UNIVERSITY OF CALIFORNIA

Los Angeles

Sound Source Localization in Complex Indoor Environment:
A Self-Supervised Incremental Learning Approach

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Computer Science

by

Zeyu Zhang

2019

ABSTRACT OF THE THESIS

Sound Source Localization in Complex Indoor Environment:
A Self-Supervised Incremental Learning Approach

by

Zeyu Zhang
Master of Science in Computer Science
University of California, Los Angeles, 2019
Professor Song-Chun Zhu, Chair

Sound source localization is essential in robotics, which broadens the possibilities of human-robot interactions by enriching the robot's perceptual capabilities. Localizing an acoustic source in a complex indoor environment is especially challenging due to the high noise-to-signal ratio and reverberations. In this thesis, we present 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. The framework allows robots to accumulate training data and improve the performance of the prediction model over time using an incremental learning scheme. A self-supervision process is developed such that the model ranks the priority of rooms to explore, assigns the ground truth label to the collected data, and updates the learned model on-the-fly. In experiments, we demonstrate that the framework can be directly deployed in real-world scenarios without extra human interventions, and can localize the sound source successfully.

The thesis of Zeyu Zhang is approved.

Yizhou Sun

Eliezer Gafni

Song-Chun Zhu, Committee Chair

University of California, Los Angeles

2019

To my parents

For their endless love, support, and encouragement

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

This thesis consists of a conference paper, that is, ”Self-Supervised Incremental Learning for Sound Source Localization in Complex Indoor Environment” which was accepted at the 2019 IEEE International Conference on Robotics and Automation (ICRA) [LZZ19]. The list of coauthors is Hangxin Liu, Zeyu Zhang, Yixin Zhu, and Song-Chun Zhu. This work 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.

I am grateful to my advisor, Professor Song-Chun Zhu, for offering me the opportunity to join the Center for Vision, Cognition, Learning, and Autonomy, so that I have a chance to explore the cutting-edge of technologies in Artificial Intelligence. I would also like to thank Hangxin Liu and Yixin Zhu of the UCLA Computer Science Department for their generous guiding and mentoring.

Finally, I would like to express my sincere gratitude towards my parents for their endless love, support, and encouragement throughout my life, and give me the strength to chase my dream.

CHAPTER 1

Introduction

Sound Source Localization (SSL) is essential in robotics, and it broadens the possibilities of human-robot interactions by enriching the robot's perceptual capabilities. The Sound Source Localization (SSL) problem in robotics [ON15, RM17] tackles the issue of acquiring the position of the sound source by determining its direction and distance from audio signals received. A typical setup usually requires the robot equipped with a microphone array [GLF13] or binaural microphones [LPZ15] such that multi-channel acoustic signals could be collected for calculating direction-of-arrival (DOA) or spectral cues from the raw audio signals. Such multi-channel acoustic information is further processed to estimate the sound source position.

However, the majority of the field in SSL is focusing on the precision of the localization, though limited, and currently restricted to localizing sound source inside a single room [GLF13, LPZ15, KEM13, SVM17, HMO18], or in simple non-line-of-sight (NLOS) scenarios, *i.e.*, behind a corner [TFK16b, TLF16], or blocked by objects [CST18]. A major issue is that the acoustic signal is severely polluted due to the high noise-to-signal ratio, reverberation, *etc.*, in a multi-room indoor environment. It is extremely hard to explicitly model the reverberation in a multi-room environment, and such model may vary from room to room, which further prohibits the conventional sound source localization methods from adapting to a multi-room setup. The effective range of the sound source localization greatly limits a domestic robot or service robot to react rapidly from users across multiple rooms, and further hinders the practical uses of SSL in service robots that deployed in large-scale scenarios. Figure 1.1 shows an example of multi-room indoor scenario where the mobile robot (highlighted with a black bounding box) stations in the hallway and an user sitting in another room. Given a verbal command from the user, the robot needs to determine from which room the verbal command comes, so that it can explore to that room, find the user, and perform the task that the user assigned. In such multi-room scenario, conventional sound source localization algorithms may not work due to the aforementioned issues. Therefore, we proposed an self-supervised incremental learning framework for mobile robots acquiring the capability of localizing the human sound source using a microphone array in a complex indoor environment consisting of multiple rooms. 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.

1.1 Related Work

1.1.1 Sound Source Localization

In the field of SSL, prior work mainly adopts a wide range of signal processing methods [ON15, RM17], 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 [GM15, GM16]. They are, however, limited to the single room scenario due to the high noise-to-signal ratio, reverberation, *etc.*, in multi-room scenarios. 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. 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.

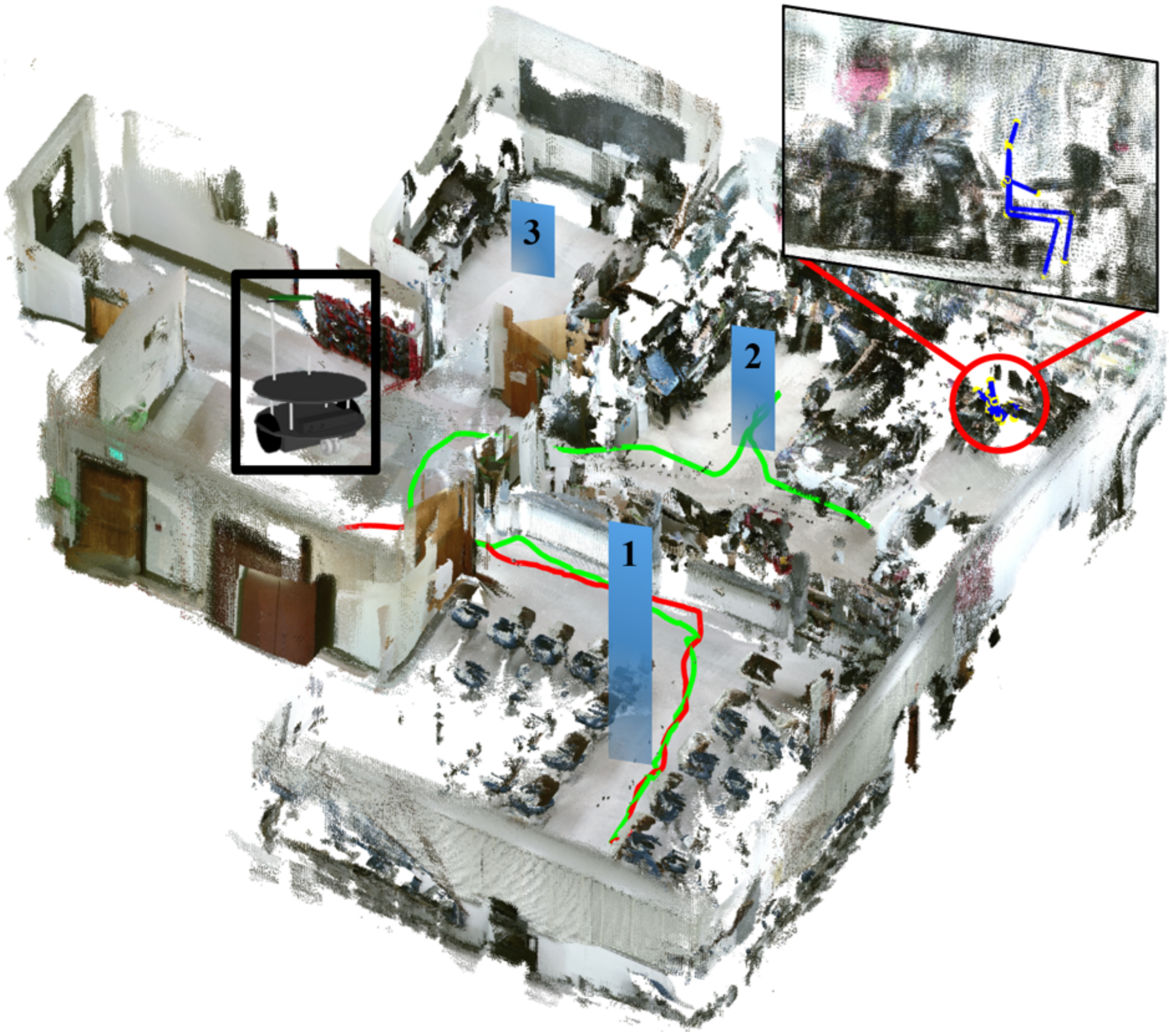

Figure 1.1: A typical indoor environment setup consisting of multiple rooms

SSL for the NFOV target has also been attempted. For instance, [TFK16b, TFK16a] 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, [TLF16] combines diffraction and reflection directions to localize the target around a corner in an anechoic chamber. Similarly, [CST18] 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.

1.1.2 Deep Neural Networks

The recent advancements of Deep Neural Networks (DNNs) [HOT06] allow machine learning methods reach a remarkable level in some specific tasks, even arguably better than human, *e.g.*, control [DCH16, MKS15], grasping [MLN17, LLS15], object recognition [HZR15, IS15], learning from demonstration [ACV09]. It is proven to be an effective way to extract implicit features that are robust against noises and interference if a large amount of training data is provided. DNNs are widely used in natural language processing and speech recognition [DY14], which are orthogonal to the SSL problem. Although DNNs-based methods have been applied to SSL problems and demonstrated decent performance [YWH16, TK16, YNO17, PC17, HMO18]. They are also limited to the single room scenario, and requiring training data of sound sources inside every new environment. Such prior methods suffer from two major issues that prevent them from being applied in a larger scale environment: (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 of SSL in complex, large-scale indoor environments.

1.1.3 Active Sensing

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 [OKK17] actively based on the data received to improve performance. However, it lacks the capability of exploration in the environment, and cannot adapt to a large-scale environment where acoustic signal are severely polluted by the reverberations from multiple rooms. By contrast, mobile robots with sound source mapping can actively navigate in large space to localize sound sources [KEM13, SVM17]. [LVH15] also utilized a mobile robot to collect ground truth acoustic data. However, they do not leverage the observed new data and the capability of exploration to improve the model.

1.2 Framework Overview

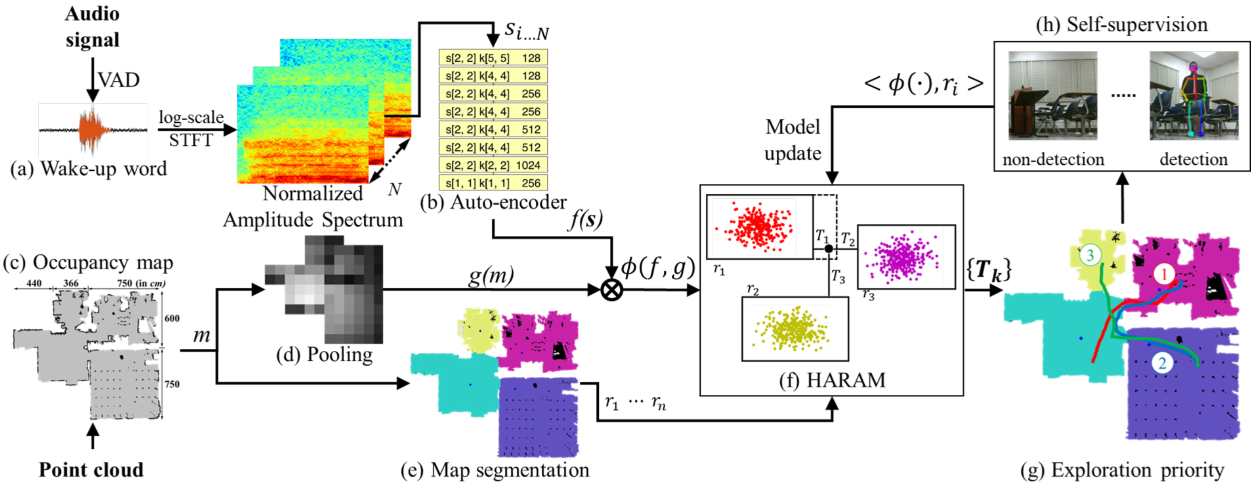

Figure 1.2: A self-supervised incremental learning framework

We propose a three-step self-supervised incremental learning framework for mobile robot's SSL in indoor environments, summarized in Figure 1.2, to address the difficulties of collecting training data and adapting the trained model to estimate the location of the acoustic signals from untrained sources in unknown indoor environments. Figure 1.2 captured the whole picture of the proposed framework, where Figure 1.2(a) illustrates the multi-channel signals of the user's wake-up word that is picked up by the microphone array mounted on the robot and further processed by the Voice Activity Detection (VAD). Signal from every channel is transferred to a log-scale amplitude spectrum via Short-Time Fourier Transform (STFT) and further normalized to $[0, 1]$, from which Figure 1.2(b) an auto-encoder is trained to extract implicit acoustic features. In Figure 1.2(b), each block represents a 2D convolution with stride s[·, ·], kernel size k[·, ·] and the number of channels. In addition, Figure 1.2(c) represents an occupancy map obtained from the reconstructed point cloud is down-sampled by pooling Figure 1.2(d). Figure 1.2(b)(d) together form the feature for the learning model. Figure 1.2(e) represents the map segmentation, where individual rooms are segmented and colored by different colors, and is automatically generated from the occupancy map. Figure 1.2(f) indicates the Hierarchical Adaptive Resonance Associative Map (HARAM) model which is adopted to predict the priority rank of rooms that the robot should visit in order.

The exploration priority is shown in Figure 1.2(g) where the robot self-supervises the learning by exploring the rooms according to the priority ranking. Figure 1.2(h) illustrates the robot self-supervised exploration and data collection, during which a data sample will be labeled as a positive sample if the robot detects the user, otherwise labeled as a negative sample, which will be used to update the HARAM model incrementally.

Our proposed framework contains three steps: (i) Localization step, 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; (ii) Incremental learning step, in contrast to batch learning methods, we adopt 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; (iii) Self-supervision step, 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.

Take a typical multi-room setup (see Figure 1.1) as an example, where the mobile robot (highlighted with a black bounding box) stations in the hallway. 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.

In summary, we introduce an incremental learning framework for SSL in the indoor setting with multiple rooms that allows the system to accumulate the training data on-the-fly. The framework is incorporated 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. Furthermore, 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.

CHAPTER 2

Feature Extraction

This chapter introduces the feature extraction process in our proposed framework. The features consist of both the acoustic features based on the collected acoustic signals from the microphone array and the geometry feature extracted from the SLAM result which encodes both the geometry structure of the environment and the robot's current position. These features encode the location information of the sound source and will be further processed by the learning model to estimate the location of the sound source.

2.1 Acoustic Feature

The acoustic feature is extracted from the raw acoustic signal from the microphone array mounted on the robot. In order to recognize the acoustic signals from the human's wake-up word and distill the data of interest out of the background noise, Voice Activity Detection (VAD) is adopted, and we utilize the state-of-the-art Google WebRTC VAD with a frame size 20 ms in duration. Figure 1.2(a) shows one example of the detected voice segment, consisting of multi-channel acoustic signals. Specifically, there are 16 channels in our experiment since there is a total of 16 microphones on the microphone array. Each channel of the detected audio segment is transformed to its amplitude spectrum using the log-scale STFT with an FFT size of 1024, and further normalized to $[0, 1]$. Figure 2.1(a) shows one example of the normalized spectrum with a dimension of 255×255, where x-axis encodes the acoustic information in the time domain, and y-axis encodes the acoustic information in the frequency domain.

The total dimension of the normalized spectrum is still large since there could be mul-

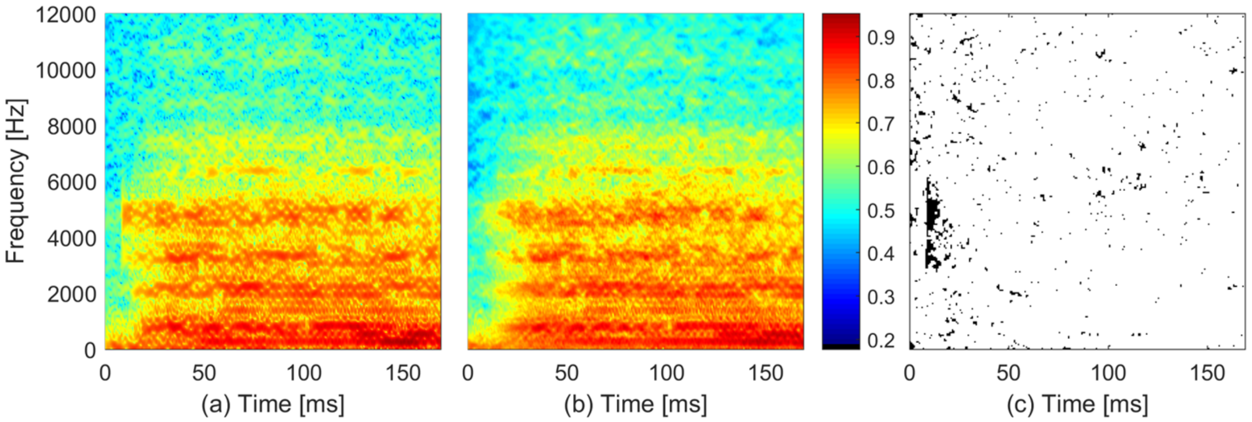

Figure 2.1: (a) The original spectrum normalized to $[0, 1]$, (b) The reconstructed spectrum using an auto-encoder, (c) The reconstruction error as a binary image

tiple channels on a microphone array, 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 log-scale amplitude spectrum per channel. Figure 2.2 depicts our spectrum auto-encoder structure where yellow blocks indicate the encoder part of the auto-encoder, the blue blocks indicate the decoder part of the auto-encoder, and the green block represents the extracted embedding for the input spectrum which is of size 255 × 255. The proposed spectrum auto-encoder contains multiple convolutional layers and fully connected layers, where the convolutional layer is represented by stacked rectangles, and the fully connected layer is represented by a vertical rectangle. Each convolutional block represents a 2D convolution with stride s[·, ·], kernel size k[·, ·] and the number of channels. A flattened operation has been conducted between the convolutional layer and the fully connected layer. Additionally, each convolutional layer is followed by a Leaky-ReLU activation layer and the batch normalization, whereas each fully connected layer is only followed by a Leaky-ReLU activation layer. 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:

$$\mathcal{L}(\theta; \mathbf{s}) = \frac{1}{N} \sum_{i=1}^{N} \ell(s_i, \psi(s_i; \theta)), \tag{2.1}$$

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 [ERR17]. Such embedding contains implicit features of one spectrum, denoted as $f(s_i)$. Figure 2.1(b) shows an example of the reconstructed spectrum with the reconstruction error shown in Figure 2.1(c), in which the black pixel indicates the relative error larger than 5%.

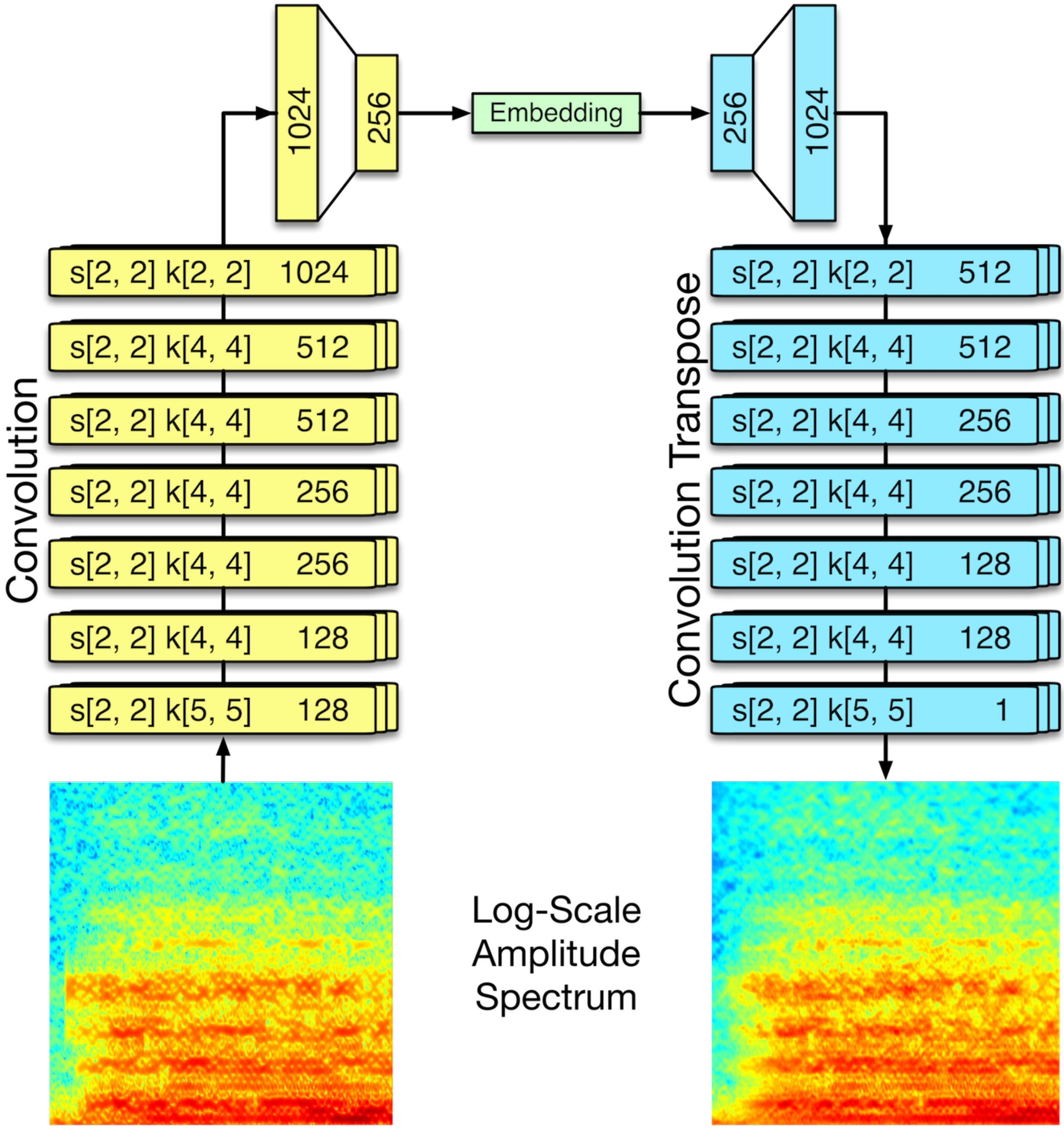

Figure 2.2: Spectrum Auto-Encoder Model

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, and also emphasize the distinction between signals from different microphone channels.

2.2 Environment Geometry Feature

To obtain richer environmental information through SLAM, we use a Kinect v2 sensor to construct the 3D structure of the environment using RTAB-Map [LM14]. Figure 1.1 depicts the reconstructed environment in the form of the registered point cloud, which can be easily converted to a 2D occupancy map by projecting the 3D point cloud to a 2D space. 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 1.2(c) shows the original map m that spans about 15.5×13.5 square meters is compressed to a 12×11 matrix $g(m)$, where each element is normalized to $[0, 1]$. Figure 1.2(d) shows an example geometry feature, the whiter the element is, the closer the element to the robot's current position.

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.

CHAPTER 3

Sound Source Localization Framework

3.1 Localization by Ranking

We adopt the Hierarchical Adaptive Resonance Associative Map (HARAM) algorithm [Tan95, BS15] 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 [Tan95, BS15] for in-depth details.

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 $\phi_i^c \equiv 1 - \phi_i$ and $r_i^c \equiv 1 - r_i$, representing both on-responses and off-responses of the input vector. The weight vectors $\boldsymbol{\omega}_k^{\phi}$ and $\boldsymbol{\omega}_k^{r}$, $k = 1, \dots, R$ 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

$$T_k(\boldsymbol{\Phi}, \mathcal{R}) = \gamma \frac{|\boldsymbol{\Phi} \wedge \boldsymbol{\omega}_k^{\phi}|}{\alpha_\phi + |\boldsymbol{\omega}_k^{\phi}|} + (1-\gamma) \frac{|\mathcal{R} \wedge \boldsymbol{\omega}_k^{r}|}{\alpha_r + |\boldsymbol{\omega}_k^{r}|}, \tag{3.1}$$

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

$$T_* = \max\{T_k : k = 1, \dots, R\}. \tag{3.2}$$

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

$$\frac{|\boldsymbol{\Phi} \wedge \boldsymbol{\omega}_*^\phi|}{|\boldsymbol{\Phi}|} \geqslant \rho_\phi, \frac{|\mathcal{R} \wedge \boldsymbol{\omega}_*^r|}{|\mathcal{R}|} \geqslant \rho_r. \tag{3.3}$$

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

$$\begin{cases} \boldsymbol{\omega}_*^{\phi(\mathrm{new})} = \lambda_\phi(\boldsymbol{\Phi} \wedge \boldsymbol{\omega}_*^{\phi(\mathrm{old})}) + (1-\lambda_\phi)\boldsymbol{\omega}_*^{\phi(\mathrm{old})} & \text{(3.4a)} \\ \boldsymbol{\omega}_*^{r(\mathrm{new})} = \lambda_r(\mathcal{R} \wedge \boldsymbol{\omega}_*^{r(\mathrm{old})}) + (1-\lambda_r)\boldsymbol{\omega}_*^{r(\mathrm{old})} & \text{(3.4b)} \end{cases}$$

where λ_ϕ and $\lambda_r \in [0,1]$ are the learning rates. Take an example shown in Figure 1.2(f), the hyperbox of cluster r_1 expands (see the dash box) to include the new sample.

Ranking By sorting $\{T_k\}$ in Equation 3.2 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 1.2(f). 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.*, $\mathrm{dist}(T_1) < \mathrm{dist}(T_2) < \mathrm{dist}(T_3)$), the robot will explore in the order of room 1, room 2, and room 3.

3.2 Self-Supervision

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. Before assigning the ground truth label to a collected sample, the types of labels needs to be automatically retrieved from the environment. Specifically, in our scenario, the types of labels is the candidate rooms that need to be explored in an indoor environment. 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 [BJL16], and an example room segmentation is shown in Figure 1.2(e). Specifically, we use the Distance Transform-based Segmentation, that is, 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.

The self-supervision is driven by the active exploration which is based on the priority ranking estimated from the HARAM model. The HARAM model produces the distance (see Equation 3.1) 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 3.1, and the performance is expected to improve with the increasing number of the sample received.

During the active exploration, the data collection is done by automatically labeling the collected data samples. The robot will subsequently navigate to each room following the priority ranking 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 [CSW17], to detect the human as the sound source in a room. Figure 3.1 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 is then updated according to Equation 3.4.

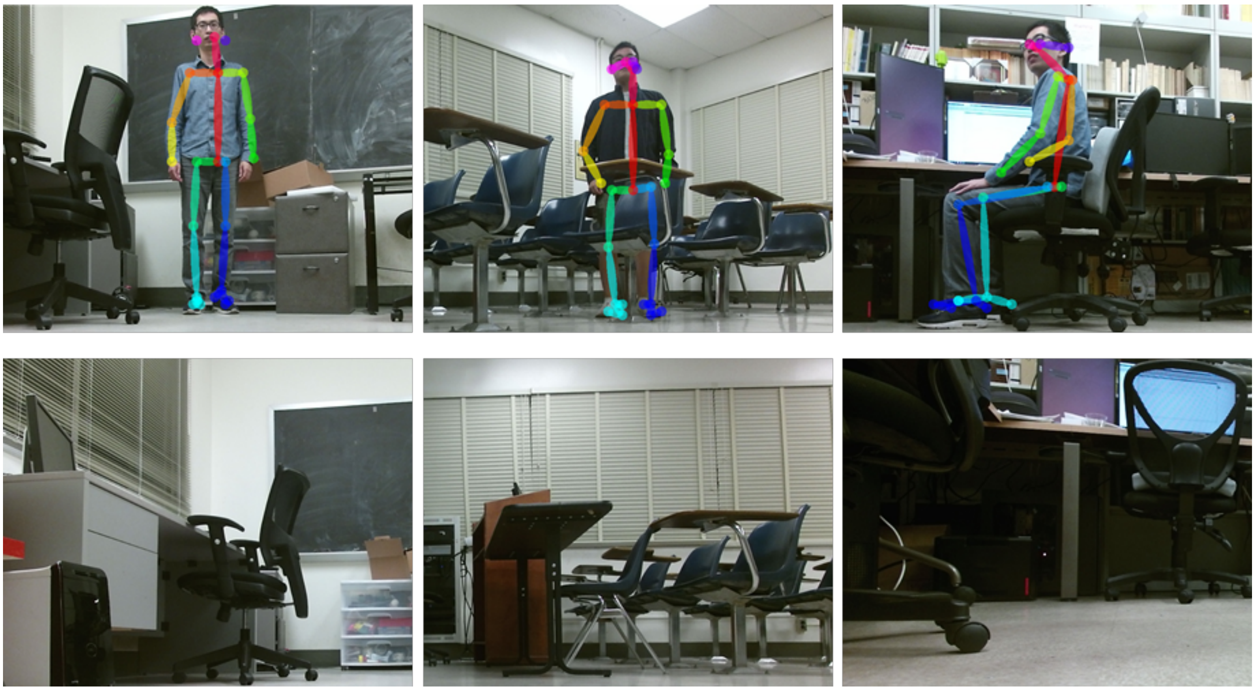

Figure 3.1: (Top) Examples of the human pose detection. (Bottom) Non-detection examples.

CHAPTER 4

Experiment

4.1 Robot Platform and Experimental Setup

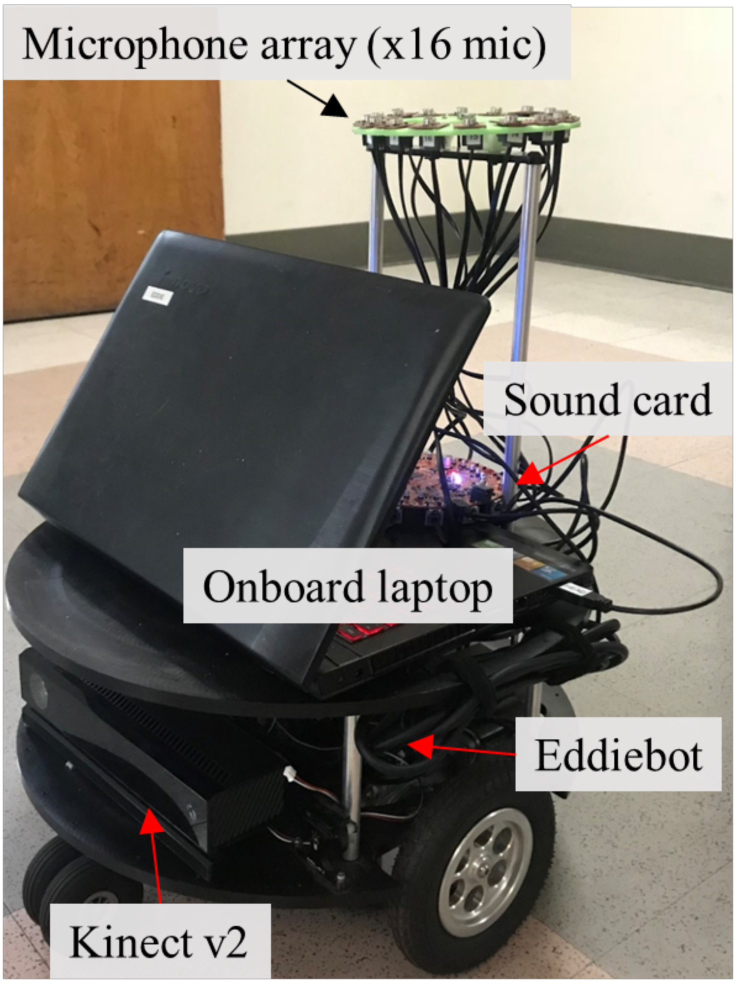

Figure 4.1: EddieBot robot setup

We evaluate the proposed framework using a system on a Parallax Eddie Robot Platform which allows *real-time* data acquisition, processing, and learning. Figure 4.1 shows the hardware details of our Eddie Robot platform where 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 reconstruct the environmental 3D structure information as well as to detect human poses. The entire system runs online in ROS with an onboard laptop. Table 4.1 lays out the specifications and parameters of the system. More specifically, the 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 onboard laptop that runs the learning algorithm in real-time.

Table 4.1: Experimental hardware specifications

Parameter	Value
RGB-D sensor	kinect V2
FOV	$70° \times 60° \ (W \times H)$
Microphone array	N = 16
Type	uniform circular array
Array radius	9 [cm]
Sampling frequency	48 [kHz]
Laptop	Lenovo Y500
CPU	2.4 [GHz]
Memory	8 [GB]
GPU	GT 750M

We test the proposed sound source localization approach in a physical world with a multi-room setup, as shown in Figure 4.2. The robot stations in the hallway, and the sound source is located at one of the rooms, which are mostly in the robot's non-field-of-view. 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 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.

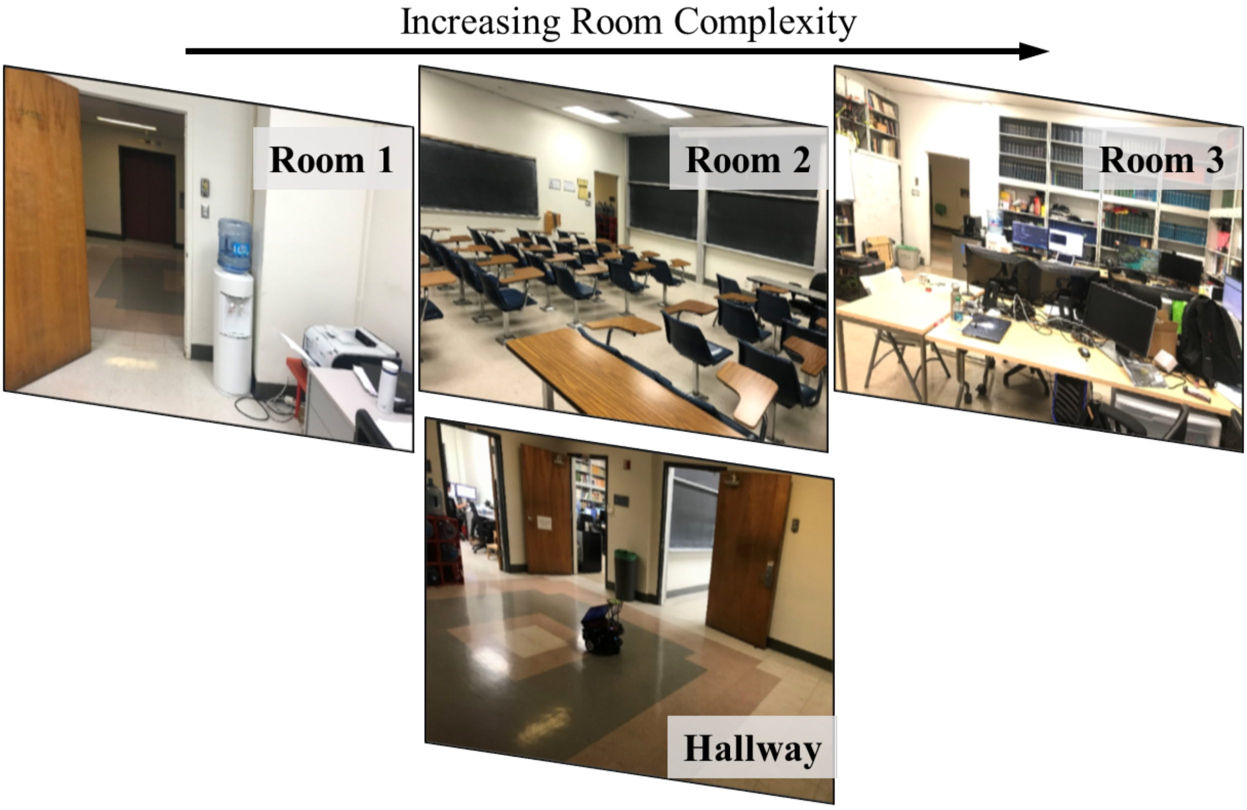

Figure 4.2: A multi-room environment used in experiments

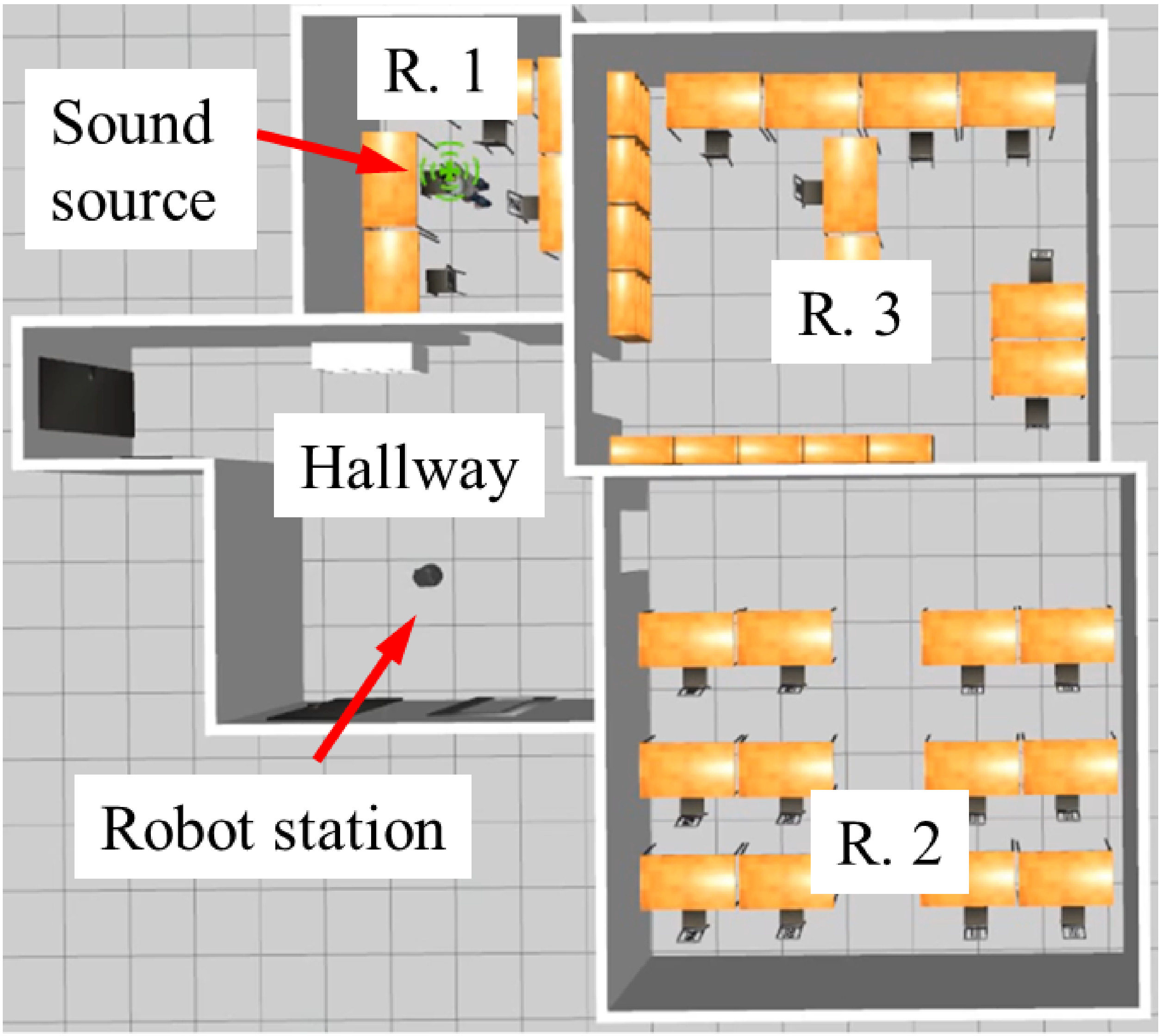

Figure 4.3: Simulation environment

4.2 Incremental Learning via Active Exploration

In order to evaluate the incremental learning process, we further reconstruct and visualize the process in the Gazebo simulator. The simulation environment is shown in Figure 4.3 where 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. All the acoustic samples are collected in a physical multi-room setup, and the evaluation is also performed in a physical environment.

Figure 4.4 illustrates several keyframes of the incremental learning process. 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, blue, and green trajectory indicates the first, the second and the third room the robot visits, respectively. The number of lines depicts the number of trails used to find the sound source location. 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 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.

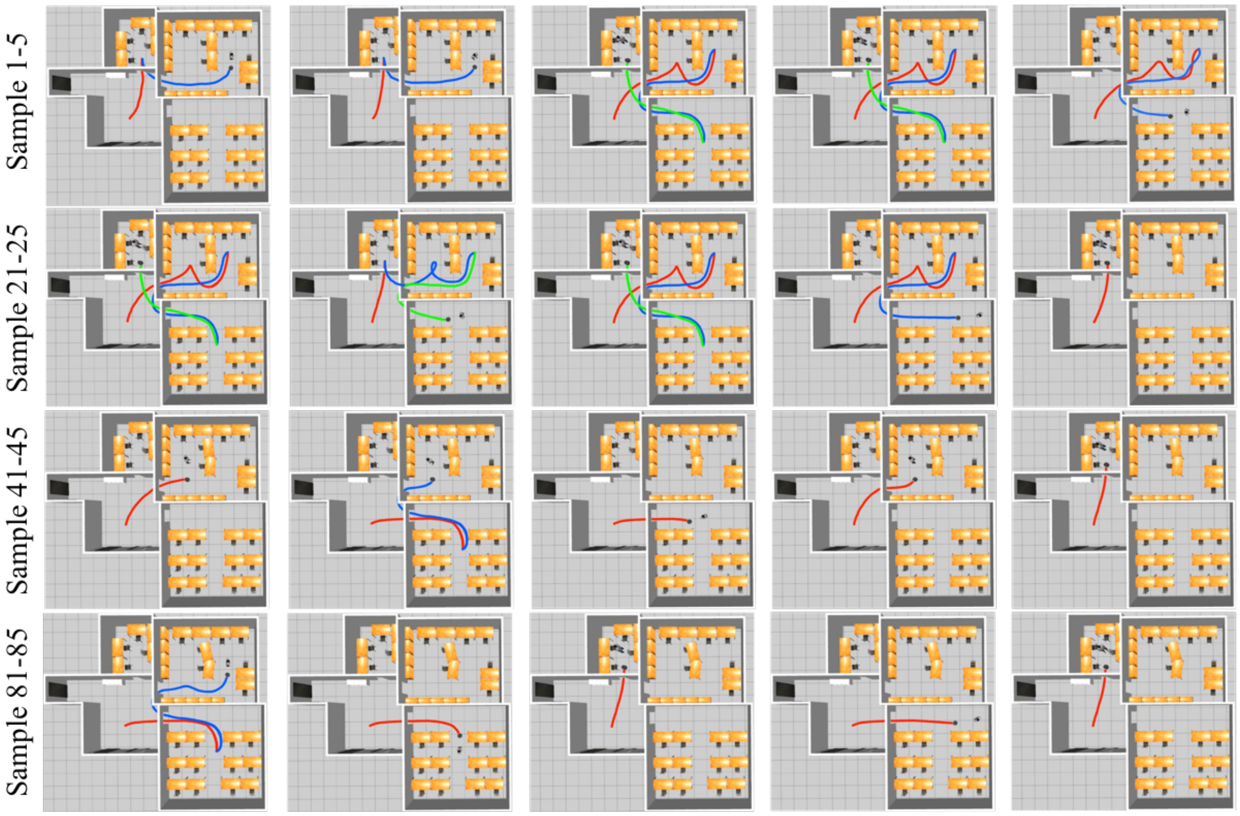

Figure 4.4: Localization performance improves as the amount of samples increases

To 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 4.5 shows the boxplot of how many incorrect visits a robot needs to find the correct sound source locations with an increment of 10 samples. In Figure 4.5, 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. 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.

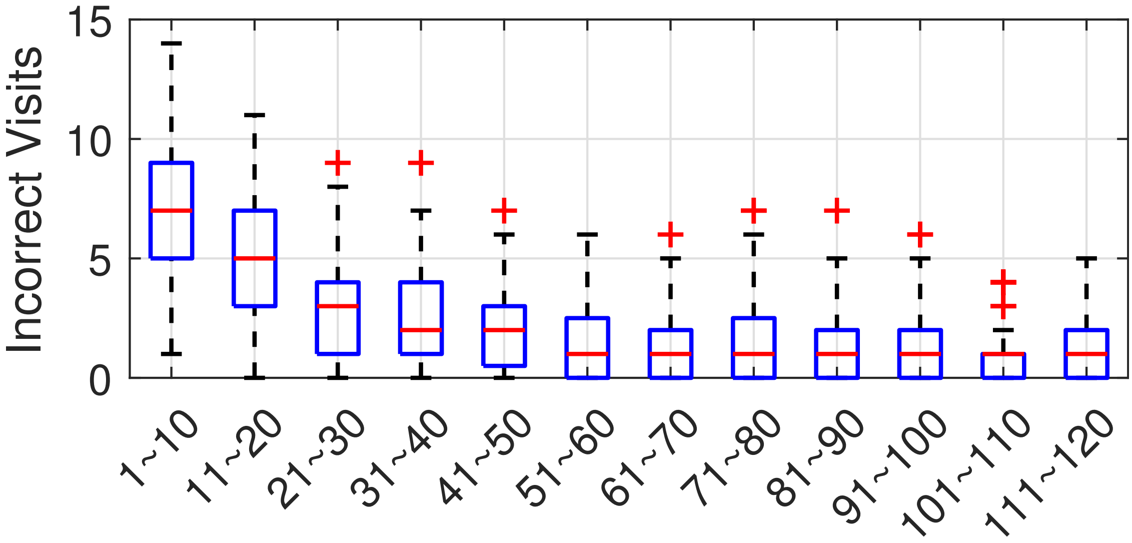

Figure 4.5: The number of incorrect visits before finding the correct sound source locations in every 10 samples over 100 trails.

Figure 1.1 shows a test running in a physical environment, in which the robot finds the user in its second visit. Figure 4.6 shows another example using only the first visit, in which the robot locates the correct sound source with only one visit. The top row shows keyframes from the robot view for navigation with human pose detection, and the bottom row presents the corresponding third-person views.

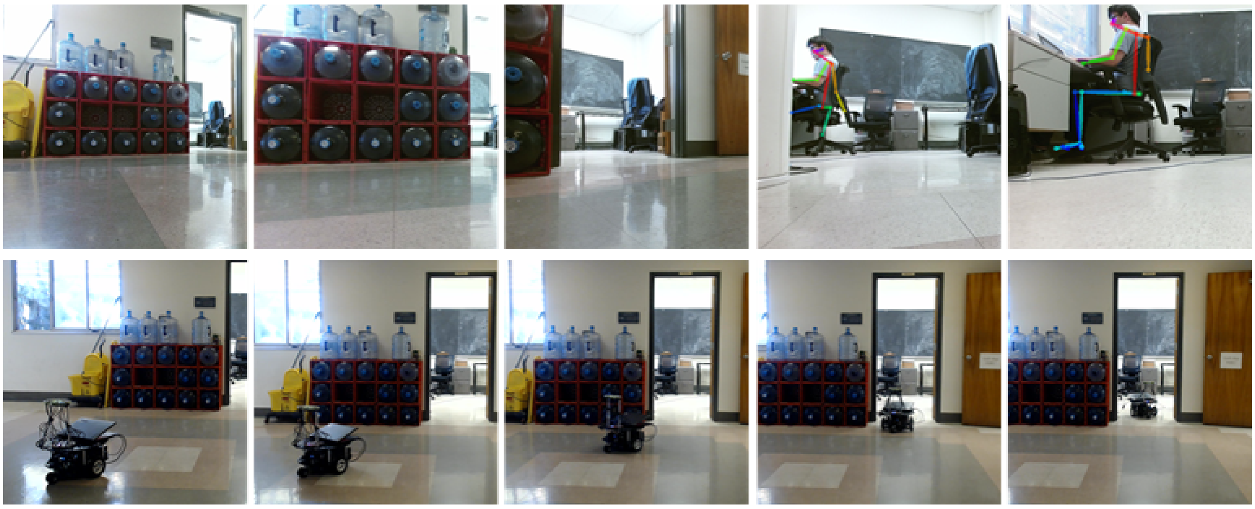

Figure 4.6: Localizing human sound source in a physical environment

According to a report from *IFTTT* [HH17], 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.

4.3 Comparison Study

The performances using different microphone array configurations are 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. Figure 4.7 shows the mean accuracy of the proposed method using four different microphone array configurations. The color strips indicate the 95% confidence interval over 100 trails. Overall, more microphones lead to better performance with minor fluctuations in the early stage.

We also 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.

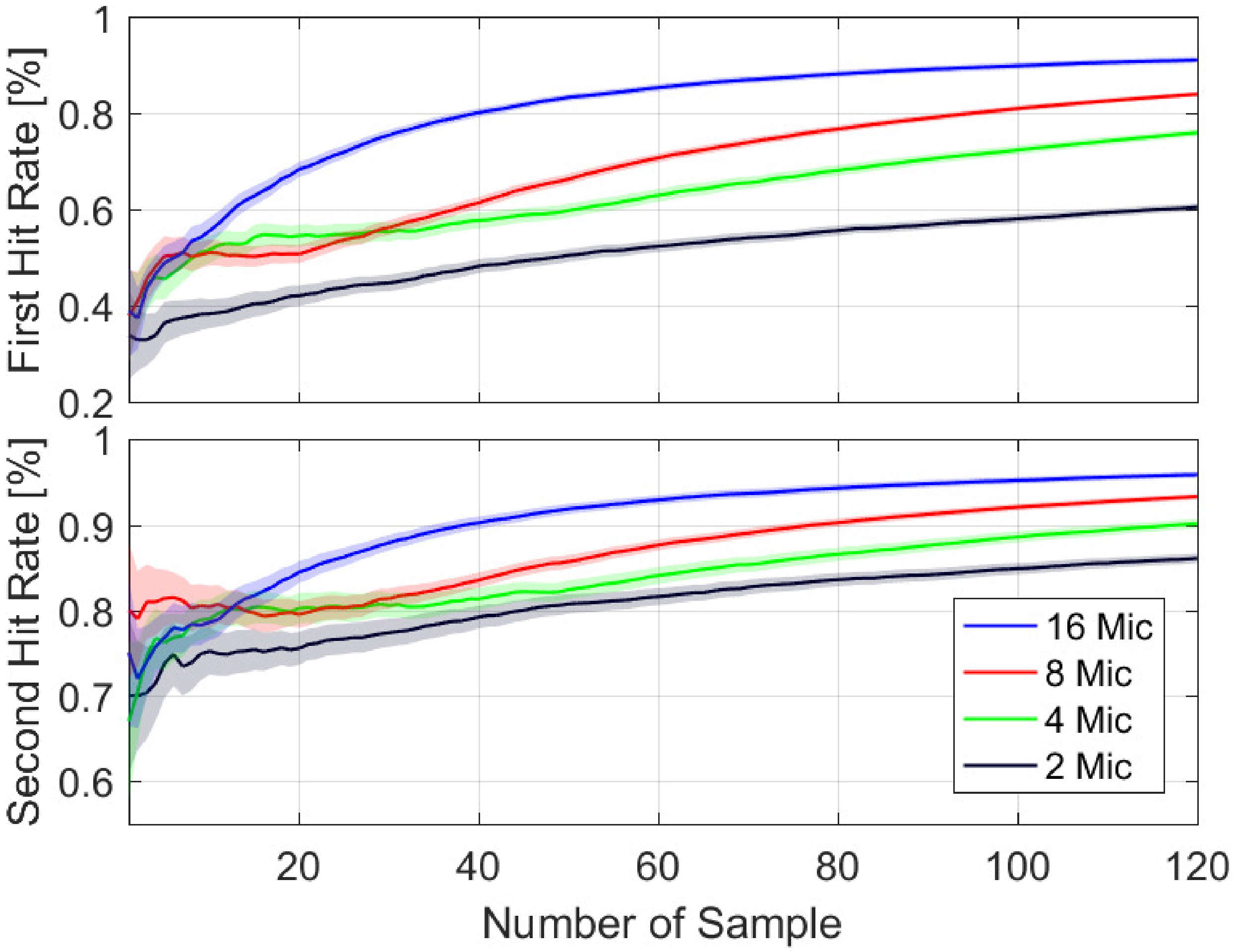

Figure 4.7: Microphone configuration comparison result

2. **HARAM + GCCFB**. We add a mel-scale filter bank on top of the GCC-PHAT [HMO18], 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 4.8 shows the comparison results. The mean accuracy of (blue) the proposed method and (green and red) two baselines. The first and the second hit rates indicate the

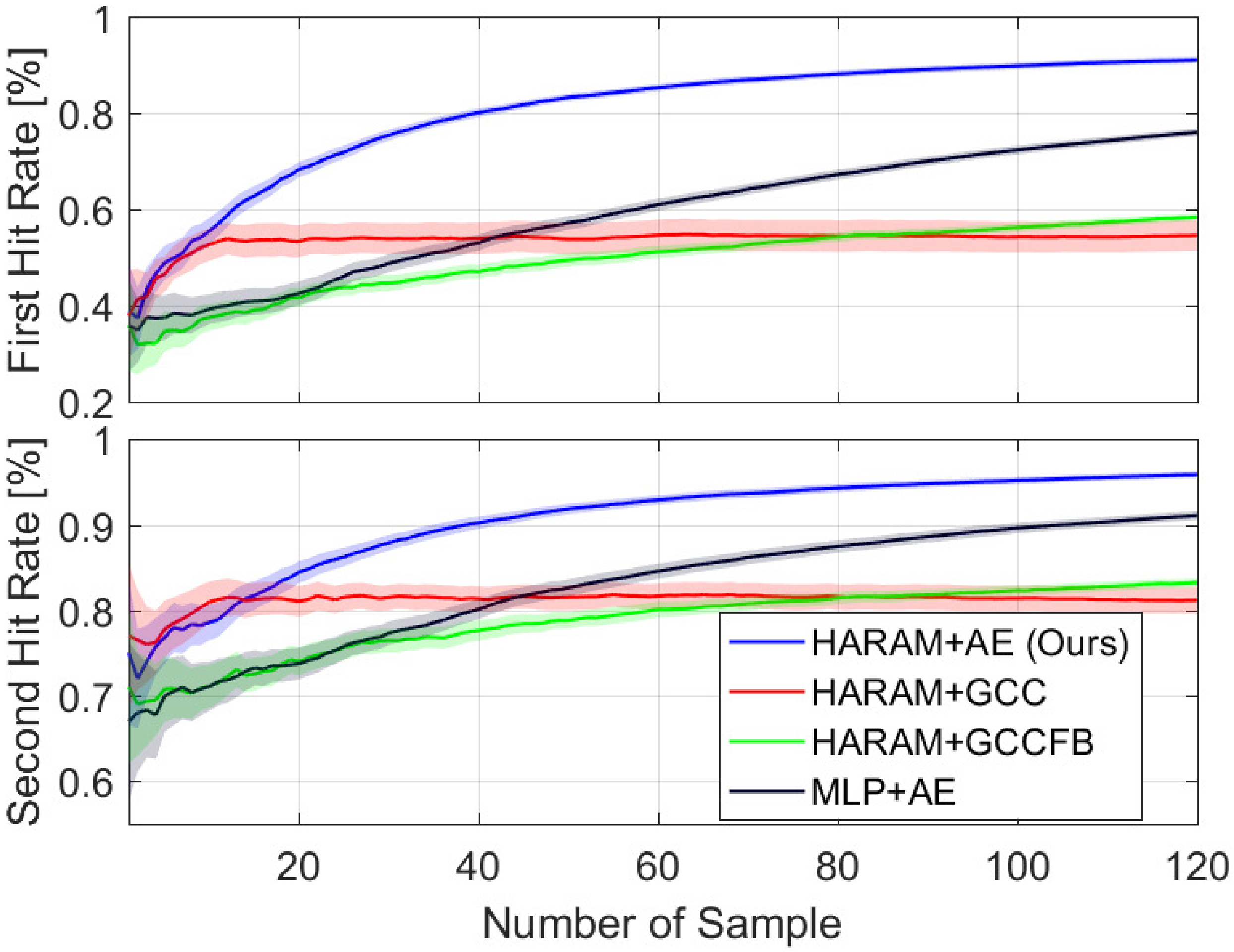

Figure 4.8: Model comparison result

robot finds the correct sound source locations within one and two visits, respectively. The color strips indicate the 95% confidence interval over 100 trails. 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 surpasses all three baselines after receiving 15 samples.

CHAPTER 5

Conclusion

In this thesis, we 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 of the sound source located from high to low. The self-supervision is achieved through robot actively exploring the rooms according to the predicted rank, detecting sound sources by human poses, assigning the ground truth label to the collected data, and improving the learned model performance incrementally. We also show that the framework does not require pre-collected data and can be directly applied to real-world scenarios without any human supervisions or interventions. 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.

5.1 Future work and 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.

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.

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. The high flexibility of the proposed framework enables the possibility of future extension and improvement.

Relation with active learning. In future work, there is possible to adopt an active learning scheme that selects the most meaningful samples regarding best deciding the decision boundary. Such that only a subset of samples can retain good performance, which can further shrink the memory required when deploying on a robot platform.

REFERENCES

[ACV09] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. “A survey of robot learning from demonstration.” *Robotics and autonomous systems*, **57**(5):469–483, 2009.

[BJL16] Richard Bormann, Florian Jordan, Wenzhe Li, Joshua Hampp, and Martin Hägele. “Room segmentation: Survey, implementation, and analysis.” In *International Conference on Robotics and Automation (ICRA)*, 2016.

[BS15] Fernando Benites and Elena 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.

[CST18] Kyunghoon Cho, Junghun Suh, Claire J Tomlin, and Songhwai Oh. “Reflection-Aware Sound Source Localization.” In *International Conference on Robotics and Automation (ICRA)*, 2018.

[CSW17] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. “Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields.” In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[DCH16] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. “Benchmarking deep reinforcement learning for continuous control.” In *International Conference on Machine Learning (ICML)*, 2016.

[DY14] Li Deng, Dong Yu, et al. “Deep learning: methods and applications.” *Foundations and Trends® in Signal Processing*, **7**(3–4):197–387, 2014.

[ERR17] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. “Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders.” In *International Conference on Machine Learning (ICML)*, 2017.

[GLF13] François Grondin, Dominic Létourneau, François Ferland, Vincent Rousseau, and François Michaud. “The ManyEars open framework.” *Autonomous Robots*, **34**(3):217–232, 2013.

[GM15] François Grondin and François 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.

[GM16] François Grondin and François 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.

[HH17] Garrett Hulfish and Garrett Hulfish. “IFTTT Data Reveals How People Most Use Google Home or Alexa.”, Jul 2017. Accessed: 2018-09-05.

[HMO18] Weipeng He, Petr Motlicek, and Jean-Marc Odobez. "Deep Neural Networks for Multiple Speaker Detection and Localization." In *International Conference on Robotics and Automation (ICRA)*, 2018.

[HOT06] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. "A fast learning algorithm for deep belief nets." *Neural Computation*, **18**(7):1527–1554, 2006.

[HZR15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification." In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[IS15] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." In *International Conference on Machine Learning (ICML)*, 2015.

[KEM13] Nagasrikanth Kallakuri, Jani Even, Yoichi Morales, Carlos Ishi, and Norihiro Hagita. "Probabilistic approach for building auditory maps with a mobile microphone array." In *International Conference on Robotics and Automation (ICRA)*, 2013.

[LLS15] Ian Lenz, Honglak Lee, and Ashutosh Saxena. "Deep learning for detecting robotic grasps." *The International Journal of Robotics Research*, **34**(4-5):705–724, 2015.

[LM14] Mathieu Labbe and François Michaud. "Online global loop closure detection for large-scale multi-session graph-based SLAM." In *International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[LPZ15] Hong Liu, Cheng Pang, and Jie 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.

[LVH15] Jonathan Le Roux, Emmanuel Vincent, John R Hershey, and Daniel PW 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.

[LZZ19] Hangxin Liu, Zeyu Zhang, Yixin Zhu, and Song-Chun Zhu. "Self-Supervised Incremental Learning for Sound Source Localization in Complex Indoor Environment." In *International Conference on Robotics and Automation (ICRA)*, 2019.

[MKS15] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. "Human-level control through deep reinforcement learning." *Nature*, **518**(7540):529–533, 2015.

[MLN17] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken 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.

[OKK17] Wataru Odo, Daisuke Kimoto, Makoto Kumon, and Tomonari Furukawa. "Active sound source localization by pinnae with recursive bayesian estimation." *Journal of Robotics and Mechatronics*, **29**(1):49–58, 2017.

[ON15] Hiroshi G Okuno and Kazuhiro Nakadai. "Robot audition: Its rise and perspectives." In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[PC17] Pasi Pertilä and Emre Cakir. "Robust direction estimation with convolutional neural networks based steered response power." In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[RM17] Caleb Rascon and Ivan Meza. "Localization of sound sources in robotics: A review." *Robotics and Autonomous Systems*, **96**:184–210, 2017.

[SVM17] Daobilige Su, Teresa Vidal-Calleja, and Jaime Valls Miro. "Towards real-time 3D sound sources mapping with linear microphone arrays." In *International Conference on Robotics and Automation (ICRA)*, 2017.

[Tan95] Ah-Hwee Tan. "Adaptive resonance associative map." *Neural Networks*, **8**(3):437–446, 1995.

[TFK16a] Kuya Takami, Tomonari Furukawa, Makoto Kumon, and Gamini Dissanayake. "Non-field-of-view acoustic target estimation in complex indoor environment." In *Field and Service Robotics*, 2016.

[TFK16b] Kuya Takami, Tomonari Furukawa, Makoto Kumon, Daisuke Kimoto, and Gamini Dissanayake. "Estimation of a nonvisible field-of-view mobile target incorporating optical and acoustic sensors." *Autonomous Robots*, **40**(2):343–359, 2016.

[TK16] Ryu Takeda and Kazunori 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.

[TLF16] Kuya Takami, Hangxin Liu, Tomonari Furukawa, Makoto Kumon, and Gamini Dissanayake. "Non-field-of-view sound source localization using diffraction and reflection signals." In *International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[YNO17] Nelson Yalta, Kazuhiro Nakadai, and Tetsuya Ogata. "Sound source localization using deep learning models." *Journal of Robotics and Mechatronics*, **29**(1):37–48, 2017.

[YWH16] Yang Yu, Wenwu Wang, and Peng Han. "Localization based stereo speech source separation using probabilistic time-frequency masking and deep neural networks." *EURASIP Journal on Audio, Speech, and Music Processing*, **2016**(1):7, 2016.