This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Diffraction- and Reflection-Aware Multiple Sound Source Localization

Inkyu An, *Student Member, IEEE*, Youngsun Kwon, *Student Member, IEEE*, and Sung-eui Yoon, *Senior Member, IEEE*

***Abstract*—In this article, we present a novel localization method for multiple sources in indoor environments. Our approach can estimate different propagation paths, including the reflection and diffraction paths of sound waves based on a backward ray tracing technique. To estimate diffraction propagation paths, we combine a ray tracing algorithm with a uniform theory of diffraction model by exploiting the diffraction properties as propagation paths bend around the wedges of obstacles. We reconstruct the 3-D environments and wedges of obstacles in the precomputation phase and utilize these outcomes to generate primary, reflection, and diffraction acoustic rays in the runtime phase. We localize multiple sources when identifying the convergence regions of these acoustic rays based on Monte Carlo localization (MCL). Our approach supports not only stationary but also moving sources of human speech and clapping sounds. Our approach can also handle nonline-of-sight (NLOS) sources and distinguish between active and inactive source states. We evaluated and analyzed our algorithm in multiple scenarios containing obstacles and NLOS sources. Our approach can localize moving sources with the average of distance errors of 0.65 and 0.74 m in single and multiple source cases, respectively, in rooms, 7 m by 7 m in size with a height of 3 m; errors are measured according to the L2 distance between the estimated and actual source positions. We observed a 130% improvement of the localization accuracy over the prior work (J.-M. Valin *et al.*).**

***Index Terms*—Localization, recognition, robot audition, sound source localization (SSL).**

I. Introduction

As MOBILE robots are increasingly applied in various areas, there is considerable interest in developing new and improved methods for localization. The main goal of localization methods is to compute the current location of the mobile robot with respect to its environment. These techniques commonly assume that a map of the environment is given, and different sensors on the robot are used to estimate its position and orientation in the environment. Some commonly used sensors include GPS, charge-coupled device, and depth cameras, acoustic types, among others.

Recently, there have been growing attention and new approaches to the use of acoustic sensors for the localization of active sound sources, including sonar signal processing for underwater localization and microphone arrays for indoor and outdoor scenes. The recent use of smart microphones in commodity or Internet of Things devices (e.g., Amazon Alexa) has triggered interest in better acoustic localization methods [2], [3].

Localization methods with acoustic sensors work by utilizing various properties of sound waves. Sound waves emitted from a source are transmitted through a medium and reach the listener or microphone directly or after undergoing different wave interactions, such as reflection, interference, diffraction, and scattering.

Some of the earliest work on sound source localization (SSL) makes use of the time difference of arrival (TDOA) at the receiver [4]–[7]. There have been many beamforming [8], [9] and subspace-based methods [10]–[12] related to localizing a sound source. These methods only exploit the direct sound and its direction, i.e., the direction of arrival (DoA) of the sound, at the receiver and do not take into account reflections or other wave effects. As a result, they do not provide sufficient accuracy of SSL for many applications.

Recent techniques have been proposed to localize positions of sound sources. Some techniques localize the positions under constraints accumulating the incoming sensor data, corresponding to the DoA of direct sound, measured from different locations and orientations [13]–[15]. Other techniques have tried to localize moving sources with intermittent sound signals using a filtering process [1], [16], [17]. However, they only consider information of direct sounds and assume that there is no obstacle between a microphone array and a sound source. Many sound sources can also be mobile, i.e., a moving source, and they may not be directly in the line of sight (LOS) of the listener, known as a nonline-of-sight (NLOS) source, due to obstruction by obstacles. Therefore, in the NLOS source case, there may not be much of a contribution in terms of direct sound, and the accuracy of these method can deteriorate.

There have been efforts to model reflection sound in addition to direct sound based on ray tracing techniques [18]–[20]. They

Manuscript received July 18, 2021; accepted September 27, 2021. This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korea government (MSIT) under Grant IITP-2015-0-00199 and in part by the National Research Foundation of Korea (NRF) Grant 2019R1A2C3002833 funded by the Korea government (MSIT). This article was recommended for publication by Associate Editor O. Stasse and Editor F. Chaumette upon evaluation of the reviewers' comments. *(Corresponding author: Sung-eui Yoon.)*

Inkyu An and Youngsun Kwon are with the School of Computing, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: inkyu.an@kaist.ac.kr; youngsun.kwon@kaist.ac.kr).

Sung-eui Yoon is with the Faculty of School of Computing, Korea Advanced Institute of Science and Technology, Deajeon 34141, South Korea (e-mail: sungeui@gmail.com).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TRO.2021.3118966.

Digital Object Identifier 10.1109/TRO.2021.3118966

1552-3098 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

approximate the direct and reflection propagation paths of sound using acoustic rays generated by a ray tracing technique, and the source positions are estimated from computing intersections of acoustic rays. These methods are efficient in localizing the NLOS source by modeling reflection propagation paths. However, they assume that there is only one sound source, and, thus, additional process should be necessary to identify multiple source positions among many intersections. Moreover, the time to calculate the intersections increases as the number of acoustic rays increases; the time complexity of computing intersections is $O(N^2)$ where N is the number of acoustic rays. In our tests, 75 acoustic rays have been produced in a single frame on average. It is difficult to compute every intersection of acoustic rays in real time.

The ray tracing technique, a type of geometric acoustic techniques [21]–[24], assumes the rectilinear propagation of sound waves and fits into high-frequency sounds; specular reflection is one of the high-frequency phenomena. They do not model many low-frequency phenomena such as diffraction, which is a type of scattering that occurs in the presence of obstacles whose sizes are of the same order of magnitude as the wavelength. In practice, diffraction is a fundamental mode of sound wave propagation and occurs frequently in building interiors, e.g., when the source is behind an obstacle or hidden by walls. These effects are more prominent for low-frequency sources, such as vowel sounds in human speech, industrial machinery, ventilation, and air-conditioning units.

Main contribution: We present a novel sound localization algorithm that takes into account diffraction as well as reflection, even from NLOS sources and intermittent sound signals. A key aspect of our work is that it models diffraction propagation paths of sounds by the acoustic rays and identifies multiple source locations from acoustic rays satisfying a real-time operation. The diffraction propagation paths are modeled by the uniform theory of diffraction (UTD) [25] along the wedges and approximated by the diffraction acoustic rays. Our method efficiently identifies multiple source positions using MCL, which finds out the convergence regions of rays, corresponding to intersections of rays.

Our approach supports the iterative computation and, during every iteration, can localize multiple dynamic sources as well as NLOS sources by modeling the reflection and diffraction of acoustic rays. Furthermore, our approach can distinguish between active and inactive states of intermittent sound signals in addition to continuous sounds.

During the precomputation phase, we use SLAM and primitive fitting techniques to reconstruct the 3-D map information of an indoor environment, specifically a 3-D triangular mesh and wedges of obstacles. At runtime, we generate primary acoustic rays toward the incoming sound directions as computed by a DoA estimator. Once the acoustic ray hits the reconstructed mesh, we generate reflection rays (Section III-B). Furthermore, when acoustic rays satisfy our diffraction criterion, e.g., hitting on the edge of the wedge, we also generate diffraction acoustic rays (Section III-C). We estimate multiple source positions by performing MCL to identify the ray convergence given generated acoustic rays (Section IV).

We evaluated our method in various scenarios in two indoor environments: one 7 m by 7 m in size and the other 7 m by 3.5 m in size and a height of 3 m. We also tested our approach in different environmental or experimental setups and applied our approach to the other task navigating to the NLOS source. Given these test environments, our method achieves low average errors, e.g., 0.6159 and 0.7364 m, for clapping sound and human speech, respectively, even with a moving source and an obstacle occluding the LOS between the listener and the source. Furthermore, our method demonstrates high performance, in this case, 0.5919 and 0.5271 m, respectively, on clapping sounds and human speech with multiple stationary and dynamic sources.

Previous versions of this article were published in ICRA 2018 [26] and 2019 [27]. Compared to these previous versions, we extend single-source localization of those prior methods to multiple-source localization techniques. We tested our algorithm in more diverse scenes to demonstrate the benefits of our method for localizing sound sources. Specifically, we tested the effects of changes in specular and diffuse materials (Section V-C), the compatibility of the method with different microphone arrays (Section V-D), and multiple-source scenarios (Section V-E). Unlike the previous versions where the robot equipping the microphone array was stationary, we made our robot move; it can be possible for the robot to perform various tasks, e.g., navigation, based on our approach (Section V-G). We also compared accuracies of our method to the previous work [1] which does not consider indirect sound.

II. Related Works

SSL methods have been studied to overcome the difficulty encountered by a robot when attempting to identify a speaker such as a human, machine, or even another robots, in a real environment. We explain previous research on SSL (Section II-A), after which we introduce physical-based modeling techniques that enable realistic sound generation for simulators (Section II-B). Our approach is inspired by these physical-based modeling methods.

A. Sound Source Localization

There have been many efforts to identify the sound source location, and many of them have focused on estimating the DoA.

For simple and fast DoA estimators in a 2-D space, many methods have been proposed using microphone pair signals and their TDOA. The TDOA can be estimated by using a generalized cross-correlation with phase transform [4], [5] and a difference singular value decomposition with phase transform [7] from a microphone pair. Using microphone pair signals and their TDOA, Carlo *et al.* [28] proposed a method for 2-D SSL, i.e., DoA estimation, by considering echoes.

Beamforming or subspace-based methods have been suggested to estimate DoA in a system using multiple microphones, i.e., microphone array. A delay and sum (DAS) beamformer was proposed for fast DoA estimation using the cross-correlation operation with eight microphones [29]. The fast and accurate speaker identification system for distributed meeting was

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

suggested by using a minimum variance distortionless response (MVDR) beamformer [9]. Nakamura *et al.* [12] suggested a multiple signal classification based on generalized singular value decomposition algorithm for a robust and real-time localization method.

Recently, there have been demands for estimating the actual location of the sound source, not just DoA. Valin *et al.* [1] suggested a 3-D SSL and tracking system based on a steered beamformer and particle filter. Portello *et al.* [17] presented an active source localization of an intermittent signal considering a motion of a moving binaural sensor. Nguyen *et al.* [16] presented a 2-D SSL method by updating sequential acoustic data using a mixture Kalman filter. Sasaki *et al.* [13] and Su *et al.* [14] presented 3-D SSL algorithms using a disk-shaped sound detector and a linear microphone array such as Kinect and PS3 Eye. Even *et al.* [18] presented a probabilistic 3-D mapping algorithm of sound sources accumulating acoustic information of direct sound on an occupancy grid map. These approaches consider only direct sound and, thus, are not designed for scenarios containing NLOS sources.

To localize the NLOS source, some methods have been presented based on a ray tracing technique. Kallakuri *et al.* [19] and Even *et al.* [20] suggested the NLOS source localization algorithm by modeling reflection. They produced reflected acoustic rays and computed the intersection of those rays corresponding to the source position. Our approach models diffraction as well as reflection. Our approach also computes the convergence of those rays based on the MCL to identify the source position; the MCL based technique is more efficient and faster than computing the intersection of rays.

B. Interactive Sound Propagation

There has been considerable work in acoustics and physically based modeling to develop fast and accurate sound simulators that can generate realistic sounds for computer-aided design and virtual environments. Geometry acoustic techniques have been widely utilized to simulate sound propagation efficiently using ray tracing techniques, and those ray tracing techniques are efficient to model sound propagations at high frequencies.

At high frequencies, the propagation of the sound waves can be approximated as traveling in straight and bouncing off the boundaries [21]. An estimation of the acoustic impulse response of high-frequency propagation between the source and the listener was performed using image-source-based ray tracing [22], Monte Carlo path tracing [23], or a hybrid combination of geometric and numeric techniques [24].

Low-frequency wave phenomena, i.e., diffraction, need to be modeled separately since ray tracing algorithms are inappropriate for sound propagation models at low frequencies. Exact methods to model diffraction are based on directly solving the acoustic wave equation using numeric methods like boundary or finite-element methods [30], [31], the wave-geometric approximation method [32], the Kresnel–Kirchoff approximation method [33], or the Biot–Tolstoy–Medwin (BTM) model [34] and its extension to higher order diffraction models [35].

Commonly used techniques to model diffraction with geometric acoustic methods are based on two models: the UTD [36] and the BTM model [34]. The BTM model is an accurate diffraction formulation that computes an integral of the diffracted sound along the finite edges in the time domain [31], [35], [37]. In practice, the BTM model is more accurate but is limited to noninteractive applications. The UTD model approximates an infinite wedge as a secondary source of diffracted sounds, which can be reflected and diffracted again before reaching the listener. UTD-based approaches have been effective for many real-time sound generation applications, especially in complex environments with occluding objects [23], [38]–[40].

Our approach, backward acoustic ray tracing, is motivated by these real-time simulations and proposes real-time source localization algorithm using ray tracing and UTD.

III. Acoustic Ray Tracing Handling Diffraction and Reflection

Motivation: After a source emits a sound, sound waves are propagated to free space and cause various interactions with obstacles; e.g., reflections occur after the sound wave hits obstacles, and diffractions arise at the boundary of the obstacles, such as an edge of wedges. While direct propagation paths are defined as paths propagating directly from a source to a listener without any interactions, a range of other interactions causes many types of indirect propagation paths of sound waves.

When the sound waves reach the microphone array through direct and indirect propagation paths, we can estimate the DoA, Θ^*, of sound waves using the DAS beamformer. However, we cannot determine whether the DoA came from a direct or indirect sound propagation path. Many beamformers have focused on estimating DoAs that came from direct propagation paths, but indirect, i.e., reflection and diffraction, propagation paths frequently occur. Especially, if the sound source becomes an NLOS source located in the invisible area for the microphone array, the indirect propagation path becomes a prominent path of sound propagation, and the beamformer cannot identify the DoA that came from direct propagation paths. Furthermore, beamforming techniques do not localize the source position in environments but compute only the DoA. Thus, a new type of SSL algorithm is needed to identify 3-D source positions.

Overview: In this article, we propose a novel SSL method that is a type of reflection and diffraction-aware SSL method. In indoor environments, there are many types of obstacles, e.g., walls, ceiling, and objects. They cause various interactions, i.e., direct, reflection, and diffraction, with sound waves, and a sequence of these interactions denotes a propagation path from a source to a measurement device. We want to estimate propagation paths considering reflection and diffraction using a ray tracing technique from signals measured by a microphone array, i.e., the eight-channel cube-shaped microphone array shown in Fig. 2. We then identify the positions of multiple sources based on the estimated propagation paths.

Before performing SSL at runtime, our method reconstructs the structures of an indoor environment, i.e., the surfaces and

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

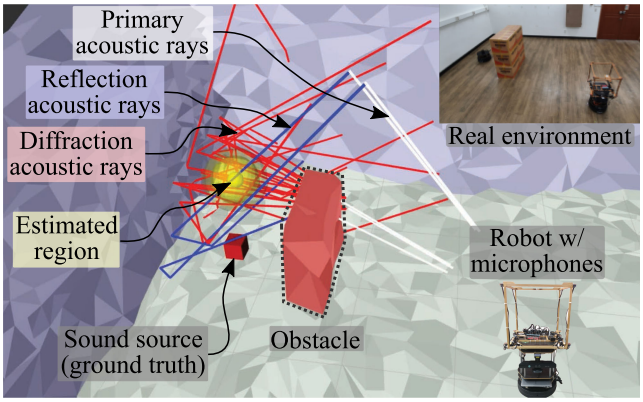


Fig. 1. Robot, equipped with a cube-shaped microphone array, localizes a source position in a 3-D space. Our formulation takes into account both direct and indirect sound propagation, given its use of acoustic rays. The acoustic rays are initialized and propagated based on our backward acoustic ray tracing algorithm that considers reflection and diffraction; primary, reflection, and diffraction acoustic rays are shown in white, blue, and red lines, respectively. The yellow disk, which is very close to the ground truth, represents a 95% confidence ellipse with regard to the estimated sound source, as computed by our approach.

wedges of objects, in order to handle the reflection and diffraction interactions of the sound waves (Fig. 3). Using a SLAM algorithm [41] with inertial measurement unit (IMU) and 3-D light detection and ranging (Lidar), we extract a registered point set representing the indoor environment. We then generate a mesh map using surface reconstruction techniques, i.e., screened poisson surface reconstruction [42] or simplification [43], from the registered point cloud. Our method uses the mesh map to estimate the reflection of sound propagation. Furthermore, our framework extracts wedges of objects to estimate the diffraction. We use a primitive fitting technique [44] to detect the wedges, given a voxelization map of the point cloud. We then extract the edges from the wedges of the primitive having no contact with the floor.

A runtime overview of our approach is shown in Fig. 2. First, we estimate incoming directions of the propagation paths using a DoA estimator (Section III-A). Our approach is basically built upon the DAS beamformer [8] of a cube-shaped eight-channel microphone array; it can be combined with different DoA estimators and microphone arrays (Section V-D), e.g., an eigenbeam-minimum variance distortionless response (EB-MVDR) beamformer of a 32-channel spherical microphone array [45], [46]. We then generate the acoustic rays considering both reflection and diffraction based on ray tracing techniques. Our acoustic ray tracing algorithm initializes a primary acoustic ray from each estimated DoA and propagates it through free space. If the acoustic ray hits a surface of obstacle, we generate a reflection acoustic ray by considering specular reflection (Section III-B). Additionally, when the acoustic ray satisfies the diffraction condition at a wedge, defined by the *diffractability*, the diffraction acoustic ray is generated based on a UTD model (Section III-C). Finally, the acoustic ray paths, a set of acoustic rays that originate from the same DoA, represent the estimated propagation paths of sound waves.

After generating the acoustic rays, we identify multiple source positions using acoustic ray paths (Section IV). Because the propagation paths of sound waves propagate from the sources to the listener, i.e., the measurement device in this case, the estimated propagation path represented by the acoustic ray path should pass through sound source positions. Acoustic ray paths can therefore converge to each source position, and we find the convergence regions of acoustic rays and determine these locations as multiple-sound-source positions.

A. Estimating the Direction of Arrival of Sound

Obstacles such as walls, ceiling, and objects cause various propagation paths in indoor environments, and different paths caused by the same source can propagate to the measurement device from different DoAs. Given the measured sound pressures of L samples in a single frame, it is necessary to estimate multiple DoAs, as there can be more than one DoA. Given this problem, we utilize a beamforming algorithm to estimate tuples containing a DoA Θ_n^* and its average beamforming power β_n over angular frequencies

$$[(\Theta_0^*, \beta_0), \ldots, (\Theta_N^*, \beta_N)] = \max_{\Theta}{}^{N} \left(\frac{1}{L} \sum_{\nu=0}^{L-1} \beta(\Theta, \omega_\nu) \right) \quad (1)$$

where $\max^N$ denotes the function of finding N tuples, (Θ_n^*, β_n) where $N = 4$, with large average beamforming powers, ω_ν is the νth angular frequency, and β is the beamforming power of the νth angular frequency at direction Θ; we refer beamforming formulas in [47]. We create 2562 points on the unit sphere from an icosahedral grid [48], and Θ is a specific direction $[\theta, \phi]$ corresponding to one of those points. We utilize a cube-shaped eight-channel microphone array with DAS beamformer [29]; however, our approach works properly with different types of microphone arrays and other beamformers as well (Section V-D).

We initialize a primary acoustic ray from a tuple (Θ_n^*, β_n). The primary acoustic ray r_n^0 is generated into the reverse direction of Θ_n^*

$$r_n^0(l) = \hat{d}_n^0 \cdot l + \dot{o} \quad (2)$$

where l is the ray length of a primary acoustic ray, $\hat{d}_n^0$ denotes the unit vector of the reverse direction of Θ_n^*, and $\dot{o}$ represents the origin of the microphone array. The superscript k of an acoustic ray r_n^k indicates the order of interactions, i.e., reflection or diffraction, along an acoustic ray path from the microphone array. For example, $r_n^0(l)$ indicates that there is no interaction and, thus, denotes a primary ray having the ray length l from the microphone array. All the other rays with a varying number of interactions, i.e., $k \geq 1$, are referred to as indirect acoustic rays with kth order interactions.

When the primary acoustic ray r_n^0 is generated in (2), the primary ray is initialized with initial energy of β_n, which represents the incoming power from the nth DoA. The energy of sound waves decreases with respect to the travel distance of the propagation path from the source to the listener and the absorption coefficient: $E(l) = E_0 \cdot 1/(1 + l^2) \cdot (1 - \alpha)^K$, where $E(l)$ is the energy when the sound wave propagates by distance l, E_0 denotes the initial energy of the sound waves at

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

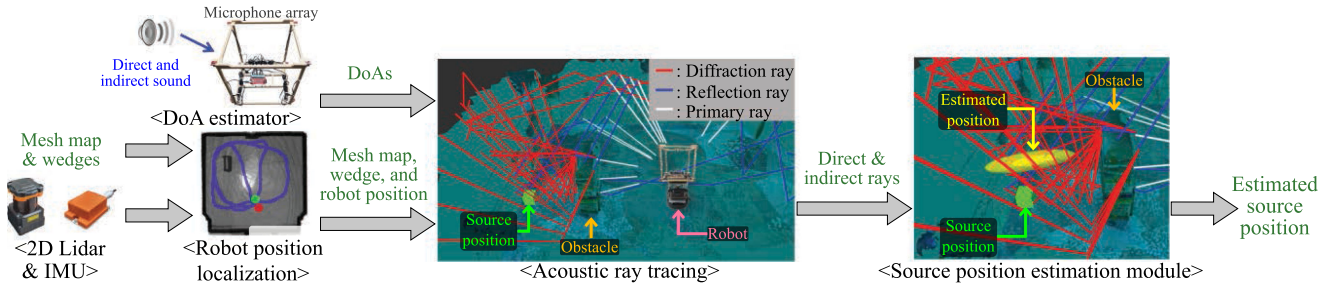


Fig. 2. Run-time computations using acoustic ray tracing for sound source localization. Acoustic ray tracing is performed from DoAs, a mesh map containing wedges, and a robot position where a DoA estimator works on a cube-shaped eight-microphone array. The robot position is estimated by 2-D SLAM from a 2-D Lidar sensor, and the mesh map and wedges are generated during the precomputation phase. Source position estimation is performed by identifying ray convergence from the generated acoustic ray paths.

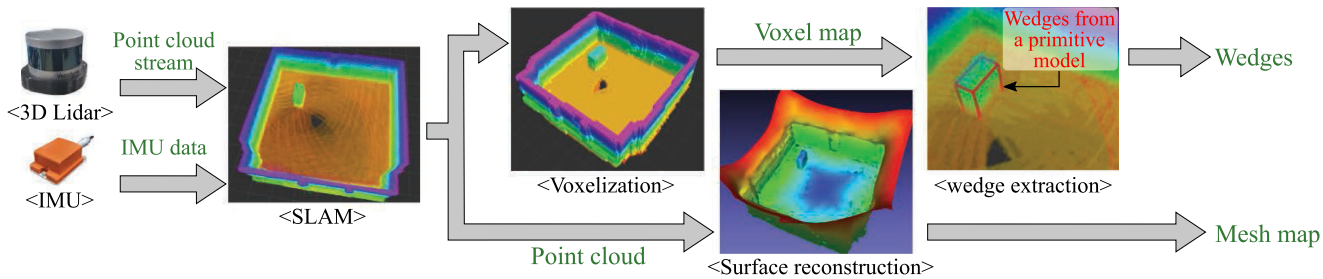


Fig. 3. Precomputation phase. We use SLAM to generate a point cloud of an indoor environment from IMU and 3-D Lidar, and the mesh map is reconstructed via surface reconstruction techniques. To extract the wedge information, we utilize voxelization from the point cloud and fit a primitive model, e.g., a box model in this case, onto the voxel map. Wedges are then extracted from the fitted primitive model. The extracted wedges of the fitted box model are highlighted by the red line.

the sound source, and α is the constant absorption coefficient, given the number of reflections K. Actually, the absorption coefficients depend on the material properties, but we assume that all materials have a constant coefficient, i.e., 0.1, since the majority of sound materials in our experiment have low absorption coefficient (Section V-C).

Because we consider backward acoustic ray tracing from a microphone array (listener) to a source, the initial power β_n should be amplified with respect to the ray length, and we can determine the maximum travel distance l_{max} of an acoustic ray path as follows:

$$l_{\text{max}} = \sqrt{\frac{\beta_{\text{th}}}{\beta_n}(1-\alpha)^K - 1}. \tag{3}$$

The propagation of the acoustic ray terminates when the power of the ray exceeds a user-defined threshold for maximum energy, denoted as β_{th}, which is set by a reasonable power bound, in this case 10^{-4}W, similar to the power of a loud alarm clock [49]: $0 \leq l \leq l_{\text{max}}$.

B. Acoustic Ray Tracing Handling Reflection

When an acoustic ray r_n^k hits a triangle of an object's mesh in the reconstructed environment, we need to simulate how the ray behaves at the hit point. Ideally, specular or diffuse reflection can occur with an energy absorption depending on the material type of the hitting surface. Since simulating all these types of interactions requires a prohibitive computation time, we support only a specular reflection in this work.

Our decision not to support diffuse reflections is based on the following two factors: 1) supporting diffuse reflections requires an expensive inverse simulation approach such as Monte Carlo simulation, which is unsuitable for real-time robotic applications, and 2) while there are many diffuse materials in rooms, each individual sound signal reflected from the diffuse material does not carry a high portion of the sound energy generated from the sound source. Therefore, when we choose high-energy directional data from the DoA estimator, the most sound signals reflected by the diffuse material are ignored automatically, and those with high energy are mostly from specular materials.

Note that our work does not require all the materials to be specular. When some materials exhibit high energy reflectance near the specular direction, e.g., tex materials in the ceiling and finished wooden floors, our method generates acoustic rays toward those specular reflection directions and can identify the location of the sound source that generates those rays (Section V-C). As a result, we focus on handling specular materials and treat each hit material as specular and generate a reflection ray from the hit point.

The operation for specular reflection is defined as follows. Whenever an acoustic ray r_n^k hits the surface of the obstacle at the particular ray length l_{hit}, we create a new, reflection acoustic ray r_n^{k+1} with the following direction:

$$r_n^{k+1}(l) = \hat{d}_n^{k+1} \cdot l + r_n^k(l_{\text{hit}}) \tag{4}$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

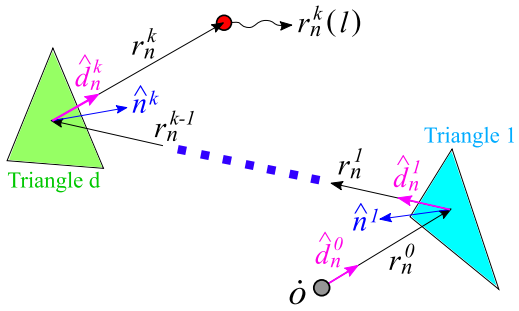


Fig. 4. Example of propagating reflection acoustic rays. The acoustic ray path containing direction and reflection acoustic rays from r_n^0 to r_n^k is propagated from the origin $\dot{o}$ of the microphone array to the red point corresponding to $r_n^k(l)$. The summation of all ray lengths l of each acoustic ray from r_n^0 to r_n^k should be identical to $l_{\max}$.

where $\hat{d}_n^{k+1}$ is the direction of the specular reflection of the ray r_n^{k+1} and is analytically computed by $\hat{d}_n^{k+1} = \hat{d}_n^k - 2(\hat{d}_n^k \cdot \hat{n}^{k+1})\hat{n}^{k+1}$, where $\hat{n}^{k+1}$ is the normal vector at the surface hit point, $r_n^{k+1}(0)$. Its example is shown in Fig. 4. The primary acoustic ray r_n^0 is initialized at the origin $\dot{o}_t$ of the microphone array at a t frame and hits the triangle 1. The reflection acoustic ray, then, is generated into the $\hat{d}_n^1$ direction considering specular reflection. The acoustic ray path is propagated into the kth order of reflection acoustic ray. The summation of all ray lengths of acoustic rays contained in the ray path should be the same as $l_{\max}$, given the power bound.

The reflection acoustic ray that we create can be reflected further by getting another hit on other obstacles. While generating the acoustic rays of a path, we maintain them in a ray sequence, called a ray path, $R_n = [r_n^0, r_n^1, \ldots]$ generated for the nth DoA.

C. Acoustic Ray Tracing Handling Diffraction

We now explain our algorithm to model diffraction efficiently within acoustic ray tracing. Since our goal is to achieve fast performance in localizing the sound source, we use the formulation based on UTD [36]. The incoming sound signals collected by the microphone array consist of contributions from different propagation paths in the environment, including reflections and diffractions.

Edge diffraction occurs when a sound wave hits the edge of a wedge. In the context of forward acoustic ray tracing from a source, when an acoustic ray hits an edge of a wedge, the diffracted signal propagates into all possible directions from that edge. The UTD model assumes that the point on the edge causing the diffraction is an imaginary source generating a spherical wave [36].

In order to solve the problem of localizing the sound source, we simulate the process of backward ray tracing from the microphone array to the source. Suppose that an nth DoA is generated by the diffraction at the point m_d on the wedge in Fig. 5(a). We generate the primary acoustic ray r_n^0 and perform backward acoustic ray tracing. In an ideal case, we can assume that the ray path R_n hits the point m_d on the edge of the wedge; for example, the ray r_n^{k-1} hits the point m_d and diffraction acoustic rays $r_n^{(k,\cdot)}$ must be generated in Fig. 5(a).

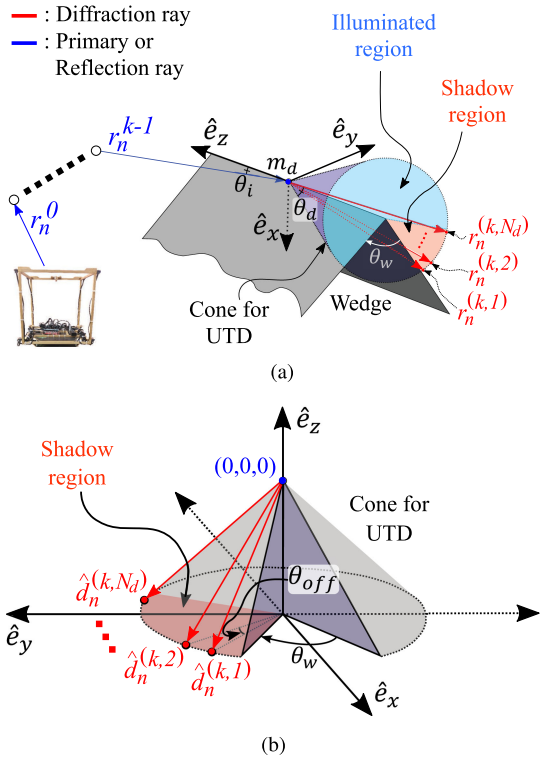


Fig. 5. Our acoustic ray tracing method devised to handle the diffraction effect. (a) Suppose that we have an acoustic ray r_n^{k-1} satisfying the diffraction condition, hitting or passing near the edge of a wedge. We then generate N_d diffraction rays covering the possible incoming directions (especially, in the shadow region) of rays that cause the diffraction. (b) Outgoing unit vector, $\hat{d}_n^{(k,p)}$, of a pth diffraction ray is computed on local coordinates $(\hat{e}_x, \hat{e}_y, \hat{e}_z)$ and used after transformation to the environment in runtime, where $\hat{e}_z$ fits on the edge of the wedge and $\hat{e}_x$ is set half-way between two triangles of the wedge.

We assume that the point m_d causing the diffraction is an imaginary source generating the spherical wave based on the UTD model. Given the diffracted propagation path estimated by the ray r_n^{k-1} in Fig. 5(a), there might be an infinite number of candidates for incident propagation paths to the point m_d causing the diffraction. Given that it is difficult to determine the specific direction $\hat{d}_n^k$ corresponding to the direction of an incident propagation path, to generate the kth order diffraction ray, we generate a set of N_d different diffraction rays which covers possible incident directions to the point m_d on the edge based on the UTD model. Intuitively, this set is generated based on the assumption that one of these generated rays may have the actual incident direction causing the diffraction, thus creating the subsequent ray r_n^{k-1}. When there are sufficient acoustic rays, including the primary, reflection, and diffraction rays, it is highly likely that those rays will pass through or close to the sound source location; we choose a proper value of N_d, which is 5, by analyzing diffraction rays (Section V-B).

Given the nth DoA caused by the diffraction, it is rare for acoustic rays of the nth DoA to intersect an edge precisely because our algorithm works in real environments containing various types of errors from sensor noise and resolution errors from the DoA estimator.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

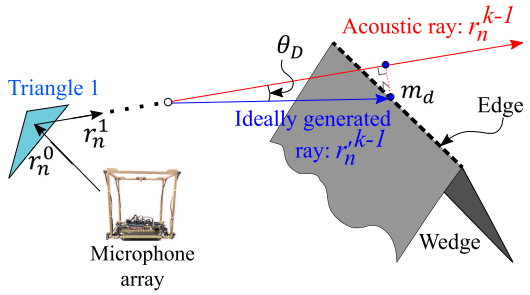


Fig. 6. Diffraction condition. When a ray r_n^{k-1} passes closely by an edge of a wedge, we consider the ray to be generated by edge diffraction. We measure and utilize the angle θ_D between the ray and its ideal generated ray that hits the edge exactly to verify our diffraction condition.

In order to support various cases that arise in real environments, we propose the use of the simple yet effective notion of a *diffraction condition* between a ray and a wedge. A diffraction condition simply measures how closely the ray r_n^{k-1} passes by an edge of the wedge. Specifically, we define the *diffractability* v_d according to the angle θ_D between the acoustic ray r_n^{k-1} and its ideally generated ray r'^{k-1}_n for the diffraction with the wedge, i.e., $v_d = \cos(\theta_D)$, where the cos function is used to normalize the angle θ_D (Fig. 6).

Suppose that the nth DoA is generated by the diffraction on the edge of the wedge, as highlighted by the dotted line in Fig. 6. In this case, the ray path R_n contains the diffraction propagation path, and we assume that the ideally generated ray r'^{k-1}_n represents the actual diffraction propagation path; the ray r_n^{k-1} does not hit the edge of the wedge due to the various errors that exist in real environments. We define the ideally generated ray r'^{k-1}_n as a ray touching the point m_d on the edge of the wedge and satisfying the smallest angle θ_D. To have the smallest angle θ_D, the distance from the point m_d to the ray r_n^{k-1} also becomes the smallest

$$m_d = \underset{m'_d}{\operatorname{argmin}}(\text{distance}(m'_d, r_n^{k-1})) \tag{5}$$

where m'_d is any point on the edge of the wedge, and the distance$(\cdot)$ denotes a minimum distance between the given point and line. The propagation direction of the ideally generated ray r'^{k-1}_n is identical to the vector from the origin of the ray r_n^{k-1} to the point m_d, and the angle θ_D can be computed using the inner product of the propagation directions of both rays r_n^{k-1} and r'^{k-1}_n.

If the diffractabilty v_d is larger than a threshold value v_{th}, e.g., 0.984 in our tests, our algorithm determines that the acoustic ray is generated from the diffraction at the wedge, and we, thus, generate the secondary diffraction ray at the wedge in the backward manner.

We now present how to generate the diffraction rays when an acoustic ray satisfies the diffraction condition. The diffraction rays are generated along the surface of the cone [Fig. 5(a)] because the UTD model is based on the principle of Fermat [25]; the ray follows the shortest path from the source to the listener. The surface of the cone for the UTD model contains every set of shortest paths. When an acoustic ray r_n^{k-1} satisfies the diffraction condition, we compute outgoing directions for those diffraction rays. Those directions are the unit vectors generated on that cone and can be computed on a local domain as shown in Fig. 5(b)

$$\hat{d}_n^{(k,p)} = \begin{bmatrix} \cos\left(\theta_w/2 + p \cdot \theta_{off}\right) \sin\theta_d \\ \sin(\theta_w/2 + p \cdot \theta_{off}) \sin\theta_d \\ -\cos\theta_d \end{bmatrix} \tag{6}$$

where $\hat{d}_n^{(k,p)}$ denotes the outgoing unit vector of a pth diffraction ray among N_d different diffraction rays, θ_w is the angle between two triangles of the wedge, θ_d is the angle of the cone that is the same as the angle between the outgoing diffraction rays and the edge on the wedge, and θ_{off} is the offset angle between two sequential diffraction rays, i.e., $\hat{d}_n^{(k,p)}$ and $\hat{d}_n^{(k,p+1)}$, on the bottom circle of the cone.

Given a hit point m_d by an acoustic ray r_n^{k-1} on the wedge, we transform the outgoing directions in the local space to the world space by aligning their coordinates $(\hat{e}_x, \hat{e}_y, \hat{e}_z)$. Based on those transformed outgoing directions, we then compute the outgoing diffraction rays, $\bar{r}_n^{(k)} = \{r_n^{(k,1)}, \ldots, r_n^{(k,N_d)}\}$, starting from the hit point m_d.

In order to accelerate the process, we only generate the diffraction rays in the shadow region, which is defined by the wedge; the outside of the shadow region is called the illuminated region. We focus on the shadow region because covering only the shadow region over the entire region generates minor errors for a simulation of the sound propagation [38].

Given the new diffraction rays, we apply our algorithm recursively and generate another order of reflection and diffraction rays. Given the nth DoA, we generate acoustic rays, including direct, reflection, and diffraction rays and maintain the ray paths R_n in a tree data structure. The root of this tree represents the primary acoustic ray, starting from the microphones. The depth of the tree denotes the order of its associated rays. Note that we generate one child and N_d children for handling reflection and diffraction effects, respectively.

We maintain the ray path R_n for the fixed duration D_{ray}, one second, to accumulate a sufficient number of ray paths; the ray path is deleted after the duration D_{ray}. The duration D_{ray} is determined to maintain a ray path caused by an early reflection until a late reflection, i.e., reverberation. If the duration D_{ray} is too long, our approach cannot properly reflect changes in the position of a moving sound source.

IV. Monte Carlo Localization for Multiple Sources

In the prior section, we generated primary, reflection, and diffraction acoustic rays starting from DoAs. Given those acoustic ray paths, we are ready to localize not only stationary sound sources but also moving sound sources in 3-D space; our approach utilizes all ray paths created within the fixed time duration D_{ray}.

The generated acoustic ray paths represent the propagation paths of sound waves from sound sources to the microphone array. In an ideal case with multiple sources, it is sufficient to find points at which acoustic ray paths intersect and treat them as source positions. However, when we deal with real environments

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

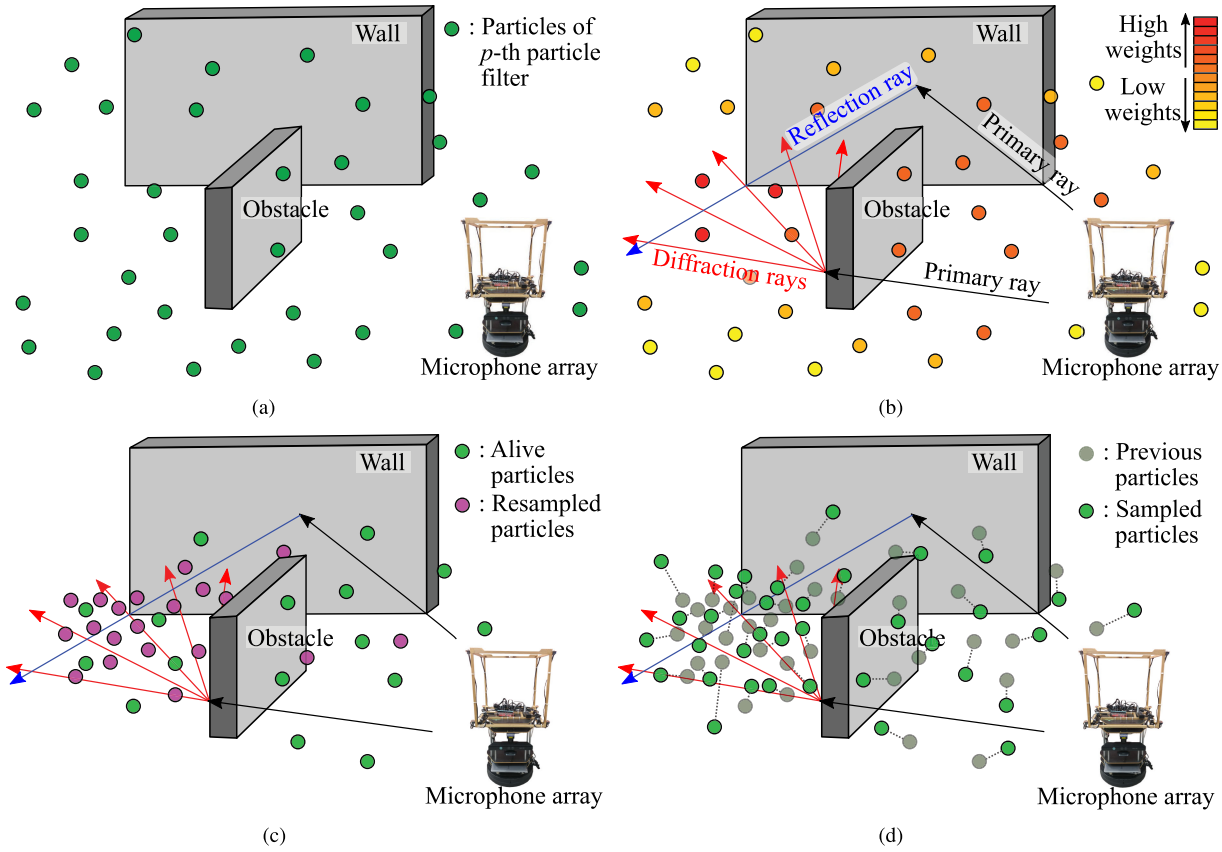


Fig. 7. Example of performing the pth particle filter at the first and second iterations, i.e., $t = 0$ and $t = 1$. At the beginning of our approach, i.e., $t = 0$, particles are initialized based on the uniform distribution in (a). In the weight computation part (b), weights of particles are computed, given acoustic ray paths; particles have higher weights when they are located near the convergence region of ray paths. In the resampling path (c), particles with low weights are resampled close to particles with high weights. Thanks to the resampling part, particles can be moved to the convergence region of ray paths. After executing the part of allocating ray paths (Section IV-D), the first iteration of our approach is finished. At the second iteration, i.e., $t = 1$, the Monte Carlo localization starts with the sampling part, and particles are regenerated based on the Gaussian distribution in (d).

in practice, acoustic ray paths may not intersect precisely, as there are diverse types of noise from sensors, e.g., microphones, IMU sensors, and Lidars. We thus need a technique that is robust to these types of noise. We cast our problem as one involving the locating of regions where many such ray paths converge and treat the convergence regions as candidate regions containing sound sources. To achieve our goal, we propose the use of MCL [13], [50], also known as a particle filter.

Sasaki *et al.* [13] proposed the source localization method based on a particle filter from estimating the convergence regions of the plane observation models, which contains direct sound information. We extend this approach to identify the convergence region of acoustic rays; the prior approach needs to satisfy some constraints, i.e., accumulating the observation models in different positions and orientations of a sound sensor, but our approach does not require those constraints by considering indirect sound.

Assuming there are P sound sources, there can be P different convergence regions of ray paths. The pth convergence region corresponds to the pth sound source, and ray paths propagating to the pth convergence region can be caused by the pth sound source. In the acoustic ray tracing phase, it is difficult to determine what acoustic ray paths are generated by which sound sources. In every iteration of our localization algorithm, therefore, we initially estimate the source positions using multiple particle filters and then determine whether or not the ray paths are caused by estimated sources.

Our approach supports P different particle filters to localize P sound sources, and each particle filter can localize only a single source. Each particle filter consists of four parts and sequentially performs them every iteration. These are sampling, weight computation, resampling, and allocating ray paths. In the first three parts, particle filters identify the convergence regions of ray paths; Fig. 7 shows an example of executing three parts sequentially. In the sampling part, our approach initializes [Fig. 7(a)] or regenerates the positions of particles randomly close to previous positions [Fig. 7(d)] to consider the movement of dynamic sources. The weights of particles are computed to ensure that the particles converge to the convergence region of the ray paths [Fig. 7(b)]. In the resampling part, our approach deletes particles located far from the convergence region and resamples new particles inside the convergence region [Fig. 7(c)].

In the last part of our method, if there are convergence regions, acoustic ray paths are allocated to each convergence region into which they propagate. The ray paths generated from the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

ray tracing phase are initially not allocated to any convergence region; they have an initial state of -1, i.e., $S(R_n) = -1$, where R_n denotes the nth ray path and $S(\cdot)$ is the function returning the state value of a given ray path. After acoustic ray paths are allocated to the convergence region of the pth particle filter, it has a p state, i.e., $S(R_n) = p$.

A. Sampling

To identify the convergence region of ray paths, the pth particle filter maintains a set of I particles, $X_t^p = [x_t^{(p,1)}, \ldots, x_t^{(p,I)}]$, which serves as hypothetical locations of a sound source; the number of particles, e.g., $I = 200$, should be sufficient to cover all 3-D indoor environments. Those particles are spread out randomly in the 3-D space based on the uniform distribution at the initial iteration, $t = 0$, and iteratively move to the convergence region of the ray paths at other iterations, $t \geq 1$. Also, if there is no ray path at a specific iteration, we treat that there does not exist any sound source and perform the initialization process, i.e., spreading out random particles, again to quickly cover the entire 3-D space.

To consider the movement of sound sources, a new set of particles X_t is incrementally created from the prior particles X_{t-1} for each iteration t other than the initial iteration. If we know a source position and its velocity, new particles can be created using the source velocity and the corresponding moving direction. However, in an SSL problem, we do not know the position or velocity of a source when our approach begins. Therefore, we randomly create a new set of particles from the prior particles and then regenerate particles near the actual source position in the ensuing weight computation and resampling parts.

A new particle $x_t^{(p,i)}$ of the pth filter is generated by offsetting a previous one $x_{t-1}^{(p,i)}$ in a random unit direction $\hat{u}$ by an offset δ

$$x_t^{(p,i)} = x_{t-1}^{(p,i)} + \delta \cdot \hat{u},$$
$$\delta = \|x_t^{(p,i)} - x_{t-1}^{(p,i)}\| \sim N(0, {\sigma_s}^2) \tag{7}$$

where $N(\cdot)$ denotes a normal distribution with a zero mean and standard deviation. Actually, the random unit direction $\hat{u}$ and the offset δ correspond to the unit vector of the velocity and the speed of the particle, respectively. Since the offset δ is sampled according to the normal distribution, (7) can cover the various movements of the stationary, constant velocity, and accelerated particles. The standard deviation σ_s is determined by the maximum speed of a moving sound source, e.g., $\sigma_s = 0.2$ m in our experiments. Our approach is designed to handle speeds up to 1 m/s, $= 0.2$ m$/0.2$ s, of moving sources, where the iteration period is 200 ms.

B. Weight Computation

We associate a weight with each particle, and the weight indicates the importance of the particle, specifically encoding how closely the particle is located to a convergence region of ray paths. Suppose that ray paths are converged in a region containing the source position. In this case, the distances from any point inside the convergence region to the ray paths must be small; in an ideal case, ray paths intersect at a certain point, and distance between an intersecting point to a ray path should be zero. Therefore, when a particle is located inside the convergence region of ray paths, the distances between the particle and the ray paths are generally short. Based on these distances, we design the weight of the particle to have a higher value when it is located inside the convergence region.

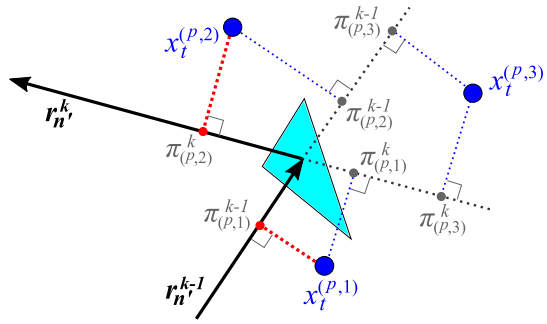


Fig. 8. Example of computing weights of the pth filter for particles against a ray path, $R_{n'} = [\ldots, r_{n'}^{k-1}, r_{n'}^{k}]$. The shortest distances for each particle over acoustic rays are shown in red and become the distances between the particles and the ray path.

The weight at the t iteration is also updated from the previous iteration $t-1$, assuming that the sound source is active from $t-1$ to t iterations; how our method handles intermittent sources is discussed later in this section. During the weight computation phase, the pth particle filter only considers ray paths in the -1 or p state. The -1 state means that the ray path is not yet allocated to any estimated source position. The p state indicates that the ray path propagated close to the estimated source position in a prior iteration and was therefore allocated to the sound source estimated by the pth particle filter. We ignore the remaining ray paths with other states. Therefore, in the pth particle filter, the weight of the particle is computed based on the observations o_t^p consisting of the acoustic ray paths $R_{n'}$ in only the -1 or p state.

The distance between a particle and an acoustic ray can be computed by calculating the distance from a point to a line segment; the distance between a point to a line segment corresponds to the distance from a point to a perpendicular foot on a line segment. We define the distance $\mathrm{dist}(\cdot,\cdot)$ between a particle and a ray path $R_{n'}$ as the shortest distance among the distances from the particle to the rays of $R_{n'}$

$$\mathrm{dist}(x_t^{(p,i)}, R_{n'}) = \min_k(\|x_t^{(p,i)} - \pi_{(p,i)}^k\| \times F(x_t^{(p,i)}, r_{n'}^k)) \tag{8}$$

where $\pi(x_t^{(p,i)}, r_{n'}^k)$, in short, $\pi_{(p,i)}^k$, defines the perpendicular foot of the particle $x_t^{(p,i)}$ to the ray $r_{n'}^k$ (Fig. 8), and $\|\cdot\|$ denotes the L2 norm. F is a filter function returning infinity to exclude irrelevant cases when the perpendicular foot is outside of the ray segment $r_{n'}^k$, e.g., $\pi_{(p,1)}^k$, $\pi_{(p,2)}^{k-1}$, $\pi_{(p,3)}^{k-1}$, and $\pi_{(p,3)}^k$ in Fig. 8. Otherwise, the filter function returns one.

Based on the distance between the particle $x_t^{(p,i)}$ and the ray path $R_{n'}$, we define the probability density $P(R_{n'}|x_t^{(p,i)})$:

$$P(R_{n'}|x_t^{(p,i)}) = N(\mathrm{dist}(x_t^{(p,i)}, R_{n'}) \mid 0, \sigma_w^2) \tag{9}$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

where $N(\cdot|0,\sigma_w^2)$ indicates a normal distribution with a zero mean and standard deviation, representing a parameter that controls how many particles converge to the estimated source position; a smaller standard deviation makes particles converge to a smaller area, meaning that the estimated convergence region of ray paths also becomes smaller. The standard deviation of σ_w only has a minor effect on the accuracy. We use $\sigma_w = 0.1$, corresponding to 0.1 m in the space, for our tests.

Simply speaking, the probability density $P(R_{n'}|x_t^{(p,i)})$ becomes higher if the ith particle $x_t^{(p,i)}$ is closer to the ray path $R_{n'}$, and $P(R_{n'}|x_t^{(p,i)})$ has the highest value if the particle lies on the ray path. From the probability density $P(R_{n'}|x_t^{(p,i)})$, we design the likelihood $P(o_t^p|x_t^{(p,i)})$ of the particle $x_t^{(p,i)}$, where the observation o_t^p consists of N' different ray paths having -1 and p states: $o_t^p = [R_1, \ldots, R_{N'}]$. We define the likelihood as the average of $P(R_{n'}|x_t^{(p,i)})$ over all ray paths

$$P(o_t^p|x_t^{(p,i)}) = \frac{1}{N'}\sum_{n'=1}^{N'} P(R_{n'}|x_t^{(p,i)}). \tag{10}$$

The likelihood $P(o_t^p|x_t^{(p,i)})$ indicates how much the particle $x_t^{(p,i)}$ is close to ray paths contained in the observation o_t^p.

We define the weight $w_t^{(p,i)}$ at the t iteration based on the likelihood $P(o_t^p|x_t^{(p,i)})$

$$w_t^{(p,i)} = \frac{P(o_t^p|x_t^{(p,i)})w_{t-1}^{(p,i)}}{n_c} \tag{11}$$

where n_c denotes the normalization constant and $w_{t-1}^{(p,i)}$ is the weight at the previous iteration $t-1$. The weight $w_{t-1}^{(p,i)}$ helps to consider the convergence region at the previous iteration $t-1$. When the particle $x_{t-1}^{(p,i)}$ is close to the convergence region at the previous iteration $t-1$, the weight $w_{t-1}^{(p,i)}$ should be large, causing the weight $w_t^{(p,i)}$ to increase. If there is no acoustic ray at iteration t, we set all weights to a uniform probability, i.e., $1/I$.

Suppose that an intermittent source is activated at iteration t, while it was inactive at the previous iteration $t-1$. At iteration $t-1$, no acoustic ray was generated for the intermittent source. Despite the fact that other active sources exist at iteration t, the ray paths generated by those sources were previously allocated to other filters identifying the convergence regions of those sources. As a result, ray paths in the -1 state are left for the intermittent source. Suppose that the pth particle filter corresponds to the source. All weights of particles of the pth particle filter have a uniform probability as the initialization process (Section IV-A). At iteration t, newly generated acoustic rays in the -1 state should be propagated to the activated source, and the weight $w_t^{(p,i)}$ is only determined by the distance between the particle and acoustic rays.

C. Resampling

There may be particles close to or far from the convergence region of ray paths, and their weights indicate how closely they are located to the convergence region. To make particles converge to the convergence region of ray paths in this part, we delete particles located far from the convergence region and regenerate them inside the convergence region. Intuitively, particles with low weights are removed, and additional particles are generated near existing particles with high weights. Regarding this process, we adopt a basic resampling method [50].

Once resampling is done, we check whether the particles are converged enough to define an estimated sound source; if the particles are thus converged, we can treat the positions of particles as the convergence region of the ray paths. To determine the convergence of the particle positions, we compute the generalized variance (GV), which is a one-dimensional measure for multidimensional scatter data and is defined as the determinant of the covariance matrix of the particles [51]. If GV is less than the convergence threshold, $\sigma_c = 0.01$, at the pth particle filter, we determine that the source emitted the sound and treat the mean position of particles as the estimated position of the source. GV is also used as a confidence measure in our estimation; we use its covariance matrix to draw a 95% confidence ellipsis disk for visualizing the estimated sound region (Fig. 1).

D. Allocating Ray Paths

Suppose that the sound source is estimated in the resampling step. In such a case, it becomes necessary to check whether or not ray paths are caused by the estimated source; if there is no estimated source, we skip the allocating ray paths phase. Assuming that a ray path is caused by the estimated source, it should propagate to the position of the estimated source. In this step, we only consider ray paths in the -1 state, indicating that we do not know from which sound sources the ray path originated. We now verify whether the ray paths in the -1 state propagate close to the estimated source position.

A simple way to do this is to compute and verify the distances between the estimated source position, i.e., the mean of the particle positions, and the ray paths. However, this simple approach does not consider the shape of the estimated sound region in Fig. 1, which represents the 95% confidence area. To deal with the shape of the estimated sound region, we examine the relationships between the ray paths and particle positions.

We define the probability, $P(S(R_{n'}) \to p)$, of allocating the ray path to the source estimated by the pth filter as follows:

$$P(S(R_{n'}) \to p) = \sum_{i=1}^{I} P(R_{n'}|x_t^{(p,i)})w_t^{(p,i)} \tag{12}$$

where $R_{n'}$ is the ray path in the -1 state, $P(R_{n'}|x_t^{(p,i)})$ is the probability density in (9), and $w_t^{(p,i)}$ is the weight of a particle as defined by (11). We allocate the ray path $R_{n'}$ to the sound source estimated by the pth filter if the probability $P(S(R_{n'}) \to p)$ exceeds a threshold probability, i.e., $P_{\text{th}} = 0.2$.

The probability density $P(R_{n'}|x_t^{(p,i)})$ represents how close the particle $x_t^{(p,i)}$ is to the ray path $R_{n'}$, and the weight $w_t^{(p,i)}$ indicates the importance of the particle. If the particle is located close to the estimated source, its weight becomes high, and it must be an important particle. Therefore, if many particles with

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

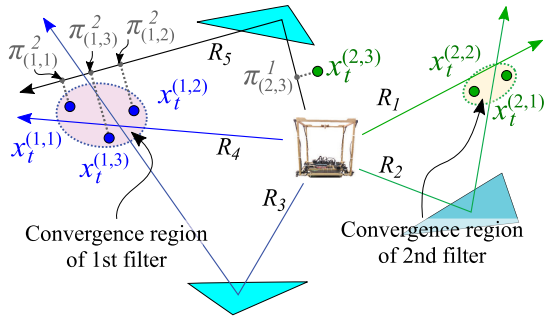
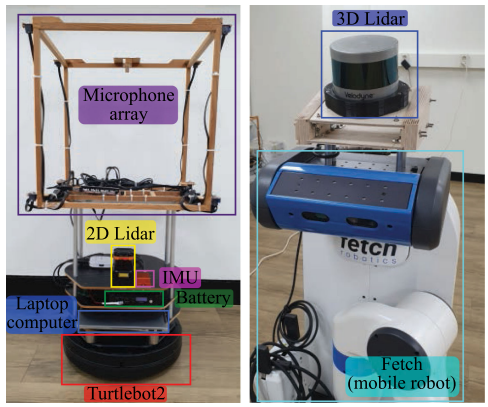


Fig. 9. Example of allocating the ray path to the convergence region of the particle filter. The ray paths, indicated here by the blue and green lines, are allocated to the convergence regions of the first and second particle filter, respectively; both convergence regions represent the estimated source positions. Ray path R_5, indicated by the black lines, is now considered to be assigned to its proper estimated source. Gray dotted lines denote the distance between the particles and ray path R_5, used to compute the probability $P(R^p_{n'}|x_t^{(p,i)})$ in (11). In this example, ray path R_5 originates from the source estimated by the first filter, and it is allocated to the estimated source of the first particle filter. The allocating probability $P(S(R_5) \rightarrow 1)$ exceeds the threshold probability P_{th}.



(a) (b)

Fig. 10. Hardware platforms of our approach. (a) To utilize our SSL algorithm in the runtime computation, we add an eight-channel microphone array onto Turtlebot2, a mobile robot, with 2-D Lidar, an IMU sensor, and a laptop computer. (b) In the precomputation phase, we extracted the point cloud of the environments using 3-D Lidar placed on the top of the Fetch mobile robot.

high weights are located close to the ray path $R_{n'}$, the allocating probability $P(S(R_{n'}) \rightarrow p)$ becomes higher than the threshold probability P_{th}. Fig. 9 shows an example of allocating the ray path.

We continue these iterations of our MCL algorithm, consisting of four parts given a fixed duration, e.g., 200 ms, until our multiple-source localization algorithm, containing acoustic ray tracing and MCL algorithms, is terminated. We decide to make the period of iterations short enough to find the source position quickly, taking into account the calculation time of our approach. If our MCL algorithm is finished within the computation budget, 200 ms, it enters an idle state until the next iteration.

V. Results and Discussion

In this section, we provide various results and discussions of our approach. The hardware platform is based on Turtlebot2 with a 2-D Lidar (UTM-30LX of Hokuyo), an IMU sensor (MTi-30 of Xsens), an eight-channel microphone array [52], and a laptop computer with an Intel i7 process shown in Fig. 10(a).

To estimate DoAs, we utilize a DAS beamforming module of ManyEars [53], which is a real-time open software for the DoA estimation, tracking, and separation. Although ManyEars tracks DoA information using the particle filter, the processes between particle filters of ManyEars and our approach are different; we just refer to the process of ManyEars and only utilize the DoA estimator, i.e., the DAS beamformer. ManyEars performs the particle filter given DoAs and energies of DoAs to track the DoA sequentially, but our approach performs the particle filter given acoustic rays to identify convergence regions of those rays.

The DoA estimator and acoustic ray tracing algorithm are performed every 10.67 ms since the sampling frequency of the audio stream of the microphone array is 48 000 Hz, and the number of sound pressure samples is 512, i.e., $L = 512$; 10.67 ms = 512 samples 48 000 Hz. For all computations, we use a single core and perform our estimation within every 200 ms, supporting five different estimations in 1 s.

Our experiments contain dynamic sound sources. To make a sound source move, we utilize the mobile robot platform, i.e., Turtlebot2, and the sound sources, an omnidirectional speaker, are placed on the mobile robot. We also measured the odometry of the mobile robot, which contains the sound source, and then utilized measured odometry as the ground truth of moving sources.

To reconstruct the 3-D environments, we perform a SLAM algorithm, i.e., Cartographer of Google [41], using sensor data collected by a 3-D Lidar (VLP-16 of Velodyne) and an IMU sensor equipped on Fetch [54] [Fig. 10(b)]. We also utilize the open source, MeshLab [55], to generate mesh maps from point clouds and improve the quality of meshes.

Wedges needed for supporting diffraction effects are extracted by using primitive fitting techniques [56], where the primitive model is defined by the box shape since our experiments contain only box-shape obstacles. We expect that different shapes of obstacles can be identified using various primitive models [44], [57].

Benchmarks: We tested our approach in various scenarios and compared our result to the prior work [1], i.e., ManyEars3D. This method is another version of ManyEars [53], i.e., the open software containing the DAS beamforming module that we utilize. While ManyEars contains a module for estimating DoAs, this method, i.e., ManyEars3D, provides a module for estimating 3-D locations of sound sources. While ManyEars3D can identify 3-D locations of the source, it considers only the direct sound; it estimates the source position by considering direct sound based on the DAS beamformer and then tracks estimated source positions using a particle filter.

We first conducted a room experiment having 7 m $\times$ 7 m area and 3-m height with a moving source (Section V-A). In this environment, we verify how well our approach identifies a source position given a direct and reflection acoustic rays. We also place an obstacle, blocking direct sound propagation paths, to show the effect of diffraction acoustic rays where the sound

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE I
QUANTITATIVE RESULTS OF SINGLE-SOURCE SCENARIOS

Environments (Fig. 11 and 15)	w/o obstacle	w/o obstacle	w/ obstacle	w/ obstacle	absorp. mat.		EB-MVDR w/ 32-ch array	
					w/o obstacle	w/ obstacle	w/o obstacle	w/ obstacle
Source signal	clapping	speech	clapping	speech	clapping	clapping	clapping	clapping
Avg. distance error / std. of prior work	1.6 m / 0.6776	1.7769 m / 0.7615	1.582 m / 0.758	1.7571 m / 0.8112	1.7724 m / 0.7767	1.6324 m / 0.7353	- / -	- / -
Avg. distance error / std. of ours **(improvement over the prior work)**	0.5967 m / 0.2617 **(168 %)**	0.7416 m / 0.4448 **(139 %)**	0.6351 m / 0.3698 **(149 %)**	0.7313 m / 0.2679 **(140 %)**	0.6176 m / 0.2106 **(186 %)**	0.6998 m / 0.3858 **(133 %)**	0.5946 m / 0.2392 (-)	0.6176 m / 0.3165 (-)
Avg. significant primary / reflection / diffraction rays of LOS source	6.49 / 10.43 / 0	8.84 / 7.23 / 0	7.36 / 9.52 / 2.32	6.9 / 9.79 / 1.15	7.14 / 11.32 / 0	7.53 / 10.67 / 1.58	6.93 / 11.2 / 0	8.8 / 8.58 / 0.26
Avg. significant primary / reflection / diffraction rays of NLOS source	- / - / -	- / - / -	0.61 / 9.3 / 3.87	1.55 / 3.83 / 6.39	- / - / -	0.83 / 8.3 / 4.67	- / - / -	0.97 / 5.43 / 3.18

source moves around the obstacle. We then analyze diffraction acoustic rays to confirm the benefits of them (Section V-B).

In the above environments, the majority of objects, e.g., wall, floor, and ceiling, consists of specular materials like bricks, thick woods, gypsum boards, and steals; specular materials reflect the most of the sound energy from incident sound waves. Those specular materials cause sufficient specular reflections helping to generate reflection acoustic rays, whereas diffuse materials absorb most of the energy; therefore, specular reflection does not occur well at those diffuse materials. By replacing specular materials by diffuse materials, we also test the robustness of our approach where the area of specular materials decreases (Section V-C); decreasing the number of specular materials means decreasing the number of reflection propagation paths. We also tested our approach with a different DoA estimator using a different microphone array (Section V-D). The quantitative results of scenarios of a single moving source are summarized in Table I.

Because we extend the single SSL to the multiple source localization algorithm in this article, we conducted experiments with multiple sources. We tested our approach in two scenes containing multiple static sound sources, which do not move, or moving sound sources (Section V-E); especially, in the three static source scenes, we show localizing intermittent sound sources by controlling the source activation period. The quantitative results of scenarios of multiple sources are summarized in Table II.

To show the robustness for different sizes of environments, we conducted the experiments in a smaller size of the room: 7 $\times$ 3 m area with 3-m height (Section V-F).

We apply our SSL algorithm to the navigation task. When the source is located behind the obstacle, i.e., the NLOS source, the robot equipped with a microphone array estimates the source position using our approach and navigates to the estimated source position corresponding to the goal position of the navigation (Section V-G).

TABLE II
QUANTITATIVE RESULTS OF MULTIPLE SOURCE SCENARIOS

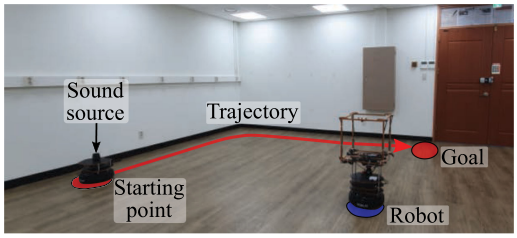
Scene	Two stationary sources	three stationary sources	two moving sources (Fig. 20(a))
Source 1 Avg. distance error / std. of prior work	1.6712 m / 0.6436 (clapping)	1.3286 m / 0.6328 (speech)	1.36 m / 0.6663 (clapping)
Source 2 Avg. distance error / std. of prior work	1.6662 m / 0.6393 (speech)	1.717 m / 0.7958 (clapping)	1.812 m / 0.7043 (speech)
Source 3 Avg. distance error / std. of prior work	- / -	1.0551 m / 0.5016 (clapping)	- / -
Source 1 Avg. distance error / std. of ours **(improvement over the prior work)**	0.5947 m / 0.3452 (clapping, **181 %**)	0.4263 m / 0.271 (speech, **211 %**)	0.7689 m / 0.4629 (clapping, **76 %**)
Source 2 Avg. distance error / std. of ours **(improvement over the prior work)**	0.4306 m / 0.1895 (speech, **286 %**)	0.4856 / 0.1488 (clapping, **246 %**)	0.7246 / 0.2843 (speech, **150 %**)
Source 3 Avg. distance error / std. of ours **(improvement over the prior work)**	- / -	0.5185 / 0.2787 (clapping, **103 %**)	- / -

A. Moving Source w/ or w/o an Obstacle

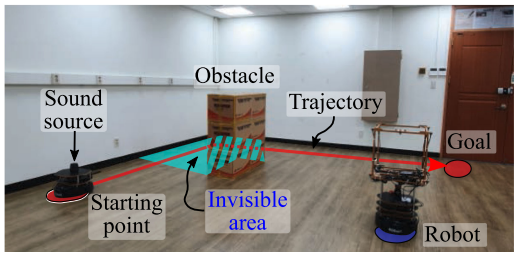
We first show results of the environments with a moving source without or with an obstacle. The sound source moves along trajectories, red lines shown in Fig. 11, and emits sound signals. We utilize two kinds of sound signals that are a clapping sound and a human speech; the dominant frequencies of a clapping sound and a human speech are 15 kHz and 275 Hz, respectively. The clapping sound consists of five claps, and the human speech is reading the sentence "Hey, robot, come here" by a woman.

The environment without an obstacle: The results of the environment without an obstacle [Fig. 11(a)] are shown in Fig. 12. We measure distance errors between the ground truth and estimated source positions, and the smaller distance error means that the accuracy is higher. The average distance errors of the clapping sound and the human speech are 0.5967 and 0.7416 m, respectively; we call the average of distance errors as the average distance errors for convenience. Note that both values are smaller than the average distance errors, i.e., 1.6 and 1.7769 m of the clapping sound and the human speech, of a prior

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

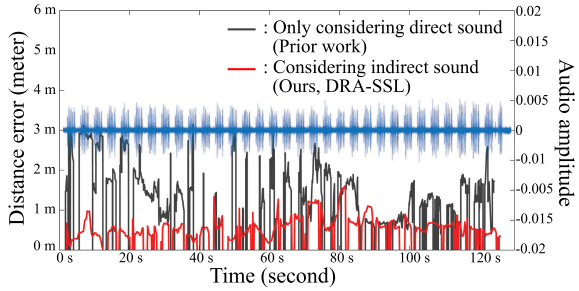


(a)

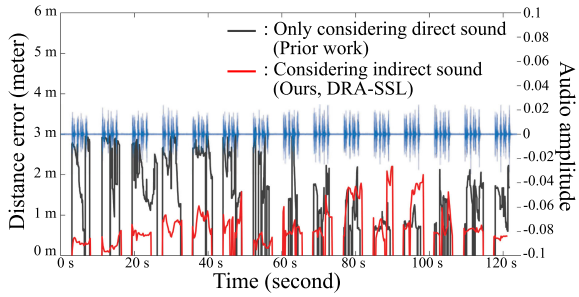


(b)

Fig. 11. Testing environments in a 7 m × 7 m room with a 3-m height given one moving source w/ and w/o an obstacle. (a) Environment without an obstacle and where the sound source moves along the trajectory, highlighted by the red line. (b) Environment with an obstacle, i.e., the box shape, where the moving source becomes a nonline-of-sight source when it is located in the invisible area due to the box.



(a)



(b)

Fig. 12. Results in the environment without an obstacle [Fig. 11(a)], where the clapping sound is used in (a) and human (female) speech is used in (b). Both show the distance error of our approach and prior work [1] in the red and gray curves, respectively, between the ground truth and the estimated source positions, and the measured signals in blue curves of the clapping sound in (a) and the human speech in (b).

work, and, thus, our SSL algorithm can identify the source position reasonably well in this case. We observe a 168% and 139% improvement for the clapping sound and the human speech. The prior work only considers the direct sound, and, thus, the better accuracies of our approach show that it is useful to consider the reflection sound. Also, the reason why the average distance error of the human voice is worse than the clapping sound is that the dominant frequency of the human voice (275 Hz) is lower than that of the clapping sound (15 kHz); the lower frequency sound more frequently causes the diffuse reflection, i.e., scattering by obstacles, rather than the specular reflection [58].

To verify how much the acoustic rays contribute in terms of helping source localization, we check how many ray paths propagate near to the source position. For example, given a ray path consisting of various acoustic rays, we find the smallest distance of acoustic rays contained by a ray path; the distance between the source position and the acoustic ray corresponds to the distance between a point and a line segment. If the smallest distance is less than 1 m, we treat this ray path as helping the source localization. We then check the type of the acoustic ray having the smallest distance and count this ray; we call those rays the significant ray. The average numbers of significant primary and reflection rays per frame are 6.49 and 10.43 of the clapping sound and 8.84 and 7.23 of the human speech. The diffraction acoustic rays are not generated because the DoA estimator can only detect prominent propagation paths; a diffraction propagation path becomes a prominent propagation path when the source is the NLOS state in other tested environments.

These results tell us that sufficient primary and reflection rays propagate near the source position. Moreover, the number of significant reflection rays of the clapping sound, i.e., 10.43, is larger than the human speech, i.e., 7.23, because the specular reflection frequently occurs on the higher frequency sound. The larger number of significant reflection rays helps to increase the localization accuracy since reflection rays increase the convergence of rays; the average distance error of the clapping sound is less than the human speech. The propagation directions of primary rays should be similar since they are generated at the robot to the source position, while the propagation directions of reflection rays are determined by a normal vector of a hit obstacle.

The environment with an obstacle: In the prior experiment without an obstacle, there is a sufficient number of significant primary and reflection rays, and we can localize the moving source by utilizing the primary and reflection acoustic rays where the diffraction propagation path was not a prominent path thus not detected by the DoA estimator. However, we need to consider diffraction on the wedges of the obstacle, especially when an obstacle is located and blocks the direct propagation path of sound, as shown in Fig. 11(b); the size of the obstacle is 0.39 m × 0.96 m area with 1.05-m height. The diffraction propagation paths become prominent, when the moving source is located in the invisible area: the source in this case becomes the NLOS source.

We present the results of the environment with an obstacle in Fig. 13; we tested with the prior work and two versions of our approach: the first version is only utilizing primary and reflection acoustic rays, and the second version is adding diffraction rays to them. We call the first version as reflection-aware SSL (RA-SSL) and the second version as diffraction- and reflection-aware SSL (DRA-SSL) for convenience.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

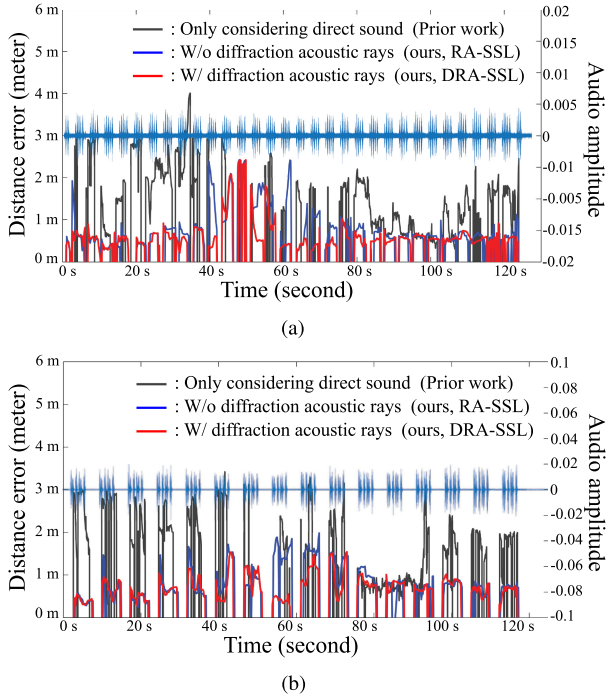


Fig. 13. Results in the environment with an obstacle [Fig. 11(b)] and two sound signals: the clapping sound and human speech. In both (a) and (b), the black curves are the distance errors of the prior work [1], the blue curves are the distance errors where we use only the primary and reflection acoustic rays (RA-SSL), and the red curves correspond to the distance errors when handling all types of acoustic rays containing diffraction acoustic rays (DRA-SSL). Measured audio signals are shown in the middle of the graphs.

When we use the clapping sound, the average distance errors are 0.6351 (DRA-SSL), 0.8112 (RA-SSL), and 1.582 m (prior work) in Fig. 13(a). When we utilize the human speech, the average distance errors are 0.7313 (DRA-SSL), 0.8803 (RA-SSL), and 1.7571 m (prior work) in Fig. 13(b). These results show that the diffraction acoustic rays help to localize the source better. We observe a 149% (clapping sound) and 140 % (human speech) improvement over the prior work considering only direct sound, and a 15% (clapping sound) and 18% (human speech) improvement when we additionally consider the diffraction rays compared to only utilizing the primary and reflection rays.

Especially, when the sound source becomes an NLOS source, located in the invisible area in Fig. 11(b), from 20 to 80 s, the average distance errors when adding diffraction rays, i.e., DRA-SSL, are 0.7336 m for the clapping sound and 0.7618 m for the human speech, while the average distance errors of the prior work and RA-SSL are 2.1515 (prior work, clapping sound), 2.3 (prior work, human speech), 0.7336 (RA-SSL, clapping sound), and 0.7618 m (RA-SSL, human speech), respectively. We observe a 193% (clapping sound) and 201% (human speech) improvement compared to the prior work, and a 31% (clapping sound) and 26% (human speech) improvement when adding the diffraction rays compared to RA-SSL.

When the sound source is on the LOS from 0 to 20 s and from 80 to 125 s, the averages of significant acoustic rays of our approach per frame are 7.36 (primary), 9.52 (reflection), and 2.32 (diffraction) of the clapping sound, respectively, and 6.9

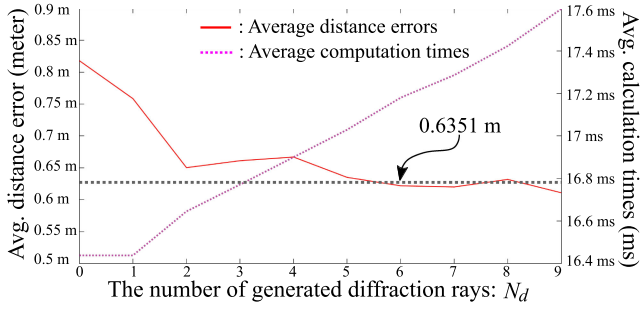


Fig. 14. Average distance errors and computation times for our method on an Intel i7 6700 processor, as a function of the number of diffraction rays generated for simulating the edge diffraction.

(primary), 9.79 (reflection), and 1.15 (diffraction) of the human speech, respectively. When the sound source is occluded by the obstacle, i.e., NLOS source, the averages of significant rays per frame are 0.61 (primary), 9.3 (reflection), and 3.87 (diffraction) of the clapping sound, respectively, and 1.55 (primary), 3.83 (reflection), and 6.39 (diffraction) of the human speech, respectively. Ideally, there should be neither diffraction rays during LOS sources nor primary rays during NLOS sources, respectively. However, in practice, primary rays generated immediately after being occluded by the obstacle and diffraction rays generated just before being occluded by the obstacle were counted in significant primary acoustic rays in the NLOS source cases and affect significant diffraction acoustic rays in the LOS source cases.

The remarkable aspect is that the most primary acoustic rays are blocked by the obstacle, and the effect of the diffraction rays increases when the source is the NLOS state; the averages of significant diffraction rays become larger compared to the LOS source. Also, the average of significant diffraction rays of the human speech is larger than the clapping since diffraction is a low-frequency phenomenon.

B. *Analysis of the Diffraction Acoustic Rays*

To see effects of considering diffraction acoustic rays in addition to the primary and reflection acoustic rays, we measure the accuracy as a function of the number of diffraction acoustic rays N_d. As N_d increases from 0 to 9, we measure the average distance errors and the average of calculation times in the environment containing the obstacle using the clapping sound (Fig. 14); the experimental setting with $N_d = 0$, i.e., no diffraction rays, is the same as the one tested in Fig. 13(a).

The average distance errors are gradually reduced until $N_d = 5$, and the accuracy is almost converged after $N_d = 5$. The averages of calculation times increase linearly, as a function of N_d. Since the accuracy changes after $N_d = 5$ are small enough, we use $N_d = 5$ across all the other experiments. Overall, we observe 29% improvement by using $N_d = 5$ over using no diffraction rays; the average distance errors of $N_d = 0$ and $N_d = 5$ are 0.8112 and 0.6351 m, respectively.

The average running times for acoustic ray tracing and particle filter are 6 and 11 ms; the total average running time is 17 ms

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

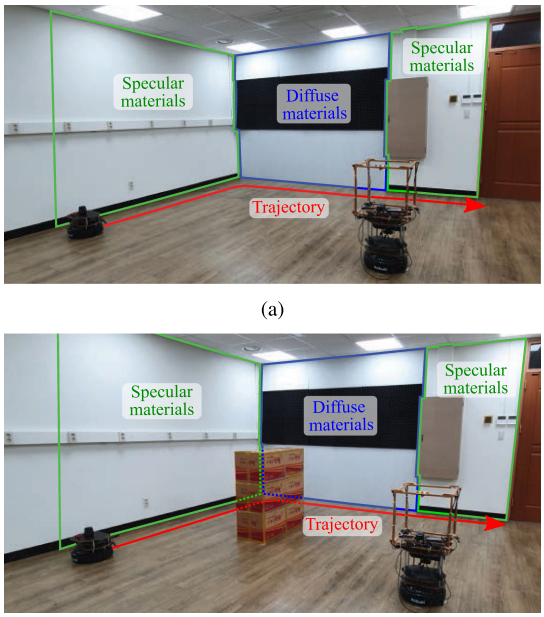


Fig. 15. Environments with one moving source containing high absorption materials, i.e., acoustic soundproofing foam consisting of a sponge, without and with an obstacle. In (a) and (b), we replace part of the specular materials with the diffuse materials from the environments in Fig. 11; the specular materials are indicated by the green rectangles and the diffuse materials are indicated by the blue rectangle. These walls strongly affect our approach, as the source moves from the left end to the right end of the walls consisting of the specular (green rectangles) and diffuse (blue rectangle) materials; many propagation paths coming from the moving source to the microphone array interact with those highlighted materials.

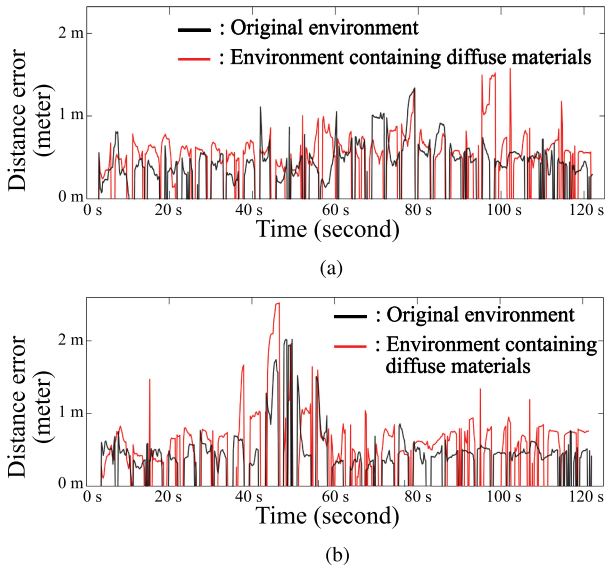


Fig. 16. Distance errors, i.e., red graphs, in the environments in Fig. 15 containing diffuse materials without and with an obstacle.

corresponding to the average calculation time at $N_d = 5$ in Fig. 14.

C. Analysis of Specular and Diffuse Materials

In the previous environments (Section V-A), most materials, such as a solid cement wall, a thick wooden floor, and a gypsum board ceiling, have low absorption coefficients and tend to generate specular reflections. We verify that most materials have a coefficient of 0.1 or lower for all frequency bands using a hand-held measurement device [59], [60].

In real environments, however, there can be diffuse materials, e.g., carpets on the floor and curtains for windows, with high absorption coefficients. These scenes can have fewer specular materials, and the number of reflection propagation paths, especially those caused by diffuse materials, can therefore decrease.

To see how diffuse materials affect our algorithm, we set the environment with a diffuse material, as shown in Fig. 15. We attached diffuse materials, having almost absorption coefficient of 1 for all frequency bands, to a part of the wall (the blue rectangle). Parts of the wall shown by the green and blue rectangles in Fig. 15 are the candidates for causing the dominant reflection propagation paths. We cover those walls by the diffuse material, i.e., acoustic foam.

We tested those situations containing diffuse materials in environments without and with an obstacle [Fig. 15(a) and (b)] using the clapping sound. The corresponding distance errors are shown in Fig. 16, where the average distance errors w/o and w/ an obstacle are 0.6176 and 0.6998 m, respectively. Because there exist direct propagation paths and a sufficient amount of specular materials in the environment without an obstacle in Fig. 15(a), the average distance error, 0.6176 m, is similar to the average distance error, 0.5967 m, of the environment consisting mainly of specular materials in Fig. 12(a); there is only a 3% decrease due to the added absorption materials.

In the scene containing the obstacle in Fig. 15(b), the average distance error, 0.6998 m, deteriorates compared to the average distance error, 0.6351 m, of the environment consisting of the majority of specular materials in Fig. 13(a)—about a 10% decrease due to the added absorption materials. When the sound source becomes an NLOS state and direct propagation paths are blocked, the wall w/ the diffuse materials (blue rectangle) becomes the main material to generate prominent propagation paths. However, those prominent propagation paths cannot be detected by our DoA estimator since most of the energy has been absorbed by the diffuse materials, and this situation is the reason for the deterioration in Fig. 16(b).

Even though the portion of specular materials decreases, our approach shows reasonable localization accuracy compared to the previous environments whose most materials are specular materials: 3% and 10% decrease in both scenes. This graceful degradation is achieved since our method still generates and processes a similar number of acoustic rays. The averages of total significant rays of environments containing absorption materials are 18.46 (w/o obstacle), 19.78 (LOS source w/ obstacle), and 13.8 (NLOS source w/ obstacle), respectively; the detailed results for primary, reflection, and diffraction rays are shown in Table I. These values are similar to the previous environments containing many specular materials, i.e., 16.83 (w/o obstacle), 19.2 (LOS source w/ obstacle), and 13.78 (NLOS source w/ obstacle), respectively. The sound propagation paths that are absorbed by absorption materials and thus are not detected

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

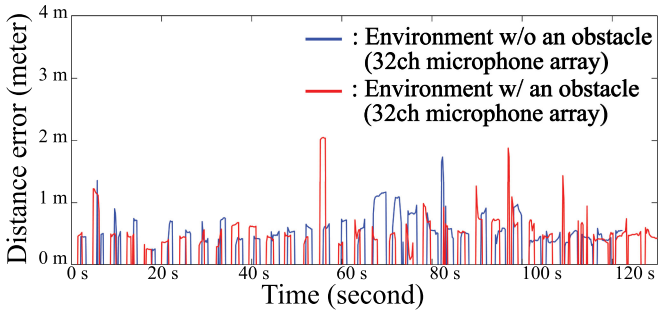


Fig. 17. Distance errors in the environment shown in Fig. 11 without and with an obstacle when using a different microphone array and the DoA estimator: the 32 channel microphone array, i.e., Eigenmike and EB-MVDR beamformer.

by microphones are compensated by other propagation paths caused by other specular materials, i.e., the green rectangles in Fig. 15.

D. Compatibility w/ Different Microphone Arrays

So far, we basically utilized the eight-channel cube shape microphone array with the DAS beamformer. To show that our approach can be combined with different types of microphone arrays and DoA estimators, we tested our method using the 32-channel microphone arrays [45] with EB-MVDR beamformer [46] that is one of the state-of-the-art DoA estimators.

We tested the different microphone array and DoA estimator in the environment without and with an obstacle (Fig. 11), and distance errors are shown in Fig. 17. The distance errors w/ and w/o an obstacle are 0.5946 and 0.6176 m, respectively. These results tell us that our approach can work well based on different types of the microphone array and DoA estimators; both average distance errors are similar or slightly smaller compared to results of the eight-channel microphone array and DAS beamformer, i.e., 0.5967 and 0.6351 m. The reason why those results are slightly better than the 8-channel microphone array is that the 32-channel microphone array has a higher number of channels, i.e., 32 channels, with the state-of-the-art DoA estimators. Even if the 32-channel microphone array has a better performance compared to the 8-channel microphone array, our approach has acceptable accuracies w/ the 8-channel microphone array, which is much cheaper than the tested 32-channel microphone array.

E. Multiple Sound Sources

In general, localizing multiple sources is more difficult than handling a single source, as reverberant sounds tend to accumulate as the number of sources increases. Moreover, our approach can detect up to N different DoAs at a single frame [see (1)]. As a result, the number of allocated rays for each source decreases as there are more sources, and this can deteriorate the localization accuracy. First, we show results in an environment with multiple stationary sources (Fig. 18), remaining at fixed positions, and then present results for multiple moving sources [Fig. 20(a)].

Multiple stationary sources: In a multiple stationary source environment (Fig. 18), we conducted experiments on two scenes, one with two stationary sources and another with three stationary

Fig. 18. Environment with multiple sources. We place up to three sound sources in a room environment. Each red circle indicates a sound location, with each source numbered as source 1, source 2, and source 3.

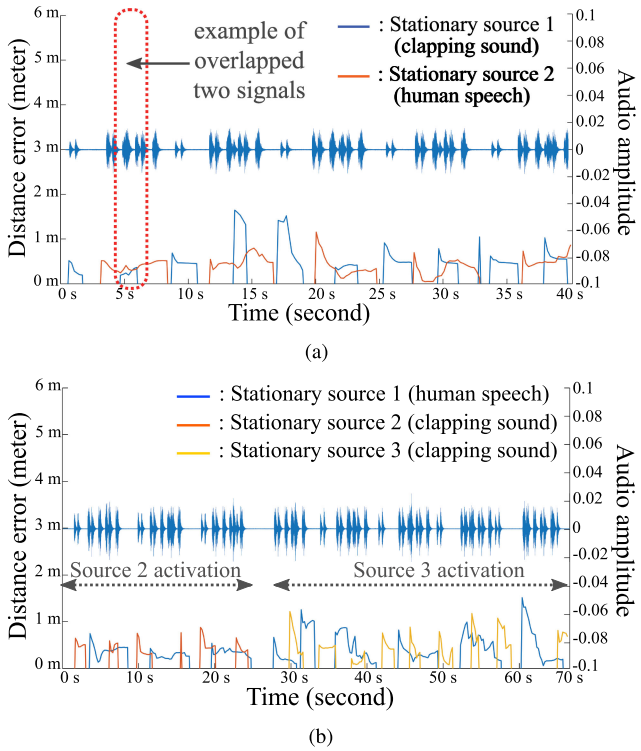
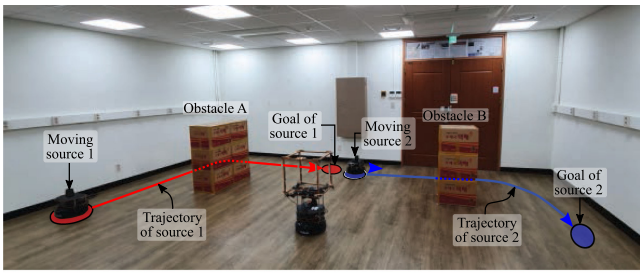


Fig. 19. Distance errors and amplitudes of the measured audio signals of scenes with (a) two and (b) three stationary sources. Sound sources numbering from 1 to 3 correspond to the sources, denoted by the red circles, in Fig. 18. The distance errors of the sources are plotted using lines with different colors, and the amplitudes of the measured audio signals are also presented.

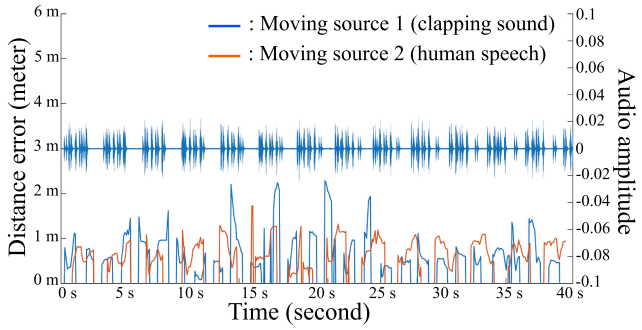
sources. For the former, we place two sources at the positions of sources 1 and 2, highlighted by red circles in Fig. 18, where sources 1 and 2 emit clapping sounds and human speech, respectively. For the scene with three stationary sources, we place three sources at the positions of sources 1, 2, and 3 in Fig. 18, where source 1 emits human speech, and sources 2 and 3 emit clapping sounds. Sources 2 and 3 are active from 0 to 25 s and from 30 to 70 s, respectively, and they are intermittent sound sources.

The localization errors of two and three stationary sources scenes are shown in Fig. 19. In the scene with two stationary sources, the average distance errors of our approach are 0.5947 (source 1) and 0.4306 m (source 2), and the average distance errors of the prior work are 1.6712 (source 1) and 1.6662 m (source 2). In the scene with three stationary sources,

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

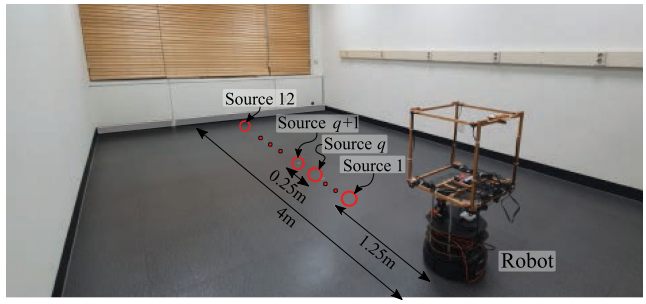


(a)

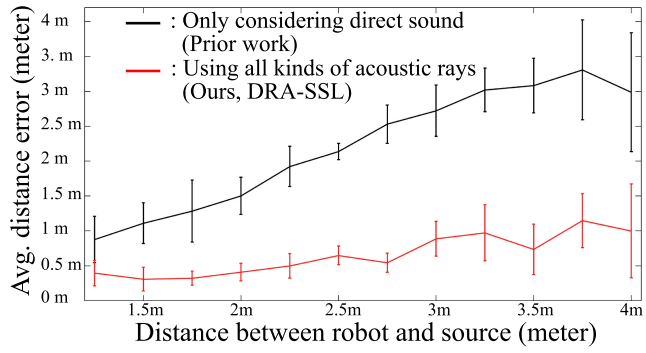


(b)

Fig. 20. Environment of multiple moving sources in (a) and its accuracy in (b). There are two moving sources, i.e., moving source 1 and 2, and they follow trajectories. Both obstacles, i.e., the obstacles A and B, cause the nonline-of-sight states of each moving source.



(a)



(b)

Fig. 21. (a) Another testing environment with a small size of 7 m × 3.5 m in area and with a 3-m height. Red circles denote tested different source positions whose distance from the robot varies from 1.25 to 4 m by a 0.25-m interval. (b) Average distance errors at different source positions. The vertical lines represent the one standard deviation of the average distance errors.

the average distance errors of our approach are 0.4263 (source 1), 0.4856 (source 2), and 0.5185 m (source 3), and the average distance errors of the prior work are 1.3286 (source 1), 1.717 (source 2), and 1.0551 m (source 3); sources 2 and 3 are intermittent sound sources. These results demonstrate that our approach can localize multiple sources reasonably well.

Especially, even if two audio signals coincide, our approach can localize both overlapped signals separately, e.g., a case at 5 s in Fig. 19(a) that is highlighted by a dotted box. Furthermore, even if there are intermittent sound sources, i.e., the sources 2 and 3 in the three-source scene, our approach can distinguish activation and inactivation of intermittent sources. In Fig. 19, when the source 2 is active from 0 to 25 s, and the source 3 is active from 30 to 70 s, our approach only localizes source positions when sources are active and does not react properly to inactivated sources.

Multiple moving sources: For testing our approach to the scene with multiple moving sources, we place two sources in Fig. 20(a), and they follow their trajectories, where sources 1 and 2 emit clapping sound and human speech, respectively. We also put two obstacles between moving sources and the robot to build a more challenging environment; both obstacles block the direct propagation paths of each source.

Distance errors of multiple moving sources are shown in Fig. 20(b). The average distance errors of our approach are 0.7689 (source 1) and 0.7246 m (source 2). Since this scenario containing multiple moving sources is challenging, these errors are higher than single-source scenarios in Fig. 13(a). The accuracy of moving source 1 (clapping sounds) is 17% decreased compared to the one moving source scene containing the obstacle with clapping sounds [Fig. 11(b)]; the environment setups of both experiments are almost the same, i.e., the same obstacle size and the similar trajectory of the source. Since multiple sources generate more reverberant sounds and the total number of generated rays for each source decreases, the accuracy of moving source 1 becomes worse.

The average distance errors of the prior work is presented in Table II: 1.36 (source 1) and 1.812 m (source 2). Compared to the average distance errors of the prior work, we can observe that our approach shows better result; 76% and 150% improvement for sources 1 and 2, respectively.

F. Different Environment Sizes

Thus far, we have tested our approach in environments of identical dimensions, i.e., 7 × 7 m in area and with a 3-m height. To determine how different sizes of the environment affect our method, we conducted an experiment in a room that is 7 × 3.5 m in size and with a height of 3 m as shown in Fig. 21(a). We also measured localization accuracy by increasing the distance between the robot and the source from 1.25 to 4 m.

Fig. 21(a) shows 12 locations of sound sources from the sources 1 to 12; two adjacent sources have the same distance interval of 0.25 m. We place the sound source at one of the source locations, and the source emits the clapping sound for 20 s. To demonstrate the benefits of our approach, we also tested the prior work and our approach. The accuracy of the two cases are shown in Fig. 21(b).

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

We observe that our method helps to improve the accuracy of SSL by using acoustic rays since all average distance errors of our approach are smaller than the prior work. By increasing the distance between the robot and the source, the accuracy generally deteriorates in both cases.

The prior work utilizes the time differences of arrival sound to each microphone to estimate the distance between the microphone array and the source. As the source becomes far away from the microphone array, the change in the time differences of arrival sound decreases. Thus, it becomes difficult to accurately estimate the distance between the microphone array and the source from time differences; the localization accuracy decreases in longer distances, as reported in Fig. 21(b).

In our case, this is attributed by the accumulated propagation errors of acoustic ray paths caused by various noises, e.g., sensor noises of the laser scanner, audio noises of microphones, and odometry noises of the mobile robot. These noises adversely affect our localization algorithm and cause propagation errors in our approach. Nonetheless, our approach shows the better and stable accuracy compared to the prior work only considering direct sound.

G. Navigating to the NLOS Source

We expect that our approach can be applied to various tasks in robotics. Especially, our approach is useful in cases of containing an NLOS source; the vision-based localization approaches do not deal with these cases due to the occlusion by obstacles. Assuming that a user orders the robot to bring something, e.g., a cup of water, the robot has to detect and localize the user and then navigate to the location of the user. If there is an obstacle between the user and the robot, the vision sensor does not see the user, but the sound can be heard through indirect sound propagation; sound becomes very crucial information in the NLOS source cases.

We applied our approach to the navigation task. When the source emits the clapping sound at a specific goal position, which is unknown for the robot, the robot localizes the source and navigates to the estimated goal position by our localization method. To simulate the NLOS case, the sound source is occluded by an obstacle [Fig. 22(a)], and we tested our method and the prior work to localize the source.

If localization methods, i.e., our approach and prior work, produce the estimated source position for 2 s, the robot sets an estimated goal position as the mean of estimated source positions of localization methods; the duration of the clapping sound is about 2 s. During the navigation tasks, the sound source plays the sound clip three times periodically in order to show the localization result over the different robot positions. The robot stops navigation process once the distance to the estimated goal position is less than 1 m.

We utilize Jackal as a mobile robot platform. Jackal provides the open source of the navigation in the ROS system, and we use this open source in this experiment where the linear and angular velocities are 0.1 m/s and 0.314 rad/s, respectively, the linear and angular accelerations are 2.0 m/s^2 and 4.0 rad/s^2, respectively, and other parameters are set to default values. The microphone array is the same eight-channel cube-shaped type as in previous experiments.

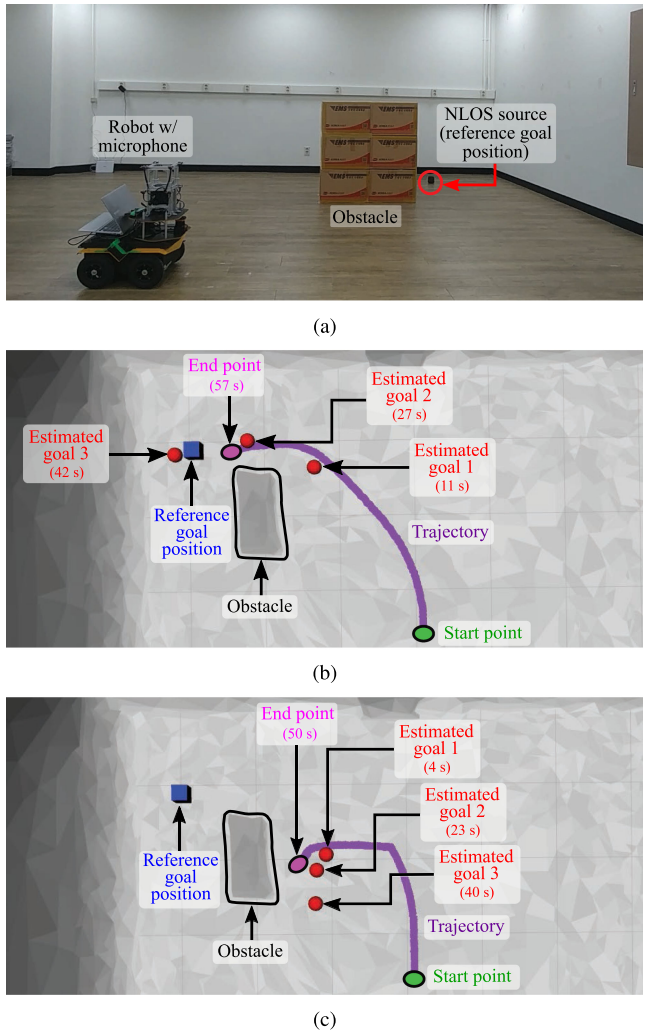


Fig. 22. (a) Test environment for the navigation task to the NLOS source. (b) and (c) Results of our navigation tasks and those of the prior work, respectively. The blue cubes denote the reference goal position generating a clapping sound, and red spheres represent the estimated goal position at each time. The purple lines are the computed trajectory of the robot, given the start point (green circle) and the end point (purple circle).

The results of our navigation tasks and those of the prior work are shown in Fig. 22(b) and (c). We observe that the robot can reach the reference goal position at 57 s, given our localization approach, as shown in Fig. 22(b). On the other hand, the robot with the prior work does not consider indirect sound paths, failing in reaching the reference position; the navigation task of the prior work is stopped at 50 s. Especially, the estimated goal positions of our approach are gradually getting close to the reference goal position; the distance errors of estimated goals 1, 2, and 3 are 1.5292 m at 11 s, 0.659 m at 27 s, and 0.3361 m at 42 s, respectively. The estimated goal positions of the prior work get worse while the robot moves closely to the obstacle; the distance errors of estimated goals 1, 2, and 3 are 1.9397 m at 4 s, 2.09 m at 23 s, and 2.1669 m at 40 s, respectively.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

The prior work that considers only direct sound was not able to handle the NLOS case caused by the obstacle, but our approach can reach the destination, i.e., the NLOS source, mainly thanks to the consideration of diffraction.

VI. Conclusion

In this article, we presented a novel reflection- and diffraction-aware SSL algorithm by utilizing acoustic ray tracing and MCL for multiple sound sources. Our approach can also localize NLOS sound sources and model diffraction using the UTD. We evaluated our method in various scenarios with static and moving single or multiple sources using different sound signals. We also analyzed the properties of our method across a diverse set of configurations with different materials, room sizes, beamforming algorithms, etc. We applied our approach to the navigation task and confirmed the usefulness of our approach.

While we have demonstrated the benefits of our approach, it has some limitations that need to be addressed by future work. The UTD model is an approximate model and is mainly designed for infinite wedges. Its accuracy can deteriorate on obstacles that have smooth surfaces. More accurate wave-based diffraction models can be used to deal with this problem, but achieving real-time performance remains as a main technical challenge.

Our approach works based on interactions, i.e., reflection and diffraction, with obstacles and is not suitable for outdoor environments where we do not have obstacles causing interaction. As mentioned in Section V-E, our method may not work properly when reverberation becomes prominent. This issue can be mitigated by utilizing semantic information of sound signal that each sound source carries. Overall, we believe that the proposed work takes a meaningful step for SSL, and considering the aforementioned issues can open up new research directions.

References

[1] J.-M. Valin, F. Michaud, and J. Rouat, "Robust 3D localization and tracking of sound sources using beamforming and particle filtering," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2006, pp. IV–IV.

[2] C. C. Douglas and R. A. Lodder, "Human identification and localization by robots in collaborative environments," *Procedia Comput. Sci.*, vol. 108, pp. 1602–1611, 2017.

[3] M. Imran, A. Hussain, N. M. Qazi, and M. Sadiq, "A methodology for sound source localization and tracking: Development of 3D microphone array for near-field and far-field applications," in *Proc. Int. Bhurban Conf. Appl. Sci. Technol.*, 2016, pp. 586–591.

[4] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, no. 4, pp. 320–327, Aug. 1976.

[5] F. Grondin and F. Michaud, "Time difference of arrival estimation based on binary frequency mask for sound source localization on mobile robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 6149–6154.

[6] W. He, P. Motlicek, and J.-M. Odobez, "Deep neural networks for multiple speaker detection and localization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 74–79.

[7] F. Grondin and J. Glass, "Fast and robust 3-D sound source localization with DSVD-PHAT," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 5352–5357.

[8] J.-M. Valin, F. Michaud, and J. Rouat, "Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering," *Robot. Auton. Syst.*, vol. 55, no. 3, pp. 216–228, 2007.

[9] C. Zhang, D. Florêncio, D. E. Ba, and Z. Zhang, "Maximum likelihood sound source localization and beamforming for directional microphone arrays in distributed meetings," *IEEE Trans. Multimedia*, vol. 10, no. 3, pp. 538–548, Apr. 2008.

[10] S. Argentieri and P. Danes, "Broadband variations of the music high-resolution method for sound source localization in robotics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2007, pp. 2009–2014.

[11] K. Nakamura, K. Nakadai, F. Asano, Y. Hasegawa, and H. Tsujino, "Intelligent sound source localization for dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 664–669.

[12] K. Nakamura, K. Nakadai, and G. Ince, "Real-time super-resolution sound source localization for robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 694–699.

[13] Y. Sasaki, R. Tanabe, and H. Takemura, "Probabilistic 3D sound source mapping using moving microphone array," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 1293–1298.

[14] D. Su, T. Vidal-Calleja, and J. V. Miro, "Towards real-time 3D sound sources mapping with linear microphone arrays," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 1662–1668.

[15] P. Misra, A. A. Kumar, P. Mohapatra, and P. Balamuralidhar, "DroneEARS: Robust acoustic sound localization with aerial drones," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 80–85.

[16] Q. V. Nguyen, F. Colas, E. Vincent, and F. Charpillet, "Localizing an intermittent and moving sound source using a mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 1986–1991.

[17] A. Portello, G. Bustamante, P. Danés, J. Piat, and J. Manhes, "Active localization of an intermittent sound source from a moving binaural sensor," in *Proc. Eur. Acoust. Assoc. Forum Acusticum*, 2014, p. 12.

[18] J. Even, J. Furrer, Y. Morales, C. T. Ishi, and N. Hagita, "Probabilistic 3-D mapping of sound-emitting structures based on acoustic ray casting," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 333–345, Apr. 2017.

[19] N. Kallakuri, J. Even, Y. Morales, C. Ishi, and N. Hagita, "Using sound reflections to detect moving entities out of the field of view," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 5201–5206.

[20] J. Even, Y. Morales, N. Kallakuri, C. Ishi, and N. Hagita, "*Audio ray tracing for position estimation of entities in blind regions*," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 1920–1925.

[21] H. Kuttruff, *Acoustics: An Introduction*. Boca Raton, FL, USA: CRC Press, 2007.

[22] M. Vorländer, "Simulation of the transient and steady-state sound propagation in rooms using a new combined ray-tracing/image-source algorithm," *J. Acoust. Soc. Am.*, vol. 86, no. 1, pp. 172–178, 1989.

[23] C. Schissler, R. Mehra, and D. Manocha, "High-order diffraction and diffuse reflections for interactive sound propagation in large environments," *ACM Trans. Graph.*, vol. 33, no. 4, 2014, Art. no. 39.

[24] H. Yeh, R. Mehra, Z. Ren, L. Antani, D. Manocha, and M. Lin, "Wave-ray coupling for interactive sound propagation in large complex scenes," *ACM Trans. Graph.*, vol. 32, no. 6, 2013, Art. no. 165.

[25] J. B. Keller, "Geometrical theory of diffraction," *JOSA*, vol. 52, no. 2, pp. 116–130, 1962.

[26] I. An, M. Son, D. Manocha, and S. Yoon, "Reflection-aware sound source localization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 66–73.

[27] I. An, D. Lee, J. Choi, D. Manocha, and S. Yoon, "Diffraction-aware sound localization for a non-line-of-sight source," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 4061–4067.

[28] D. Di Carlo, A. Deleforge, and N. Bertin, "Mirage: 2D source localization using microphone pair augmentation with echoes," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 775–779.

[29] J.-M. Valin, F. Michaud, J. Rouat, and D. Létourneau, "Robust sound source localization using a microphone array on a mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, pp. 1228–1233.

[30] B. Teng and R. E. Taylor, "New higher-order boundary element methods for wave diffraction/radiation," *Appl. Ocean Res.*, vol. 17, no. 2, pp. 71–77, 1995.

[31] S. R. Martin, U. P. Svensson, J. Slechta, and J. O. Smith, "A hybrid method combining the edge source integral equation and the boundary element method for scattering problems," in *Proc. Meet. Acoust.*, vol. 26, no. 1, ASA, 2016, Art. no. 015001.

[32] A. Rungta, C. Schissler, N. Rewkowski, R. Mehra, and D. Manocha, "Diffraction kernels for interactive sound propagation in dynamic environments," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 4, pp. 1613–1622, Apr. 2018.

[33] N. Tsingos and J.-D. Gascuel, "Fast rendering of sound occlusion and diffraction effects for virtual acoustic environments," in *Proc. 104th Conv. Audio Eng. Soc.*, 1998, pp. 1–14.

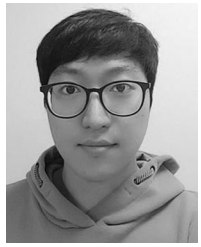
[34] U. P. Svensson, R. I. Fred, and J. Vanderkooy, "An analytic secondary source model of edge diffraction impulse responses," *J. Acoust. Soc. Am.*, vol. 106, no. 5, pp. 2331–2344, 1999.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

[35] A. Asheim and U. Peter Svensson, "An integral equation formulation for the diffraction from convex plates and polyhedra," *J. Acoust. Soc. Am.*, vol. 133, no. 6, pp. 3681–3691, 2013.
[36] R. G. Kouyoumjian and P. H. Pathak, "A uniform geometrical theory of diffraction for an edge in a perfectly conducting surface," *Proc. IEEE Inst. Electr. Electron. Eng.*, vol. 62, no. 11, pp. 1448–1461, Nov. 1974.
[37] L. Antani, A. Chandak, M. Taylor, and D. Manocha, "Efficient finite-edge diffraction using conservative from-region visibility," *Appl. Acoust.*, vol. 73, no. 3, pp. 218–233, 2012.
[38] N. Tsingos, T. Funkhouser, A. Ngan, and I. Carlbom, "Modeling acoustics in virtual environments using the uniform theory of diffraction," in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Techn.*, ACM, 2001, pp. 545–552.
[39] M. Taylor, A. Chandak, Z. Ren, C. Lauterbach, and D. Manocha, "Fast edge-diffraction for sound propagation in complex virtual environments," in *Proc. EAA Auralization Symp.*, 2009, pp. 15–17.
[40] M. Taylor, A. Chandak, Q. Mo, C. Lauterbach, C. Schissler, and D. Manocha, "Guided multiview ray tracing for fast auralization," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 11, pp. 1797–1810, Nov. 2012.
[41] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1271–1278.
[42] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Trans. Graph.*, vol. 32, no. 3, 2013, Art. no. 29.
[43] M. Tarini, N. Pietroni, P. Cignoni, D. Panozzo, and E. Puppo, "Practical quad mesh simplification," in *Proc. Comput. Graph. Forum*, vol. 29, no. 2, Wiley Online Library, 2010, pp. 407–418.
[44] R. Schnabel, R. Wahl, and R. Klein, "Efficient RANSAC for point-cloud shape detection," in *Proc. Comput. Graph. Forum*, vol. 26, no. 2, Wiley Online Library, 2007, pp. 214–226.
[45] M. Binelli, A. Venturi, A. Amendola, and A. Farina, "Experimental analysis of spatial properties of the sound field inside a car employing a spherical microphone array," in *Proc. AES Conv.*, 2011.
[46] H. Sun, E. Mabande, K. Kowalczyk, and W. Kellermann, "Joint DOA and TDOA estimation for 3D localization of reflective surfaces using eigenbeam MVDR and spherical microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2011, pp. 113–116.
[47] Y.-H. Kim and J.-W. Choi, *Sound Visualization and Manipulation*. Hoboken, NJ, USA: Wiley, 2013.
[48] F. X. Giraldo, "Lagrange-Galerkin methods on spherical geodesic grids," *J. Comput. Phys.*, vol. 136, no. 1, pp. 197–213, 1997.
[49] F. Jacobsen, T. Poulsen, J. H. Rindel, A. C. Gade, and M. Ohlrich, "Fundamentals of acoustics and noise control," Dep. Electr. Eng., Tech. Univ. Denmark, Denmark, 2011.
[50] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA, USA: MIT Press, 2005.
[51] T. W. Anderson, Ed., *An Introduction to Multivariate Statistical Analysis*. Hoboken, NJ, USA: Wiley, 1984.
[52] S. Briere, J.-M. Valin, F. Michaud, and D. Létourneau, "Embedded auditory system for small mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 3463–3468.
[53] F. Grondin, D. Létourneau, F. Ferland, V. Rousseau, and F. Michaud, "The ManyEars open framework," *Auton. Robots*, vol. 34, no. 3, pp. 217–232, 2013.
[54] M. Wise, M. Ferguson, D. King, E. Diehr, and D. Dymesich, "Fetch and freight: Standard platforms for service robot applications," in *Proc. Workshop Auton. Mob. Serv. Robot.*, 2016.
[55] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, and G. Ranzuglia, "MeshLab: An open-source mesh processing tool," in *Proc. Eurographics Ital. Chap. Conf. Eurographics Assoc.*, 2008, pp. 129–136.
[56] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Close-range scene segmentation and reconstruction of 3D point cloud maps for mobile manipulation in domestic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 1–6.
[57] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Rob. Auton. Syst.*, vol. 56, no. 11, pp. 927–941, 2008.
[58] H. Kuttruff, *Room Acoustics*. Boca Raton, FL, USA: CRC Press, 2016.
[59] H.-E. de Bree, M. Nosko, and E. Tijs, "A handheld device to measure the acoustic absorption in situ," in *Proc. SNVH*, GRAZ, 2008.
[60] R. Lanoye, H.-E. de Bree, W. Lauriks, and G. Vermeir, "A practical device to determine the reflection coefficient of acoustic materials in-situ based on a microflown and microphone sensor," in *Proc. ISMA*, pp. 2665–2675.

Inkyu An (Student Member, IEEE) received the B.S. degree in electronic engineering from Dongguk University, Seoul, South Korea, in 2016 and the M.S. degree in robotics program from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2018. He is currently working toward the Ph.D. degree with the School of Computing, KAIST.

His research interests include ray tracing-based sound source localization and acoustic material estimation. He is currently studying speech-oriented source localization.



Youngsun Kwon (Student Member, IEEE) received the B.S. degree in electronic and electrical engineering from Sungkyunkwan University, Seoul, South Korea, in 2014 and the M.S. degree in robotics program from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2016. He is currently working toward the Ph.D. degree with the School of Computing, KAIST.

His research interests include occupancy mapping, uncertainty-aware collision detection, and sensor-based motion planning. He is currently working on real-time occupancy mapping and collision avoidance.

Sung-eui Yoon (Senior Member, IEEE) received the B.S. and M.S. degrees in computer science from Seoul National University, Seoul, South Korea, in 1999 and 2001, respectively, and the Ph.D. degree in computer science from the University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, in 2005.

He is currently a Professor with Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea. He was a Postdoctoral Scholar with Lawrence Livermore National Laboratory, Livermore, CA, USA. He has authored or coauthored more than 70 technical papers in top journals and conference related to graphics, vision, and robotics. He also gave numerous tutorials on ray tracing, collision detection, image search, and sound source localization in premier conferences like *ACM SIGGRAPH*, *IEEE Visualization*, *CVPR*, and *ICRA*. In 2008, he authored or coauthored a monograph on real-time massive model rendering with other three coauthors. He also authored or coauthored an online book on rendering in 2018. His research interests include rendering, image search, and motion planning spanning graphics, vision, and robotics.

Dr. Yoon served as a conference Co-Chair and paper Co-Chair for ACM I3D in 2012 and 2013, respectively. Some of his papers received a test-of-time award, a distinguished paper award, and a few invitations to IEEE TRANSACTIONS ON VISUALIZATION AND GRAPHICS. He is currently a Senior Member of ACM.