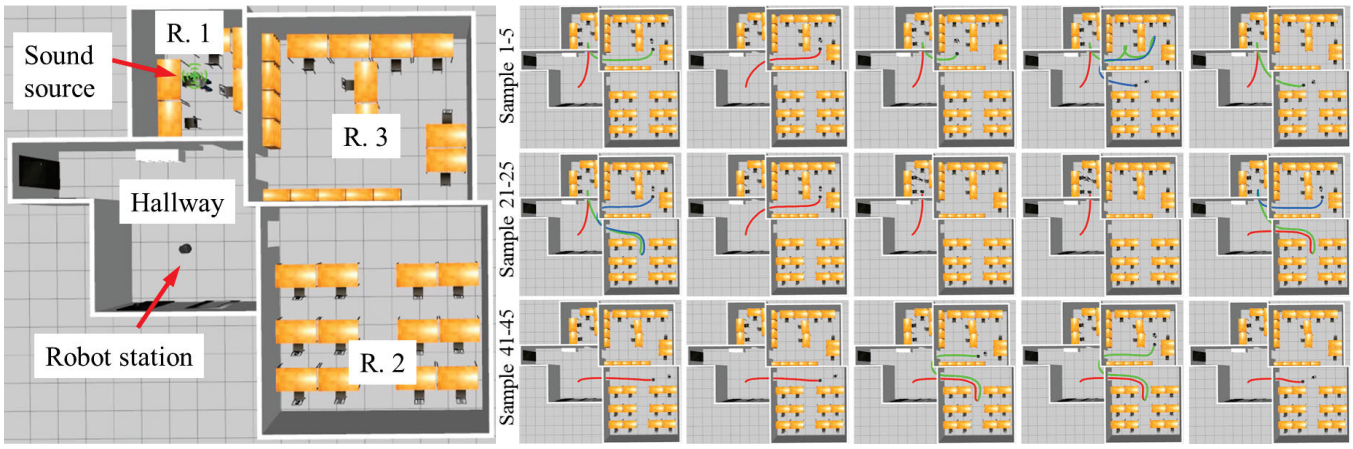
Labeling by Detection: The robot can subsequently navigate to each room following the rank (see Figure 2g) and use its optical sensor to verify the correctness of the prediction. Specifically, we adopt the state-of-the-art human pose detection method, OpenPose [35], to detect the human as the sound source in a room. Figure 4 shows various detection and non-detection examples. Once a successful detection is triggered in a room (note it is not necessarily the room on the top of the rank), a labeled data pair $\langle \boldsymbol{\phi}, \boldsymbol{r}_* \rangle$ is obtained. The model then updates according to Equation 5.

V. Experiment

A. Robot Platform and Experimental Setup

Our system allows *real-time* data acquisition, processing, and learning. We evaluate the proposed method using a system on a Parallax Eddie Robot Platform (see Figure 5a). A uniform circular microphone array with an $18cm$ diameter is equipped, consisting of 16 microphones. The microphones are connected to a sound card with a multi-channel ADC for satisfactory signal synchronization. A Kinect v2 RGB-D sensor is used to capture environmental 3D structure information as well as to detect human poses. The entire system runs online in ROS with an onboard laptop.

We test the proposed method in a physical world with a multi-room setup; see Figure 5b for the corresponding visualization in the simulator. The robot stations in the hallway, and the sound source is located at one of the rooms, which are mostly in the robot's NFOV. Additionally, these rooms have an increasing room complexity, containing various cluttered objects and obstacles. Such setup is especially difficult for acoustic experiments, as the background noise is not negligible, and the reverberation is intractable using prior methods. A total of 155 sound samples are collected



(a) Simulation environment (b) Localization performance improves as the amount of samples increases

Fig. 6: An illustration of the incremental learning process in simulation. The sound source is visualized at the location in the corresponding real world. The robot visits the room subsequently following the rank predicted by the model. The red, green, and blue trajectories indicate the first, second, and third rooms the robot visits. The number of lines depicts the number of trails used to find the sound source location.

in 13 different locations distributed in all rooms. Out of the 155 samples, 35 are randomly selected to train the auto-encoder for acoustic feature extraction. The rest 120 samples are used in the learning and testing process. In a real-world application, the auto-encoder can be pre-trained using general acoustic data.

B. Incremental Learning with Active Exploration

All samples are collected in a physical multi-room setup, and the evaluation is also performed in a physical environment. We further reconstruct and visualize the process in the Gazebo simulator to illustrate the incremental learning process (see Figure 6a): the robot stations in the hallway and the sound source is placed in other rooms according to the locations where the samples are collected. The robot will visit the rooms sequentially following the predicted rank. Once the robot detects the sound source, it labels the sample, updates the HARAM model, and returns to the hallway, waiting for the next sample.

Figure 6b illustrates several keyframes of the incremental learning process. The red, blue, and green trajectory indicate the first, the second and the third room the robot visits, respectively. While the robot can eventually find the locations by visiting all three rooms, we define the evaluation of the performance as the first hit rate and the second hit rate: how

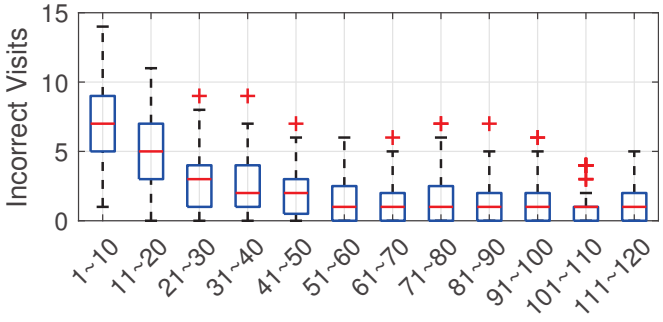


Fig. 7: The number of incorrect visits before finding the correct sound source locations in every 10 samples over 100 trails. The horizontal lines indicate median mistakes, and the bottom and top edges of the blue boxes indicate the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme data points that are not considered outliers, whereas the red cross marks are the outliers. The number of mistakes decreases rapidly.

many times the robot could find the sound sources within one and two visits, respectively. The model performs poorly in the first 5 samples and gradually improves as the number of received samples increases. After about 40 samples, the robot can find the sound source correctly in one visit frequently.

To further validate the robustness of the proposed method, we run 100 repeated trails by feeding the collected samples in random orders to eliminate the randomness in the learning process. Figure 7 shows the boxplot of how many incorrect visits a robot needs to find the correct sound source locations with an increment of 10 samples. After receiving 40 samples, the median number of the incorrect visits before finding the correct sound source locations decreases to < 2 mistakes in every 10 samples. The performance further improves with only 1 mistake per 10 samples after receiving 60 samples. Note that the expectation of the incorrect visits per 10 samples using a random guess is 10. Figure 1 shows a test running in a physical environment, in which the robot finds the user in its second visit. Figure 8 shows another example using only the first visit.

According to a report from *IFTTT* [36], a web-based service with 11 million users, 60% of users use their voice assistance devices more than 4 times a day. Therefore, the performance reported herein indicates that a domestic robot could correctly localize the sound sources across multiple rooms merely based on the wake-up word (four times a day) reliably (< 1 mistake per 10 calls) in two weeks, without any human supervisions or interventions.

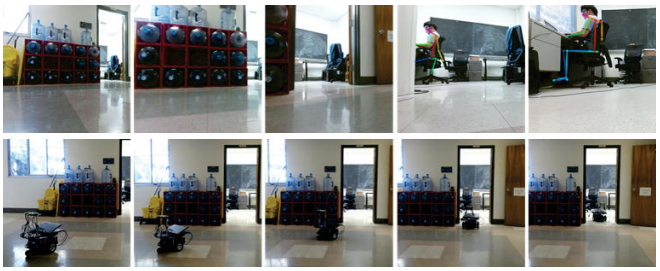


Fig. 8: Testing in a physical environment, in which the robot locates the correct sound source with only one visit. (Top) Key frames from the robot view for navigation with human pose detection. (Bottom) The corresponding third-person views.

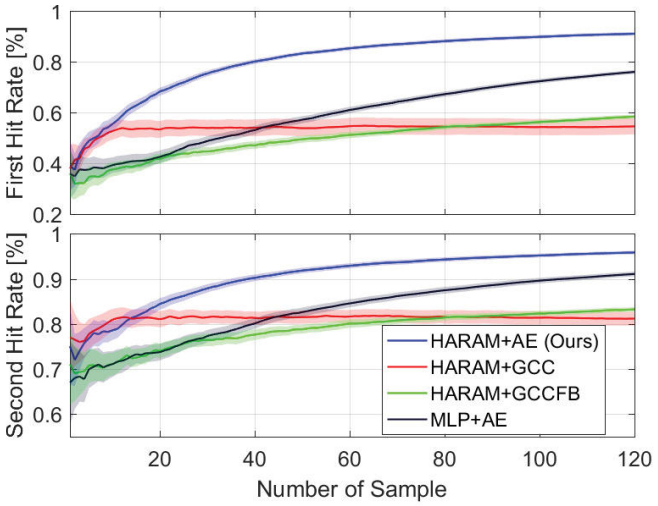


Fig. 9: The mean accuracy of (blue) the proposed method and (green and red) two baselines. The first and the second hit rates indicate the robot finds the correct sound source locations within one and two visits, respectively. The color strips indicate the 95% confidence interval over 100 trails.

C. Comparison

We compared the proposed method with three baselines:

1) **HARAM + GCC**. We combine the HARAM algorithm with GCC-PHAT feature and geometry feature, a popular acoustic feature that can be extracted *explicitly* for SSL.
2) **HARAM + GCCFB**. We add a mel-scale filter bank on top of the GCC-PHAT [7], designed specifically for human voices.
3) **MLP + AE**. We choose an incremental learning version of the classic multi-layer perceptron (MLP) classification method instead of HARAM and learn from the encoded implicit acoustic feature.

Note that some popular machine learning methods, such as SVM, are not comparable in our setting, because they cannot be trained incrementally—a retrain over all samples is required for each new sample arrives.

Figure 9 shows the comparison results. Learning with *explicit* GCC-based features do not lead to satisfactory results as the performance saturates quickly, which validates the conjecture that *explicit* acoustic features underperform in a complex indoor environment. Similarly, MLP combined with the same implicit acoustic feature obtained from the auto-encoder does not perform as well as the one using the HARAM algorithm. The proposed method surpass all three baselines after receiving 15 samples.

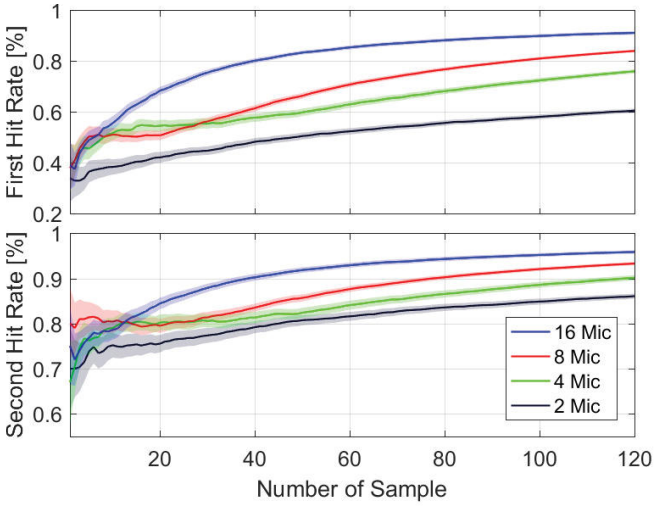


Fig. 10: The mean accuracy of the proposed method using four different microphone array configurations. The color strips indicates the 95% confidence interval over 100 trails.

The performances using different microphone array configurations are also investigated, which profiles the trade-off between the cost of the setup and the performance. By maintaining uniform microphone placements, we compare current 16-microphone setup with 2, 4, and 8-microphone setups (see Figure 10). Overall, more microphones lead to better performance with minor fluctuations in the early stage.

VI. Discussion and Conclusion

A. Discussion

How to allow localization in higher resolution? Typical SSL approaches aim to obtain the exact positions. Although the present setup and results only showcase the resolution at the room level, which is sufficient enough to enable most of the services as a domestic robot, the proposed method could provide localizations in a grid world with higher resolution. However, more samples are likely needed.

Flexibility of the framework. Other popular methods can replace some of the modules in our framework. For example, by treating the received features as states, the rank of a room as actions, and assigning rewards when a correct detection occurs, one can use reinforcement learning to replace HARAM. Other incremental learning models or other features (*e.g.*, GCC-related) can be used; some of which have demonstrated in the baselines.

How to scale up to scenarios with multiple sound sources? Current framework does not distinguish multiple sound sources. To address this issue, we need to incorporate an extra module of voiceprint recognition. However, the overall pipeline is still sufficient to handle such scenarios.

B. Conclusion

This paper has proposed a self-supervised incremental learning method for SSL in a complex indoor environment consisting of multiple rooms. Specifically, the method localizes the human sound source to one of the rooms. We designed an auto-encoder to extracted implicit acoustic features from the signals collected from a uniform circular microphone array with 16 microphones. These features are concatenated with the environment geometry features obtained from pooling the occupancy map of the 3D environment. A HARAM model is adopted to learn the rank of rooms to explore with a probability from high to low. The self-supervision is achieved through robot actively exploring the rooms according to the predicted rank and detecting sound sources by human poses, which improves model performance incrementally. In the experiment, we demonstrate that the proposed method has first and second hit rates of 67% and 84% after 20 samples, and of 90% and 96% after 120 samples, which significantly outperform three baselines.

Acknowledgement: The work reported herein was supported by DARPA XAI grant N66001-17-2-4029, ONR MURI grant N00014-16-1-2007, ARO grant W911NF-18-1-0296, and an NVIDIA GPU donation grant.

REFERENCES

[1] H. G. Okuno and K. Nakadai, "Robot audition: Its rise and perspectives," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[2] C. Rascon and I. Meza, "Localization of sound sources in robotics: A review," *Robotics and Autonomous Systems*, vol. 96, pp. 184–210, 2017.

[3] F. Grondin, D. Létourneau, F. Ferland, V. Rousseau, and F. Michaud, "The manyears open framework," *Autonomous Robots*, vol. 34, no. 3, pp. 217–232, 2013.

[4] H. Liu, C. Pang, and J. Zhang, "Binaural sound source localization based on generalized parametric model and two-layer matching strategy in complex environments," in *International Conference on Robotics and Automation (ICRA)*, 2015.

[5] N. Kallakuri, J. Even, Y. Morales, C. Ishi, and N. Hagita, "Probabilistic approach for building auditory maps with a mobile microphone array," in *International Conference on Robotics and Automation (ICRA)*, 2013.

[6] D. Su, T. Vidal-Calleja, and J. V. Miro, "Towards real-time 3d sound sources mapping with linear microphone arrays," in *International Conference on Robotics and Automation (ICRA)*, 2017.

[7] W. He, P. Motlicek, and J.-M. Odobez, "Deep neural networks for multiple speaker detection and localization," in *International Conference on Robotics and Automation (ICRA)*, 2018.

[8] K. Takami, T. Furukawa, M. Kumon, D. Kimoto, and G. Dissanayake, "Estimation of a nonvisible field-of-view mobile target incorporating optical and acoustic sensors," *Autonomous Robots*, vol. 40, no. 2, pp. 343–359, 2016.

[9] K. Takami, H. Liu, T. Furukawa, M. Kumon, and G. Dissanayake, "Non-field-of-view sound source localization using diffraction and reflection signals," in *International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[10] K. Cho, J. Suh, C. J. Tomlin, and S. Oh, "Reflection-aware sound source localization," in *International Conference on Robotics and Automation (ICRA)*, 2018.

[11] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[12] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *International Conference on Machine Learning (ICML)*, 2016.

[13] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[14] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *arXiv preprint arXiv:1703.09312*, 2017.

[15] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[17] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning (ICML)*, 2015.

[18] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[19] Y. Yu, W. Wang, and P. Han, "Localization based stereo speech source separation using probabilistic time-frequency masking and deep neural networks," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2016, no. 1, p. 7, 2016.

[20] R. Takeda and K. Komatani, "Sound source localization based on deep neural networks with directional activate function exploiting phase information," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[21] N. Yalta, K. Nakadai, and T. Ogata, "Sound source localization using deep learning models," *Journal of Robotics and Mechatronics*, vol. 29, no. 1, pp. 37–48, 2017.

[22] P. Pertilä and E. Cakir, "Robust direction estimation with convolutional neural networks based steered response power," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[23] F. Grondin and F. Michaud, "Time difference of arrival estimation based on binary frequency mask for sound source localization on mobile robots," in *International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[24] F. Grondin and F. Michaud, "Noise mask for tdoa sound source localization of speech on mobile robots in noisy environments," in *International Conference on Robotics and Automation (ICRA)*, 2016.

[25] K. Takami, T. Furukawa, M. Kumon, and G. Dissanayake, "Non-field-of-view acoustic target estimation in complex indoor environment," in *Field and Service Robotics*, 2016.

[26] L. Deng, D. Yu, *et al.*, "Deep learning: methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[27] W. Odo, D. Kimoto, M. Kumon, and T. Furukawa, "Active sound source localization by pinnae with recursive bayesian estimation," *Journal of Robotics and Mechatronics*, vol. 29, no. 1, pp. 49–58, 2017.

[28] J. Le Roux, E. Vincent, J. R. Hershey, and D. P. Ellis, "Micbots: collecting large realistic datasets for speech and audio research using mobile robots," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2015.

[29] "Google webrtc." Accessed: 2018-08-15.

[30] J. Engel, C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan, "Neural audio synthesis of musical notes with wavenet autoencoders," in *International Conference on Machine Learning (ICML)*, 2017.

[31] M. Labbe and F. Michaud, "Online global loop closure detection for large-scale multi-session graph-based slam," in *International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[32] A.-H. Tan, "Adaptive resonance associative map," *Neural Networks*, vol. 8, no. 3, pp. 437–446, 1995.

[33] F. Benites and E. Sapozhnikova, "Haram: A hierarchical aram neural network for large-scale text classification," in *Data Mining Workshop (ICDMW), 2015 IEEE International Conference on*, pp. 847–854, 2015.

[34] R. Bormann, F. Jordan, W. Li, J. Hampp, and M. Hägele, "Room segmentation: Survey, implementation, and analysis," in *International Conference on Robotics and Automation (ICRA)*, 2016.

[35] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[36] "2017 voice assistant trends." Accessed: 2018-09-05.