

PathFinder: Designing a Map-less Navigation System for Blind People in Unfamiliar Buildings

Masaki Kuribayashi
Waseda University
Tokyo, Japan
rugbykuribayashi@toki.waseda.jp

Jayakorn Vongkulbhaisal
IBM Research - Tokyo
Tokyo, Japan
jayakornv@gmail.com

Hironobu Takagi
IBM Research - Tokyo
Tokyo, Japan
takagih@jp.ibm.com

Tatsuya Ishihara
IBM Research - Tokyo
Tokyo, Japan
tishihara@jp.ibm.com

Karnik Ram
Carnegie Mellon University
Pittsburgh, United States
karnikr@andrew.cmu.edu

Shigeo Morishima
Waseda Research Institute for Science
and Engineering
Tokyo, Japan
shigeo@waseda.jp

Daisuke Sato
Carnegie Mellon University
Pittsburgh, United States
daisukes@cmu.edu

Seita Kayukawa
Waseda University
Tokyo, Japan
k940805k@ruri.waseda.jp

Chieko Asakawa
Carnegie Mellon University
Pittsburgh, United States
IBM Research
New York, United States
chiekao@us.ibm.com

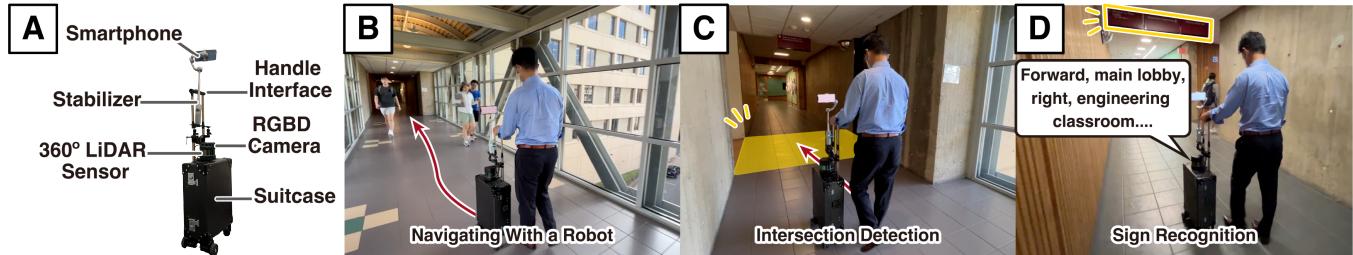


Figure 1: We present PathFinder, a map-less navigation system that can navigate blind people in unfamiliar buildings by detecting intersections and recognizing signs.

ABSTRACT

Indoor navigation systems with prebuilt maps have shown great potential in navigating blind people even in unfamiliar buildings. However, blind people cannot always benefit from them in every building, as prebuilt maps are expensive to build. This paper explores a map-less navigation system for blind people to reach destinations in unfamiliar buildings, which is implemented on a robot. We first conducted a participatory design with five blind people, which revealed that intersections and signs are the most relevant information in unfamiliar buildings. Then, we prototyped PathFinder, a navigation system that allows blind people to determine their way by detecting and conveying information about intersections and

signs. Through a participatory study, we improved the interface of PathFinder, such as the feedback for conveying the detection results. Finally, a study with seven blind participants validated that PathFinder could assist users in navigating unfamiliar buildings with increased confidence compared to their regular aid.

CCS CONCEPTS

- Human-centered computing → Accessibility systems and tools;
- Social and professional topics → People with disabilities.

KEYWORDS

visual impairment, orientation and mobility, intersection detection, sign recognition

ACM Reference Format:

Masaki Kuribayashi, Tatsuya Ishihara, Daisuke Sato, Jayakorn Vongkulbhaisal, Karnik Ram, Seita Kayukawa, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. 2023. PathFinder: Designing a Map-less Navigation System for Blind People in Unfamiliar Buildings. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*, April 23–28, 2023, Hamburg, Germany. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3544548.3580687>

1 INTRODUCTION

Blind people face a significant challenge when navigating independently to a destination in an unfamiliar building. To be able to navigate in such buildings with their regular navigation aids such as white canes or guide dogs, long-term familiarization by learning non-visual cues is necessary. Thus, in unfamiliar buildings, they usually need sighted people to accompany them to the destination [14], and when they are alone, they have to ask passersby for directions and further help on-site. Despite its difficulty, blind people still travel in unfamiliar buildings [14, 25, 45], and hope to do so without being accompanied by a sighted person [14]. This suggests that there is a need for a navigation tool that will help blind people navigate unfamiliar buildings without having sighted people accompany them to the destination.

To navigate blind people in unfamiliar buildings, previous researches have utilized static route maps (*i.e.*, a map with the route topology and points-of-interest (POIs) annotated) and localization methods on various devices (*e.g.*, smartphones [5, 31, 56], wearable devices [40, 53] and robots [21, 33, 41]). By using static route maps, systems can provide users with turn-by-turn instructions and environmental information (*e.g.*, POIs and intersections), which allows users to reach their destinations. In particular, autonomous navigation robots [21, 29, 33] can guide blind people fully automatically to a destination, while ensuring users' safety by avoiding surrounding obstacles using additional Light Detection And Ranging (LiDAR) map (*i.e.*, a two-dimensional occupancy grid map made from LiDAR sensor data). Our previous studies [21, 29] have shown that the use of autonomous navigation robots is effective, as blind people only need to follow it, and therefore it can increase users' confidence and decrease their cognitive load when navigating buildings. However, they cannot always benefit from such navigation systems, as these prebuilt maps (static route maps and/or LiDAR maps) require tedious and time-consuming labor from experts to build, verify, and deploy, making them expensive [56]. Therefore, we decided to prototype a map-less navigation system that uses the surrounding information to help guide the blind user. To build such a system, we aim to address the following research questions. (1) What kind of information is useful for blind people to reach a destination in unfamiliar buildings? (2) How can blind people interact with the assistance system to reach a destination in unfamiliar buildings?

To answer these questions, we used a scenario-based participatory design approach [55] with five blind participants to understand what kind of environmental information would facilitate their navigation in unfamiliar buildings when using a navigation robot as their aid. We used a navigation robot [29], which is grounded with wheels, for the implementation of our system as it can utilize a variety of sensors to collect environmental information, which in turn can be efficiently processed and conveyed to the users via the attached high-performance computers and interfaces. Also, its guidance using motorized wheels allows users to focus on the navigation by guiding the users safely without veering and collisions while walking to a specific direction [21]. During the study, an experimenter gave the participants a description of a route, which was gathered from an interview session with ten sighted passersby. Then, assuming the experimenter as a navigation robot, the experimenter accompanied the participants along the explained route

while describing several indoor features along the way. Throughout the study, the blind participants mainly expressed that intersections and signs, such as directional signs (*i.e.*, signs which contain arrows to indicate where places are) and textual signs (*i.e.*, signs which only contain texts, such as room numbers and names of places), are the most useful information when navigating unfamiliar buildings.

Based on these findings, we designed and prototyped a map-less navigation system, called *PathFinder* (Fig. 1), on top of a suitcase-shaped robot. *PathFinder* is designed for navigating blind people in the scenario where the user has acquired the route description from sighted passersby. The user can command the system via its handle interface to find the next intersection (Fig. 1-C) or the end of a hallway, and describe visible directional and textual signs (Fig. 1-D) to identify the path to the destination. The system adopts audio feedback to the user to convey detection results, such as the shapes of intersections and descriptions of signs.

A session for design iteration was conducted with the same five blind participants to gather feedback and comments about the interface and functionalities of the system. Through the study, we obtained suggestions regarding the system's audio feedback and handle interface. The participants also requested an additional "Take-me-back" functionality, where the system takes the user back to the location where they started their navigation.

Finally, we conducted a user study with seven blind participants on the system after incorporating the suggestions from the participatory study. During the study, we prepared two routes with several intersections and signs and asked the participants to navigate them using *PathFinder* and a topline system, which is a navigation system with prebuilt maps. Through our interview with the participants, we found that the participants felt they were able to navigate to the destination with increased confidence and less cognitive load with *PathFinder* compared to their daily navigation aid. In addition, while all participants mentioned that *PathFinder* required more effort for them to control than the system with prebuilt maps, they agreed that *PathFinder* is a useful navigation system as it can operate in more places, and they were able to navigate to their destinations without having to be accompanied by a sighted person.

Below, we summarize the contribution of this paper.

- (1) We propose a map-less navigation robot system for navigating blind people in unfamiliar buildings. To design the system, we performed a participatory study with blind people to gather their insights and suggestions. Based on the study, we designed the system to recognize signs and detect intersections, then convey information to the blind user via audio feedback.
- (2) We conducted a quantitative and qualitative user study of the proposed map-less navigation robot system with seven blind participants. Based on the result, we discuss the functionalities and the limitations of the system, and also provide insights for designs of future map-less navigation systems.

2 RELATED WORK

2.1 Navigation in Unfamiliar Buildings by Blind People

The navigation of blind people in unfamiliar buildings has been studied broadly. A study by Jeamwathanachai *et al.* [25] concluded

that navigating unfamiliar buildings is challenging for blind people because determining their current location and their route to a destination while also having to maintain their orientation is challenging without visual information and sufficient knowledge of the environment. Engel *et al.* [14] conducted a large-scale study about the travel behavior of blind people. In their study, 59.4% of 63 blind participants answered that they travel to an unfamiliar building several times a week, despite its difficulty. According to their study, there are two main ways for blind people to find their route to a destination. One way is to search for the textual description of the route on the internet. Although this might seem like a feasible way, it has been reported that textual descriptions of routes are often not available and this preparation is time-consuming, hence some blind people do not prepare a route at all [14]. Another way is to ask sighted people for the route description on site. As this requires no preparation with the added chance that sighted people would accompany them to the destination, this is the most common option for blind people [14, 25, 51]. Finally, Engel *et al.* also showed that blind people want to navigate without having sighted people accompany them to the destination, which is the main motivation for our study.

2.2 Indoor Navigation Systems for Blind People

Past studies have proposed various navigation systems that help blind people navigate inside buildings [30, 34]. They usually utilize a static route map and different localization methods (*e.g.*, blue-tooth low energy (BLE) beacons [31, 56], ultrawide-bandwidth beacons [41], and visual features [35, 40, 67]) to navigate a blind user to their destination. There are also several works that aimed to navigate blind people to avoid obstacles in their proximity with only real-time sensing results [42, 48, 61]. Such systems have been proposed using smartphones [31, 35, 36, 50, 56, 67], wearable devices [10, 12, 26, 39, 40, 53], suitcase-shaped device [27, 28], and robots [21, 37, 41]. For example, Kayukawa *et al.* proposed obstacle avoidance systems using a suitcase-shaped device that emits an alerting sound to clear nearby pedestrians from blind users' way [27], and a suitcase with a directional lever that points a safe path to follow using a LiDAR map. Both systems require users to push the suitcase, which may cause navigation errors induced by users. On the other hand, robots have shown high potential in navigating blind people, as blind people only have to follow the movement of robots. Some researchers explored the interaction between such robots and blind people [1, 37, 64, 68], and other researchers have explored navigation algorithms for their assistance [21, 41, 62]. In particular, Carry-on-robot (CaBot) is a suitcase-shaped navigation robot that we presented earlier [21]. The robot plans its path by considering the user's position relative to the robot and guides the user using a handle interface. However, the drawback of all these systems is that blind people cannot use them everywhere, as these systems require either or both static route maps to provide turn-by-turn instructions and additional LiDAR maps for planning obstacle-avoiding paths, both of which require tedious labor. Therefore, we design a system that navigates users to their destination in unfamiliar buildings without requiring any prebuilt maps, *i.e.*, both LiDAR map and static route map, so that blind people can benefit from the capabilities of navigation systems in a wide variety of

places. To this end, we implement the map-less system on a robot as it can utilize various sensors and high-performance computers to process environmental information while allowing the users to focus on the task by relying on its obstacle-avoiding capability. Thus, in the next section, we introduce map-less navigation technologies used in robotics.

2.3 Map-less Navigation Technology for Robots

In robotics, many works have studied the problem of navigation in environments without prebuilt maps. To navigate a robot to a destination, several approaches rely on vision-based techniques; for example, matching real-time RGB images with sequences of pre-captured images for path following [9, 46], recognizing the surrounding objects to help with localization [11, 46], and using an image of the target location and applying reinforcement learning to find the path to the goal [43, 71]. While these methods do not use prebuilt maps, they use alternative information to navigate the robot to the destination. On the other hand, methods of navigation that create maps during exploration have also been proposed. Examples of these approaches include the construction of topological maps by detecting intersections [66] and occupancy maps using RGB images and deep reinforcement learning [52] while the robot is navigating the space. These methods do not require any information prior to the navigation because they aim to explore the environment. Inspired by these works, we utilize the navigation technique proposed for robot exploration to help navigate blind people in unfamiliar buildings, particularly the intersection detection technique.

There have been several intersection detection algorithms proposed for robots to determine their path in an environment where prebuilt maps are unavailable. Garcia *et al.* proposed a method to detect indoor intersections only with RGB images using a rule-based algorithm [19] and convolution neural networks [20] for quadcopters. Intersection detection in complex environments such as outdoor and underground mines has also been explored in past studies using a LiDAR sensor [38, 44, 66, 70] and an RGB camera [24]. In particular, Yang *et al.* [66] proposed a method to detect intersections in arbitrarily shaped environments using a 360° LiDAR sensor and a real-time simultaneous localization and mapping (SLAM) algorithm. However, since their motivation is to explore novel environments quickly and create extensive LiDAR maps in a short time, the robot may travel in a manner where it does not take an accompanying blind person into account. For example, a robot may move very close to a wall or change its orientation frequently. To build a map-less navigation system for blind people, we apply the method of Yang *et al.* [66] to detect intersections, and include additional functionalities to take into account the accompanying blind user.

2.4 Shared Control for Robots

Without prebuilt maps and reference information of the destination, navigation systems cannot determine the path to reach the destination. To handle this issue, we employ shared control, *i.e.*, a method to control a robot using both human decision and the functionality of a system. According to Wang and Zhang [63], shared control is defined as a “*case in which the robot motion is determined by*

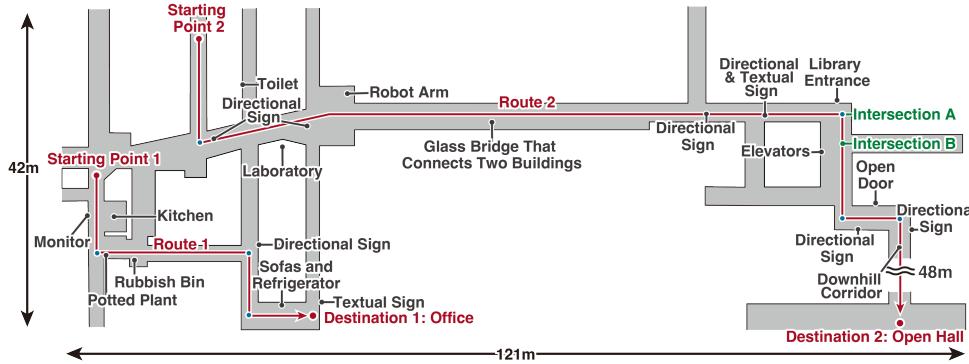


Figure 2: A floor map of the building where the study was conducted, showing the two routes, various POIs along the routes, and examples of the route descriptions.

both the human operator and robot decisions in a mostly balanced fashion.” Shared control can be separated into near-operation, in which the operator perceives the scene with their direct sense, and teleoperation, in which the operator perceives the scene indirectly, such as through a screen. For example, near-operation has been used for assisting a driver to keep their vehicles in lane [47], controlling a wheelchair [49, 57, 58], and for assisting blind people to navigate in familiar buildings [23, 32, 37], while teleoperation has been used for navigating where a human cannot go [2, 13, 22], or for reconnaissance [60]. Similar to previous work, PathFinder adopts near-operation shared control, so that users can complement missing map information that comes from map-less restriction, while the system can help users to navigate safely through buildings. Furthermore, our proposed system allows users to effectively determine the way to the destination in unfamiliar buildings by conveying environmental information, which is described in the next section.

2.5 Conveying Environmental Information to Blind People

For the purpose of supporting blind people during navigation, researchers have worked on systems that can convey environmental information such as traffic lights [6, 59], doors [17, 54], intersections [36], and signs [1, 54, 65]. We particularly focus on conveying information about intersections and signs, as blind participants in our study have indicated that they are useful in unfamiliar buildings, which will be described in Section 3.2.

2.5.1 Intersection Information. Detecting and/or utilizing intersection information has also been done in the field of accessibility to provide turn-by-turn instructions to blind users [16, 36, 37, 56]. Lacey and MacNamara explored a smart walker with passive traction to navigate the elderly blind by informing intersections as landmarks in a controlled and familiar building, such as a residential home [37]. Also, Kuribayashi *et al.* proposed a system for blind people that conveys the location and shape of intersections by detecting them using a LiDAR map constructed with a smartphone [36]. In their study, they revealed that conveying the shape of the intersection is effective for the navigation of blind people, as

Route 1 description example

“Go straight, turn on the first left. Then go straight down the hallway until you meet the right corner. Basically, that’s an end. Turn right there. Go straight. When you see a corner, turn left. If you walk straight for around 12 steps, you will see the destination with a specific room number in front of you.”

Route 2 description example

“Go straight, turn on the first left. Then go straight down the hallway until you meet the end which leads to the right. Turn right there. Go to the next end which leads to left. Turn left there. Walk ten steps, turn right. Go all the way down the hallway. Once you reach an open space, you have reached the destination.”

it helps them to localize themselves and to learn about the environment [36]. Therefore, based on their findings, we also convey the shape of the intersection every time blind users reach it.

2.5.2 Sign Information. Sign information has been considered a useful object to detect, as they generally contain information about the surroundings. In the area of computer vision, researchers have aimed to recognize texts on signs in the real-world [8] or detect sign boards [3]. On the other hand, signs that appear in indoor environments contain arrows that correspond with words that represent locations. Thus, to assist blind people to determine their way, a different system has been proposed in the field of accessibility. Saha *et al.* [54] developed a system that can read signs on a smartphone and revealed that reading textual signs (*e.g.*, names of surrounding shops) can help blind people reach their destination. Yamanaka *et al.* [65] proposed a method to recognize all directional signs using a 360° RGB camera and verified that their system helped blind participants make a decision at intersections in tactile pavings. However, each of them reads either directional signs or textual signs. In contrast, we propose a sign recognition algorithm that can distinguish and read both directional and textual signs. We achieve this by utilizing an object detection model to detect arrows and words, and by proposing a new algorithm that will analyze their correspondence.

3 PARTICIPATORY DESIGN WITH BLIND PEOPLE

This section describes the participatory design of our proposed navigation robot system. To do so, we adopted a scenario-based approach to consider the design of the system [55]. The scenario used for our study sessions is as follows: *A blind person is navigating in an unfamiliar building with a navigation robot to reach his/her destination. As the building is unfamiliar to the blind person, he/she acquires the route description from sighted passersby in the building.* We prepared two routes with different characteristics for this scenario-based study. To make the scenario more specific, we first conducted an interview with ten sighted passersby to investigate what route description they would convey to blind people. Then, we had a design session with five blind participants to understand what information would be useful for them to navigate an unfamiliar place with only the route description from sighted passersby.

Table 1: Top three pieces of information and the number of intersections described by sighted passersby.

Route	Top Three Information in Route Descriptions by Sighted Passersby		Number of Intersections Described	
	For Sighted Questioner	For Blind Questioner	For Sighted Questioner	For Blind Questioner
R1	Intersections with directions (70%) End of the hallway (40%) Doors along the way (40%)	Intersections with directions (100%) Distance to walk (60%) Where the wall are (60%)	Mean = 2.0 Median = 2.0	Mean = 2.4 Median = 3.0
	Intersections with directions (100%) Downhill corridor (60%) Existence of the library (50%)	Intersections with directions (100%) Downhill corridor (60%) Distance to walk (60%) Where the wall are (60%)	Mean = 2.9 Median = 3.0	Mean = 3.6 Median = 4.0

This study was approved by the institutional review board (IRB) of our institution, and an informed consent was obtained from every participant.

3.1 Routes For The Study

Route 1 (R1), shown on the left side of Fig. 2, is a narrow corridor in a building. The route has three intersections (indicated in blue dots) and its length is approximately 46 m. It also has furniture, rubbish bins, a kitchen, and signs indicating room numbers along the way. Route 2 (R2) is a wide corridor that spans two buildings. The route has four intersections and its length is approximately 166 m. The route also has a glass bridge, a library, an elevator, and signs indicating the names of the buildings along the way.

3.1.1 Interview with Sighted Passersby. We recruited ten passersby who knew both R1 and R2 and conducted a ten-minute interview with \$5 of compensation. For each route, we asked the participants to describe the route two times for two different cases: one for a sighted questioner, and another for a blind questioner. An example of a route description given by them is illustrated on the right side of Fig. 2.

Table 1 shows the top three pieces of information and the number of intersections described by the participants. We found that descriptions of intersections are always mentioned when assuming the questioner is blind, but could be omitted when assuming the questioner is sighted. Also, when we counted the number of intersections described in the route descriptions, both the mean and median values were higher when assuming the questioner is blind. Participants described the difference in explaining the routes to sighted and blind people as follows. C1: (Comment number 1) “It’s not possible for blind people to read any graphical signs. For example, signs like the map of the building, plates on the wall which have room numbers, and signs hanging from the ceiling. Without any of these details, the best information I can convey is about which directions to take when it’s needed.” These results indicate that sighted passersby describe which directions to turn at intersections particularly carefully to blind people.

3.2 Scenario-Based Study With Blind People

We recruited five blind participants (P01–P05 in Table 2). For each participant, we conducted a brief interview to gather their experience in navigating unfamiliar buildings. We then asked the blind

participants to provide information that would make them confident in unfamiliar buildings if they are navigating with a navigation robot system. To do so, an experimenter first explained R1 and R2 with the description given by the sighted passersby in the previous interview (Section 3.1.1), and then guided blind participants along the routes, asking them to think of the experimenter as a robot. During guidance, the experimenter explained indoor features, such as intersections, directional signs, textual signs, furniture (e.g., sofas, refrigerators, monitors, and tables), landmarks (e.g., a robot arm and elevators), facilities (e.g., libraries, kitchens, laboratories, and toilets), obstacles (e.g., rubbish bins and potted plants), and building structures (e.g., glass bridge and downhill corridor). After that, we asked the participants to rate each piece of information based on how confident it made them about their surroundings on a seven point Likert scale (1: Strongly disagree, 4: Neutral, and 7: Strongly agree). The interview took 75 minutes and each participant was compensated \$35.

3.2.1 Results. All the participants answered that they have experience navigating unfamiliar buildings (e.g., hospitals, airports, and universities) and agreed that they usually rely on sighted passersby. P04 mentioned their experience of navigating in unfamiliar buildings as follows. C2: “When I’m navigating in an unfamiliar building, the only information I have is the room number (of the destination). I don’t get any information about the route I have to take, so I have to rely on the first person I meet in the building to get there.” (P04)

Fig. 3 shows the provided ratings for how useful each indoor feature was. The result shows that intersections, directional signs, and textual signs are relatively the most useful information when navigating an unfamiliar building with a robot. Taking this result into account, we designed the functionality of our proposed system as discussed in the next section.

3.3 System Design

Based on the interviews with both the sighted passersby and the blind participants, we designed Pathfinder to have two modules: An intersection detection module and a sign recognition module.

3.3.1 Intersection Detection. Based on our interview with sighted passersby, the route description conveyed to blind people mainly consists of which turns to take at intersections (Section 3.1.1). Blind participants also indicated that intersections are one of the most useful information in unfamiliar buildings. Therefore, we designed

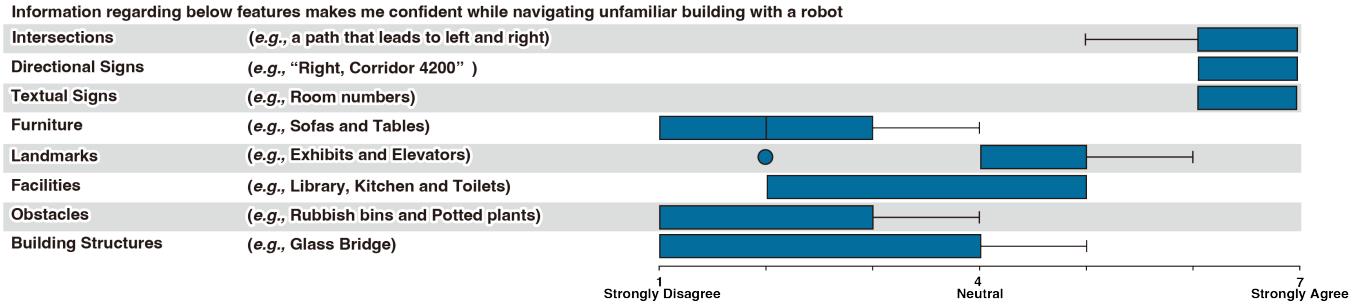


Figure 3: Questions and responses from our participatory study with five blind persons about useful indoor features. Responses are ratings on a seven-point Likert scale.

Table 2: Demographic information of the blind participants in our participatory study (P01–P05).

ID	Age	Age of onset	Gender	Navigation Aid
P01	74	32	Male	Cane
P02	67	10	Female	Guide dog
P03	38	0	Female	Guide dog
P04	76	0	Female	Cane
P05	59	0	Male	Guide dog

the system to be able to detect intersections together with their locations and shapes (*i.e.*, which way each intersection leads). Once the system detects an intersection, the system should convey the intersection’s shape to the user through audio feedback. By doing so, the blind user will be able to decide which way to go based on the original route information obtained from sighted passersby.

3.3.2 Sign Recognition. While intersection information may be sufficient to reach a destination, signs were found to make blind people more confident about their location. As indicated by the interview results (Section 3.2.1), the system should detect two types of signs, directional signs and textual signs. Directional signs are expected to help blind users confirm that they are on the correct route and help them make a decision at an intersection. Textual signs are expected to help blind users verify where they are and if they have reached their destination. As blind people cannot notice the existence of signs, the system should detect and notify the possible existence of a sign. Finally, as not all signs are relevant, the system should read out signs only if the user wants the system to.

3.4 Prototyping

We then prototyped the first version of Pathfinder based on the design derived from the first study session. The details of the implementation will be described in Section 4. Below, we first describe the handle interface and audio feedback of the prototype version of Pathfinder.

3.4.1 Employing Suitcase-shaped Robot. We adopted a suitcase-shaped robot, as its appearance would allow the user to blend into a surrounding environment, such as a building in a metropolitan environment [21, 27, 29] and where the study was conducted

in. The system will run alongside and slightly ahead of the user, enabling the system itself to be a protection by colliding with obstacles first [21]. Unlike quadruped robots that make a lot of walking noise [7], the form of a suitcase enables the system to take images from sensors stably with less motion blur [27, 28], which would allow the system to gain better recognition results of signs and intersections. While the weight of our current system is approximately 40 pounds, we expect that size and weight of the computers and sensors in the suitcase to get lighter and smaller, which will enable users to carry the system around more easily.

3.4.2 Map-less Navigation States and Interface. The user can provide instructions to the system via the four buttons attached to the system’s suit-case handle, consisting of a front button, a left button, a right button, and a back button (Fig. 4). These buttons function differently when Pathfinder is in the idle state or the moving state, as described below.

When the system is in the idle state (*e.g.*, pausing at an intersection or at its initial position) pressing the left/right button will instruct the system to face the next path which is on the left/right of the current facing direction while saying “*Turning left/right.*” Pressing the front button will switch the system to the moving state and instruct it to move to the next intersection, saying “*Going to the next intersection.*” Finally, pressing the back button will initiate Pathfinder’s sign recognition module while also saying “*Recognizing signs.*” Then, recognized signs will be read out after it finishes processing. An example of the audio feedback when three signs are recognized is as follows: “*There are three signs. 1. Forward, main lobby. 2. Right, Mechanical Engineering, and 3. Entering [proper noun] Hall.*”

In contrast, if the system is in the moving state, only two buttons, the front and the back buttons can be used. Pressing the front button will make the system switch between the *Next Intersection* mode and the *Hallway-end* mode. In the next intersection mode, the system will navigate until it reaches the next intersection, where it would stop and convey the intersection’s shape. We adopted the clock position to convey the shape of intersections as it is capable of conveying non-perpendicular turns, and can be easily generalized in public buildings. An example of the feedback when an intersection that leads to forward, left, and right is found is as follows: “*Found route to forward, two o’clock, and nine o’clock.*” Meanwhile, the hallway-end mode instructs the system to move forward until

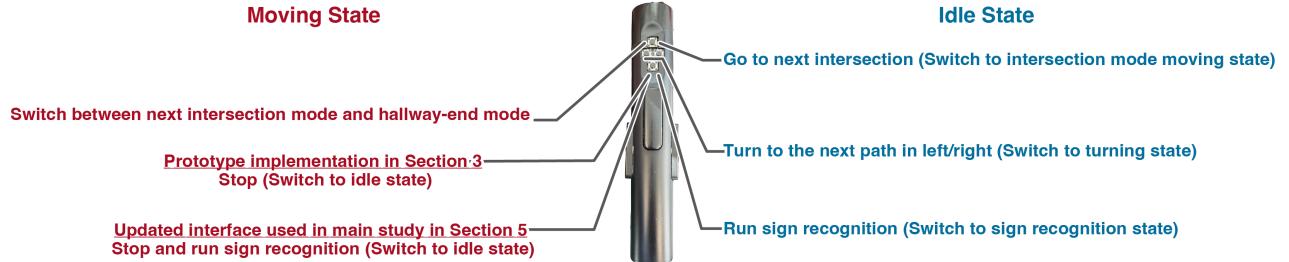


Figure 4: Handle interface of our PathFinder. It has four buttons, and each button is assigned different functionalities according to the system’s state as indicated.

it reaches the end of the hallway, ignoring all intersections along the path. This is useful when there are a lot of intersections and the blind user knows they will not be making any turns. Note that when this button is pressed, the system will say, “*Going to the next intersection/end.*” On the other hand, pressing the back button causes the system to stop and switch back to an idle state while also saying “*Stop.*” For simplicity of design, the system does not take any input while it is turning until the turn is done.

3.5 Design Iteration

After implementing the system, we conducted another session with the same group of five blind participants (P01–P05 in Table 2). The aim of the study was to improve the interface and functionality of the system. For each participant, we first introduced the system and asked them to use the system while walking along R1 and R2. We then interviewed the participants about areas for improvement. The interview took 75 minutes and each participant was compensated \$35.

All five participants generally agreed that the prototype version of the system will be helpful when navigating an unfamiliar building. Still, we received comments to improve the system when we asked for suggestions. Below, we list the major suggestions obtained from participants and a summary of updates made to the interface of the system.

3.5.1 Intersection shape should be conveyed using “left, right, forward, backward” terminology. Three participants mentioned that intersection shape should be conveyed with left and right, and not with clock position. As the two routes in the study contained only perpendicular intersections, we updated the audio feedback so that the system conveyed the intersection shapes using “left, right, forward, backward” terminology.

3.5.2 Position of textual signs should be conveyed, and fewer signs should be read. P04 pointed out that the system should convey the position of textual signs. He indicated that conveying its position is important, so that he knows where exactly the destination is if the system reads out room numbers. As such a feature is important for the last-few-meters problem [54], we updated the system to read out the distance and position of textual signs. Specifically, the system will convey the direction (e.g., left, left wall, right, right wall, and front) and distance to a textual sign.

In addition, three participants mentioned that the amount of information from signs was overwhelming as PathFinder read out

all signs in its field of view. In the study, there were several situations where it read out more than six signs. Therefore, we updated the system so that it will read a maximum of four signs. The system will first read directional signs, then textual signs if directions in the directional signs were fewer than four. When reading textual signs, the room number is read preferentially. Overall, we updated the audio feedback as follows: “*There are two directional signs. Left, corridor 4600, and right, (corridor) 4508 to 4533. Also, there is one textual sign saying room number 4521 to your front, 2.1 m ahead.*”

3.5.3 Associating information from directional signs with the turn direction at intersections. In the prototype version of PathFinder, the system only read out directional signs when the user initiates sign recognition. However, P01 pointed out that it would be helpful if the system could also read out where the system is turning to, from a nearby directional sign when a turn is being made at an intersection. For example, if a sign with “Right, corridor 4200” is recognized near an intersection that leads to the right and the user instructs the system to turn right, the system should say, “*Turning to the direction of Corridor 4200*” when making the turn. We implemented this feature on the system as it may increase its usability.

3.5.4 Merge stop button and sign recognition button. Three participants stated that the current interface of needing to press the back button twice to recognize a sign from the moving state is a cumbersome process. Hence, we updated the button layout of the handle interface so that sign recognition can be initiated even if the system is moving. Pressing the back button causes the system to switch back to the idle state and instructs the system to run the sign recognition algorithm, saying “*Recognizing sign.*”

3.5.5 Add “Take-me-back” functionality. P01 pointed out that it would be helpful if the system could take them back to the initial position where they started from, as such a task is difficult for blind people [18]. Therefore, we added a “Take-me-back” functionality to the system. As the system constructs a cost map and accumulates the map information over time, the system is able to maintain information about the initial position and return to it.

4 IMPLEMENTATION

This section describes our implementation of PathFinder based on the design considerations in the previous section. PathFinder requires a 360° LiDAR and an IMU sensor to construct a LiDAR map on the fly and detect intersections. It also requires a camera that

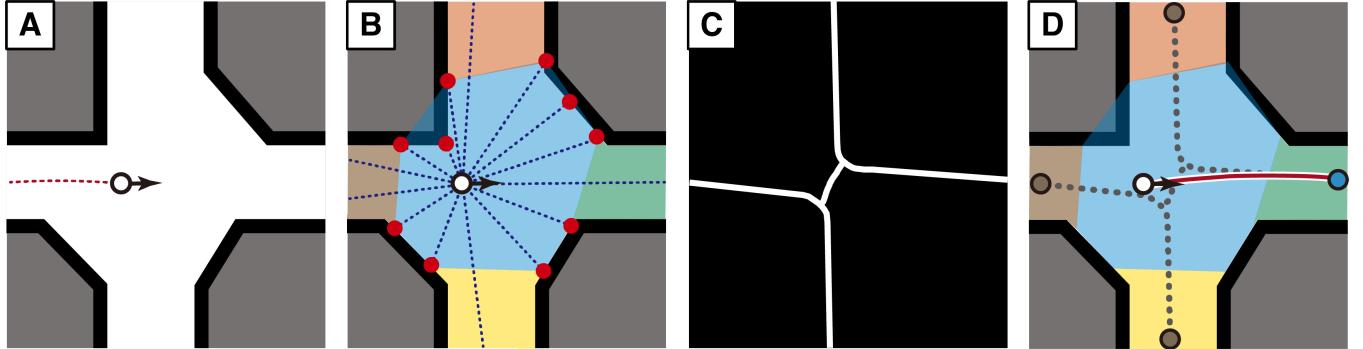


Figure 5: Steps for intersection detection: (A) Raw LiDAR map created by the system using SLAM (B) Detecting the closest convex hull similar to the method of Yang *et al.* [66] (C) Skeletonizing the regions outside of the convex hull to identify waypoints (D) Detected waypoints in the corridor regions, with the waypoint selected by the user shown in blue.

can capture indoor signs as clearly as possible. We chose the open source robot platform CaBot¹ as our base platform and extended it for PathFinder.

To navigate in a building without prebuilt maps, our system constructs a LiDAR map of its surrounding environment in real-time by using Cartographer², an open-source SLAM implementation. Our system operates an algorithm on this real-time map to find the paths that the system can navigate, and informs the user about intersections.

Although the system has an RGBD camera (installed at about 0.7 m from the ground), we attached a smartphone with a high resolution camera (iPhone 12 Pro) on an extendable stabilizer (Fig. 1-A, at 1.1 m from the ground) so that it could effectively capture indoor signs from a higher position and with a higher resolution compared to the RGBD camera. Here, the iPhone was chosen to enable fast prototyping of the system. Future versions may instead use an extra camera in an integrated manner. We selected a mini PC equipped with an NVIDIA RTX 3080 graphic board as the processing unit of the system so that it can perform sign recognition while also operating the CaBot. The iPhone and the PC are connected via Bluetooth to provide audio feedback and via Wi-Fi local network for fast image transmission. Below, we describe the details of our intersection detection and sign recognition algorithms.

4.1 Intersection Detection

To detect intersections, we use the method proposed by Yang *et al.* [66]. The system first extracts waypoints for where the system can move from the latest LiDAR map. If there are waypoints found on the left or the right sides of the system, this means that the system has detected an intersection. In such a case, the system will stop and inform the user of the detected directions so that the user can determine which way to turn. In addition, the algorithm is further used to make the system move in the middle of the corridor for their safety [36], and keep the heading direction of the system facing the corridor.

Our overall algorithm to extract waypoints runs at about 10 Hz, and can be described as follows.

¹<https://github.com/CMU-cabot/cabot>

²https://github.com/cartographer-project/cartographer_ros

- (1) The system extracts a region of size $20 \text{ m} \times 20 \text{ m}$ around the system from the latest LiDAR map (Fig. 5-A).
- (2) Based on the extracted map, the system applies the following steps proposed by Yang *et al.* [66] to detect all the corridors leading to the intersection: (Fig. 5-B).
 - (a) The system samples points starting from the surrounding obstacles (e.g., walls) within a certain radius (8 m) of the system, at constant angular intervals (10°), as shown with red circles.
 - (b) The system computes a convex hull using the sampled obstacle points, shown as the blue region.
 - (c) The system extracts the obstacle-free areas from outside the convex hull as the corridor region(s) (colored regions outside the convex hull).
- (3) The system extracts the topology of the middle of the detected corridors by skeletonizing the image of the LiDAR map (Fig. 5-C).
- (4) Finally, the system assigns the furthest point on the topology from the system in each corridor region as waypoints that the user can instruct the system to move to (Fig. 5-D).

Note that if the system does not detect any corridor on either side of the system, then the system will continue to move forward until it is stopped by the user or until it finds an intersection or the end of the hallway.

4.2 Sign Recognition

Indoor signs are often placed on the ceiling or on the walls at a distance, and they appear very small in the images. To recognize texts and arrows accurately, we need an OCR model and arrow detection model which require about 5 seconds on average to process on our PC (including communication times), which may be too long for a blind user to wait for [54]. Therefore, we implemented a sign detection module. The sign detection module informs the user about the possible existence of a sign in real-time, without fully recognizing the sign, and the user can initiate sign recognition if they think the sign may benefit them to reach the destination. Here, we describe the sign detection module and the sign recognition module.

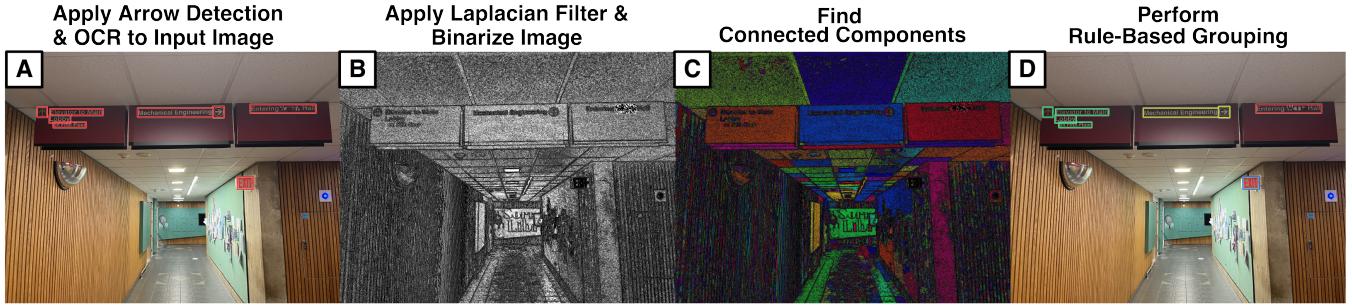


Figure 6: Our sign recognition module consists of four steps. The system first applies arrow detection and OCR (A), followed by Laplacian filtering and image binarization (B). Then the system finds the individual regions in the image using connected component analysis (C), followed by a rule-based grouping to associate the arrows and recognized text (D).

4.2.1 Sign Detection Module. The module is implemented as an iPhone app by using the Optical Character Recognition (OCR) model from the iOS Vision API³. The model detects texts approximately at five Hz on the iPhone, and the depths of the detected texts are then measured using its LiDAR sensor. If a piece of text is detected in five consecutive frames and is found to be within 6.5 m, the system notifies the user of the possible existence of a sign by saying, “*There might be a sign.*” If the user then chooses to initiate the full sign recognition, the system will then send the image taken by the smartphone to the PC on the system via the local network. The system will not inform the existence of a sign which is within 5 m from the previously detected sign.

4.2.2 Sign Recognition Module. When the user initiates the sign recognition by pushing the back button, the system commands the iPhone to send an RGB and depth image to the sign recognition server running on the system’s PC via the local Wi-Fi network. Once the image is sent to the server, the system runs OCR using the EasyOCR Python library⁴ which is more accurate than the iOS OCR, and runs YOLOv5 [15] object detection model to detect the arrows in the image. We trained the object detection model to detect and classify eight categories of arrows (four horizontal and vertical directional arrows, and four diagonal directional arrows). An example illustration of the detection result is shown in Fig. 6-A. According to the system design (Section 3.3.2), it is necessary for the system to recognize both directional and textual signs. To do this, the system needs to associate the detected arrows with the detected texts to recognize a directional sign, while also separating the texts from the arrows if there is a textual sign. To do this, the system assumes texts and arrows can be grouped together if they have a similar background color. Below, we describe the steps of our grouping algorithm for signs.

- (1) The system first detects edges in order to separate regions with different background colors. This is done by applying a Laplacian filter to each of the RGB channels, then for each pixel, selecting the highest value over the channels to create a single-channel image. This will result in an image where pixels with high values represent edges, while those with low values represent non-edges.

- (2) The system then obtains a binarized image by assigning 0 to pixels whose values are higher than a pre-determined threshold value, and 1 to all other pixels (Fig. 6-B).
- (3) The system applies the connected component labeling algorithm to the binarized image to determine the regions with similar background colors, and obtain a region label for each pixel (Fig. 6-C).
- (4) The arrows and texts whose bounding boxes have the same region label are then grouped together (Fig. 6-D).

As a result, the grouped sets of arrows and texts will be obtained. Note that a text may not necessarily have to be grouped with arrows, as the algorithm only considers the background color of each bounding box. Note that in practice, a grouped set may include multiple arrows and texts. Deciphering matches between multiple arrows and texts would involve a more complicated algorithm outside of the scope of this study. So we only consider signs where each text corresponds to only one arrow. The system calculates the euclidean distance between the center of the bounding boxes of arrows and texts, and groups texts with arrows that have the smallest distance between them. Finally, using the LiDAR sensor of the iPhone 12 Pro, the system removes signs that are further than 6.5 m from the recognition results, so that only signs with accurate detection results are conveyed to the user.

5 MAIN STUDY

The main goal of this study is to understand the effectiveness of our complete system and how well it can assist blind people. For comparison, we used a system with prebuilt maps as the topline reference, which we assumed to provide the best possible navigation experience. We conducted the main study with seven blind participants (P06–P12 in Table 3). This study was approved by the IRB of our institution, and an informed consent was obtained from every participant. Each study took 150 minutes and participants were compensated \$70.

5.1 Tasks and Conditions

We asked the participants to navigate R1 and R2 (Section 3.1) from the starting points to the destinations. We prepared two conditions, one where participants used PathFinder and the other where they used the topline system *i.e.*, a system that uses prebuilt maps. We

³https://developer.apple.com/documentation/vision/recognizing_text_in_images/

⁴<https://github.com/JailedAI/EasyOCR>

Table 3: Demographic information of blind participants in our main study (P06–P12). Also listed are the main study participants’ normalized task completion times, and the number of times they asked for the route during navigation. *1: normalized as they chose a slower speed. *2: normalized as the previous user’s topline setting was used accidentally

ID	Age	Age of onset	Gender	Navigation Aid	SUS	Normalized task completion time [sec]				Times asked for route	
						Route 1		Route 2		Route 1	Route 2
						PathFinder	Topline	PathFinder	Topline	PathFinder	
P06	68	0	Male	Cane	87.5	398.3	59.2	491.5	240.7	4	6
P07	69	49	Female	Cane	95.0	174.5 *1	66.3 *1	919.8 *1	234.7 *1	0	3
P08	63	0	Female	Cane	72.5	854.3 *1	66.1 *1	914.9 *1	239.0 *1	2	6
P09	63	56	Male	Cane	80.0	260.4	66.7 *2	601.5	230.8	0	3
P10	74	0	Female	Cane	92.5	223.6	60.1	404.2	259.9	0	2
P11	63	3	Female	Cane	90.0	147.3	60.9	350.3	240.6	0	3
P12	50	1	Male	Guide dog	52.5	163.1	63.0	573.0	232.1	3	2
Mean	64.29				85.25	317.36	63.16	607.88	239.69	1.29	3.43
±SD	±7.52				±15.74	±251.68	±3.20	±228.85	±9.78	±1.70	±1.80

did not include the condition of using only the regular aid for safety concerns, as blind people usually do not navigate in unfamiliar buildings without an assistant [14, 25].

5.1.1 PathFinder. For the condition using PathFinder, we first described the route to the destination to the participants, then asked them to navigate to the destination. They were allowed to ask the experimenter for the route during the task if they needed it, in which case the experimenter would describe the route again from their current position to the destination. The number of times they asked for the route description is reported in Table 3. The experimenter intervened in the study only if the participants turned at the wrong intersection or the system malfunctioned.

5.1.2 Topline System. For the topline system, we used software⁵ that can navigate blind people in buildings with prebuilt maps, which run on the same suitcase-shaped robot. The system also has a handle interface with three vibrators, on the right, left, and top of the handle. To operate the topline system, we constructed full prebuilt maps of R1 and R2, which are annotated with POIs such as intersections, building names, and facility names. When the system is initiated, it localizes itself in the prebuilt map using BLE beacons, which are attached to the building. In the study, the experimenter manually set the destination remotely via a smartphone application, and the system started navigating to it once the user pressed a button on the handle. During navigation, the system reads out annotated POIs along the way, and while turning the vibrator which is in the direction of the turn vibrates. When the navigation ends, the topline system indicates that they have arrived at the destination. Note that the duration to navigate through R1 and R2 does not depend on a participant, but mostly on the social context of the building (e.g., a crowd of people), as the topline system automatically navigates to the destination at a constant speed.

5.2 Procedure

We first introduced PathFinder and explained its map-less navigation feature and conducted a 30-minute training session. In the session, we adjusted the speed of the system to 0.75 or 0.50 meters per second (m/s) based on each participant’s walking speed. The adjusted speed was used for all tasks and scenarios for the participant. Then, the participants were asked to navigate through R1 and R2 with PathFinder based on the route descriptions we gave them. At the end of each route, they were also asked to use the “Take-me-back” functionality and go back to the initial position. Next, the participants were asked to navigate R1 and R2 again using the topline system. We did not counter-balance the order of the PathFinder and the topline systems because we did not want to induce any route learning in the user by using the topline system prior to using PathFinder. If we counter-balance the order of conditions and let several participants perform the topline system first, the task using PathFinder will be significantly easier due to the prior knowledge of the route walked with the topline system, as PathFinder will require users to memorize the route description while topline system does not. The tasks were video recorded so that it allows us to complement quantitative results. After the navigation, we asked the participants to answer a set of questions (Q1–8 in Fig. 7) on a seven-point Likert scale (1: Strongly disagree, 4: Neutral, and 7: Strongly agree). We asked Q1–Q3 three times, to compare their experience when using their regular aid (*i.e.*, canes or guide dogs), PathFinder or the topline system. Then we asked Q4–8 regarding the usability of PathFinder. Finally, we asked participants to rate PathFinder using system usability scores (SUS) [4] and gathered open-ended questions for qualitative feedback.

5.3 Metrics

We used two metrics to evaluate and analyze the usage of the proposed system.

5.3.1 Normalized Task Completion Time. We measured the time taken to complete each task. We started the timer when the participants pressed the button to initiate the system at the starting

⁵<https://github.com/CMU-cabot/cabot>

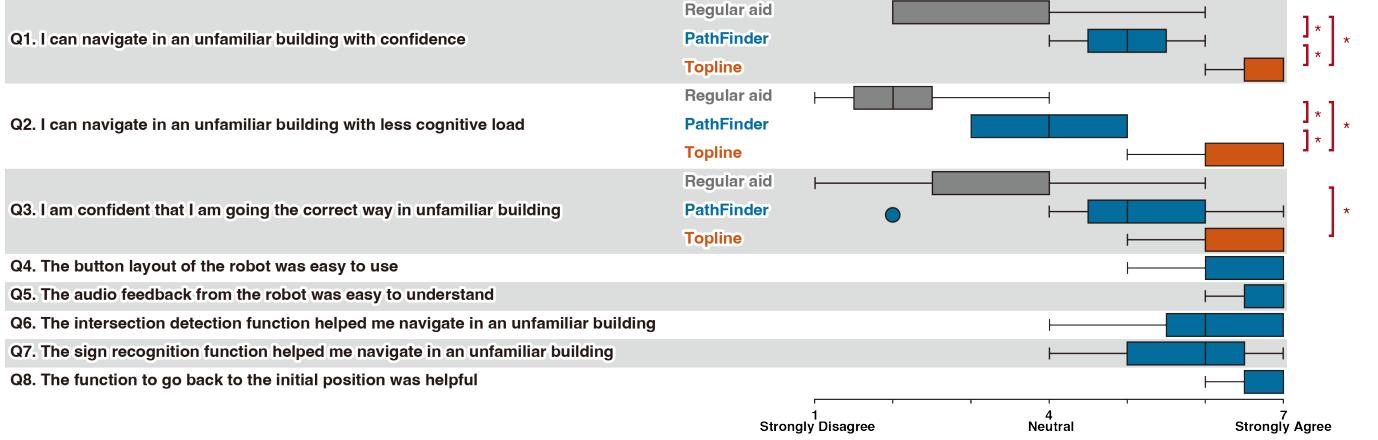


Figure 7: Questions and responses from our main study with seven blind participants. Responses are rated on a seven-point Likert scale. Responses marked with * indicate a significant difference between the systems when applying the Wilcoxon signed-rank test ($p < .05$).

point. The timer was stopped when the participants verbally indicated that they had arrived at the destination and were 5 m within the destination. As some of the participants used a slower speed (0.5 m/s), we normalized their task completion times such that all times correspond to 0.75 m/s. This is calculated as $T_m \times \frac{0.5}{0.75} + T_s$, where T_m is the total duration the user and the system are moving together, and T_s is the duration for which they are standing still (which is thought of as the users' decision making time).

5.3.2 Performance of Intersection Detection and Sign Recognition. To evaluate the performance of the system, we measured two metrics based on the logs recorded by the system during the study. For intersection detection, we classified a detection result into four cases: (1) *correct detection* when the system correctly detected the intersection's shape, (2) *partially correct detection* when the system detected the turn direction, but missed some directions which are not a direction of turn, (3) *failed detection* when it did not detect the turn direction, and (4) *false positive detection* when the system detected an intersection where there was none (*i.e.*, at straight corridor).

For sign recognition, we classified detection results into four cases: (1) *correct and relevant recognition* where the detection result contained information to reach the destination, (2) *correct but irrelevant recognition* where the result was correct but contained only irrelevant information, (3) *null recognition* where the system did not recognize any sign, and (4) *wrong recognition* where the system was unable to recognize sign correctly as either arrow detection, OCR or the grouping algorithm failed. Note that the number of correct and useful recognitions and correct recognitions may vary depending on the route and destination.

6 RESULTS

6.1 Overall Performance

6.1.1 Normalized Task Completion Time. Table 3 shows the results of the task completion time. Statistical analysis using the Wilcoxon

signed-rank test revealed that tasks with PathFinder took significantly longer to complete compared to those with the topline system ($p < .05$ for both R1 and R2). Below, we summarize four reasons why our proposed system took a longer time to complete a task. 1) Our system took extra time because PathFinder stopped at each intersection. 2) Participants took time to recall and determine the direction to turn. 3) Our system required time to run sign recognition each time it was initiated. 4) There were four times that participants turned at a wrong intersection (Occurred twice for P08, and once for P06, P07, and P09) and it took extra time for them to return to the correct route. 5) P08 was particularly confused about the interface of the system and took extra time to complete the tasks.

6.1.2 Subjective Ratings. Fig. 7 shows the results for Q1–8. Statistical analysis using the Wilcoxon signed-rank test revealed that participants felt that PathFinder is significantly better compared to their regular aids for Q1 and Q2 ($p < .05$). The same test also revealed that the topline system received significantly better ratings than the proposed system for Q1 and Q2 ($p < .05$). For Q3, there was no significant difference neither between PathFinder and regular navigation aids ($p = .14$) nor between PathFinder and the topline system ($p = .09$). Finally, Table 3 shows the SUS scores given by each participant.

6.2 Performance of Intersection Detection and Sign Recognition

PathFinder detected intersections 108 times in total throughout the whole study. There were 61 correct detections, nine partially correct detections, five failed detections, and 33 false positive detections. The partially correct and false positive detections did not harm the performance of the task, as it still detected the way the blind participants have to go. The false positive detections mainly occurred when navigating through a glass bridge in R2. The system detected false intersections on this straight bridge, as glasses are transparent from the LiDAR sensor. Two failed detections out of five occurred at

intersection A and intersection B (Fig. 2) when P07 was navigating R2 in a significantly crowded situation. In intersection A, the path was crowded in front of the elevators in R2, and the system did not detect the path that leads to the right. Also, the system detected intersection B, which leads to the front and left, as an intersection that leads to left and right, as the crowd affected the orientation of the system to face in between the two paths. The other three failed detections were related to system errors.

Participants initiated sign recognition 62 times in total while navigating R1 and R2. Throughout the whole study, P06–12 initiated sign recognition 1, 16, 22, 10, 4, 4, and 5 times, respectively. There were 27 correct and relevant recognitions, 12 correct but irrelevant recognitions, 13 null recognitions, and ten wrong recognitions. Null recognition occurred when the participant initiated sign recognition where there was no sign. They initiated the sign recognition as the sign recognition module notified them of the existence of signs, but by the time they press the back button, the sign was out of the camera's field of view. Also, participants who took the wrong path in their tasks tend to use the sign recognition function much more times, even when the system did not notify the possible existence of signs (P07 and P08).

6.3 Video Observation

6.3.1 Confusion When Using Sign Recognition. During the study, we observed an occurrence where the interface of the PathFinder confused two participants. P08 and P09 performed sign recognition on a directional sign (e.g., “Left, Corridor 4200, and right, Corridor 4100”) at a spot before the desired intersection (about 1-2 m). After listening to the feedback, they immediately pressed the left/right button as they thought it would take them to the next path. As the system was before the intersection and had not detected it yet, the system turned backward. While P09 was able to recover after a short time of confusion, P08 was quite confused with the occurrence, because the system did not move as she wanted it to. **C3:** *“It confused me when it was telling me that a sign was available, but I am not at an intersection... like when it was telling me to take the right turn to Destination 2, but I went down the wrong corridor because I wasn't yet at the intersection to make the turn.”* (P09)

6.3.2 Navigation Error Occurred When Intersection Detection Failed. The intersection detection error in intersections A and B (Figure 2) both occurred when P07 was performing the task. In intersection A, the system was able to detect the shape of the intersection correctly when the system ran the intersection detection algorithm again after the crowd was eased. When the detection failure in intersection B happened, she instructed the system to turn in the wrong direction instead of going forward. She realized that she had turned in the wrong intersection after reaching the dead end of the hallway and managed to recover back to intersection B.

6.4 Qualitative Feedback

6.4.1 Positive Feedback. Through the study, we received many comments indicating that they found the functions of PathFinder useful. All participants found the intersection detection feature of the system helpful, as it can find an intersection more accurately and quickly compared to their usual navigation aid. **C4:** *“(To find an*

intersection) I would have to stay close to the walls and feel with my cane, which is not always possible. But with the robot, I can quickly find the intersection without being close to a wall.” (P10)

In addition, the sign recognition feature of the system was generally appreciated by the participants. Participants described the advantage of textual signs and directional signs as follows: **C5:** *“I really liked the sign recognition. If I'm in an unfamiliar place like an airport, and it read out the gate number (textual sign), I would immediately know where I am and that there are more gates”* (P06) and, **C6:** *“Yes (directional signs are useful), it gives confirmation which is important, and you can get confidence about where you are going.”* (P11)

Also, the “Take-me-back” functionality was appreciated by all seven participants from the main study. **C7:** *“It was very useful and felt almost like the with-map robot. It would be very convenient when I'm in an office building and want to quickly find my way back to the entrance. I can also envision the ability to key in my favorite spots while I'm exploring and then trust the robot to directly take me to those spots while I'm navigating the second time.”* (P09)

When we asked whether PathFinder is acceptable compared to the topline system, all participants agreed and were appreciative of its advantage that it could be used in more places compared to the topline system: **C8:** *“If a map is available, it is definitely the most useful. But maps are not available everywhere. However, even without map information, the robot is very useful because it reads out information around me and that lets me find my way.”* (P07) While the topline system generally received higher ratings in Q1–3, two participants still found the controllability of PathFinder to be better than the topline system. **C9:** *“With PathFinder I felt like I was in more control as I had the ability to get feedback from it at every intersection and use my own judgment. But with the topline system, I just had to let it do its thing, which I'm not fully trusting of.”* (P09)

6.4.2 Negative Feedback. Participants were confused when the system indicated the possibility of the existence of a sign, without indicating what types of signs they were: **C10:** *“I felt like the system wasn't picking up all the signs, and when it did, it didn't say what kind of sign it was right away. It is important for some kind of signs to be known right away, like fire exits, instead of having to ask for it each time.”* (P07)

P12 rated the SUS score the lowest. When we asked him for the reason, he commented that guide dogs would be a better tool for their secondary travel, as follows: **C11:** *“I think once I learn the layout of a building, I will be able to navigate much faster with my guide dog than with the robot.”* (P12)

6.4.3 Comparison with Guide Dogs. When we asked P12, who is a guide dog user, about the difference between the proposed system and the guide dog, he mentioned two points, that guide dogs may miss an intersection unless the user is aware of it, and guide dogs do not remember how to return to previous places, as follows: **C12:** *“With guide dogs, I have to know when to turn. But the robot will tell me, so I won't miss the intersection... The guide dog doesn't always know how to go back. I have to remember the route and teach the dog. With the robot, I would just have to hit the button, and it would take me back. I didn't have to remember the route by myself.”* (P12)

7 DISCUSSION

7.1 Comparison with Topline and Regular Aids

PathFinder may be considered to be an option that could be in between the topline system and regular aid for its performance, functionality, and usable area. The confidence scores were significantly higher than the regular aids, and lower than the topline system (Fig. 7–Q1). The ratings for cognitive load were significantly higher than the regular aids (lower cognitive load), but lower than the topline system (higher cognitive load) (Q2). As for the usable area of the aid, regular aids are the largest, as there are routes in which PathFinder cannot be used, such as routes with steps or rough surfaces. The topline system has the smallest usable area, given the requirements of the prebuilt maps. The task completion time for PathFinder was significantly longer than the topline system. This result is expected since it is necessary for each participant to stop at each intersection to choose a direction. The topline system can announce POIs, and PathFinder can do this partially using its sign recognition module. The regular aids do not have the capability to announce POIs.

In short, PathFinder can be a unique “in-between” option for blind people. The topline system can provide highly reliable and robust navigation while announcing POIs, but its usable area is small. Regular aids, on the other hand, can be used in most environments, but their reliability is low. PathFinder’s approach can realize novel scenarios such as visiting an unfamiliar indoor public space and navigate without prior preparation and sighted companion. We still have a long way towards the goal, especially for the recognition capabilities, but we believe this study showed a new possible solution.

7.2 Usability

The system received favorable ratings with medians of 6 and 7 on the Likert scale for usability scores (Q4: 6, Q5: 7, Q6: 6, Q7: 6, Q8: 7). The ratings for the interface show the success of our participatory design process. Also, the median ratings for intersection and sign detection were both 6. One of the reasons why neither of the scores was 7 may be the interface design related to sign recognition. PathFinder requires the participants to select directions via buttons after reaching an intersection, not after the announcement of directional signs (Section 6.3.1, and C3). P08 was particularly confused with the occurrence of accidentally turning backward after listening to information about directional signs that the system recognized (Section 6.3.1), and rated the score for Q3 (confidence in the walking direction) with a 4 for the regular aids and a 2 for PathFinder. Although all other participants rated PathFinder to be higher than the regular aids for Q3, there was no significant difference between PathFinder and the regular aids. Also, there was no significant difference between PathFinder and the topline system for Q3 as P09 rated PathFinder higher (PathFinder: 6, Topline: 5) and P12 rated both the same (both 7). To prevent the occurrence of accidentally turning backward, the system can ask the user for verification when turning before an intersection so that it can prevent the unnecessary backward turn (e.g., “*You are trying to turn before an intersection. Are you sure you want to make a turn?*”).

7.3 Controllability

Two participants indicated that PathFinder was even better than the topline system in terms of the trust, as they gained more controllability with the system (C9). They preferred a more controllable system in spite of the time disadvantage. The current navigation robot systems are designed by putting weight on automation more than controllability. It may be possible to integrate some controllability into the topline system, such as a route-choice interface at an important intersection toward a destination. At this moment, it is not clear how we can balance automation and controllability, but future navigation robot systems for blind people may wish to consider this as part of the design.

7.4 “Take-me-back” Functionality and Gradual Map Creation

All participants unanimously agreed that the “Take-me-back” function was useful, as returning to the entrance is generally challenging for blind people, especially in unfamiliar buildings. P12 indicated that guide dogs could not complete such a task, as guide dogs do not remember an unfamiliar building layout (C12). This feature also leads toward the discussion of the gradual creation of maps equivalent to prebuilt maps, by blind users themselves. It may accumulate the necessary data to enable with-map navigation for routes to typical destinations at the building if the system preserves constructed LiDAR map data and provides an interface for blind people to annotate the LiDAR map with visited routes and POIs (C7). Such a feature can be a game changer for navigation systems by paving the way toward city-wide maps.

7.5 Possible Improvements for PathFinder

The current intersection detection algorithm works robustly only on limited situations and buildings. For example, in the main study, PathFinder sometimes misinterpreted the shape of an intersection (Section 6.2). The system detected an intersection in R2 as a dead end, because the corridor was packed with a crowd of people. In such situations, it may be necessary for the system to determine the level of congestion, and emit a path-clearing sound, so that people may move out of the way for the system to eventually find an intersection [27]. In addition, the system made multiple false positive detections at the glass bridge in R2, as the system failed to extract the shape of the bridge to a LiDAR map because of the transparency of glass to LiDAR. Besides, the system can not detect intersections at buildings with open spaces such as open halls or wide corridors, such as an atrium, because the current implementation has an assumption of the size of an intersection (Section 4.1). It is also difficult for PathFinder to define a local destination in an open space, resulting in the system to go in a random direction, as the topology extraction is likely to fail. To increase the generalizability of PathFinder, it is necessary to consider various environments (e.g., open hall, hallway with open doors, glass bridge, and a wide corridor) to improve the intersection detection algorithm, for example by also using RGB camera as well as LiDAR sensor.

As for the sign recognition feature, while it was appreciated by the participants (C5 and C6), the function that conveyed the existence of the signs was insufficient (Section 6.2). As P07 pointed out, the system needs to detect types (*i.e.*, whether it is directional

or textual) or relevancy (*i.e.*, whether it contains necessary information) of signs before notifying the user and running the full sign recognition (C10). As determining the types of signs will need to go through a sign recognition which takes 5 seconds to process, one solution is to determine the relevancy of a sign by using the information which the user can input prior to their travel [65]. The user may input some keywords, such as the name of the destination or room number, so that the relevancy can be determined by only the result of OCR.

7.6 Limitations

7.6.1 Limitation of Study Design. This study design had several limitations. Firstly, the study was conducted on only two routes in our institution's buildings. The performance of intersection detection and sign recognition may vary in other environments, therefore, a comprehensive evaluation of those functionalities is one of the important future works. We also made some assumptions with our choice of the study's environment to prototype the PathFinder system, such as routes without any steps and floor transitions. Features such as elevators or stairs may have been rated higher in the participatory study if the route contained floor transitions. Therefore, it is necessary to conduct a further study by considering routes with various features. Secondly, participants did not navigate through the route with their regular aid. The lack of empirical comparison with the regular aid may have led to influencing their critique of the proposed system (*e.g.*, some of them may have completed the route with their cane and rated their regular aid better in Likert questions). In addition, there were limitations regarding the recruitment of participants. First, we were not able to recruit younger participants from the target population, and the number of guide dog users recruited in the main study was not sufficient. Also, while several participants participated in user studies for the first time in our institution, others have not. For whom have participated in the user study in our institution in the past, there may have had a positive bias to our study.

7.6.2 Limitation of Form Factor. In this study, we used a wheeled robot for its advantages (third paragraph of Section 1), but the form of the wheeled robot may induce several limitations on real-world usage. Currently, its battery life is limited to only 2.6 hours of deployment, which is insufficient for long journeys. PathFinder may not be able to navigate through uneven terrains or outdoor environments due to the small size of the wheels. In such cases, we should consider using larger wheels and stronger motors. Also, users would have to carry the suitcase when navigating through the stairs, which could be physically demanding. We believe the gradual improvement of weights and capability of each component used would ease these issues. As we did not consider other devices which are preferred by blind people such as wearable devices or single smartphones in this study, these devices may potentially serve as a better solution for map-less navigation. Future work should explore these alternatives as well.

In addition, there are also other problems that have to be solved. Firstly, in our implementation, the smartphone and the computer of the system were connected through Bluetooth. Interference by the surrounding usage of Bluetooth may cause the connection to be disturbed. In actual deployment, a robust way to connect

the smartphone and the computer is required. Secondly, as the system keeps recording the surrounding environment, the privacy of pedestrians may be compromised. Although it has been shown that sighted people accept the usage of RGB images as long as they are used for assisting blind people to some extent [29], this problem should be carefully considered when actually deploying the system in the real world. Thirdly, PathFinder will not be able to navigate in a congested space. This is a well-known issue that is still being studied in robotics [69]. Finally, surrounding people would not be able to determine easily whether users of PathFinder have a disability as users do not hold their regular aid, thus the users may not be able to get help if needed. In such cases, the system should include some functionalities that can notify surrounding pedestrians to provide assistance to blind users. However, note that while some blind people have mentioned that they are afraid they will not be able to get help from surrounding people as much, some blind people prefer to blend into the environment without being known that they are blind [29].

8 CONCLUSION

This paper presents PathFinder, a map-less navigation system that navigates blind people to their destinations in unfamiliar buildings. We adopted a participatory design approach, and first investigated types of useful information for blind people when navigating an unfamiliar building. We found that intersections, directional signs, and textual signs provide the most useful information. Based on these insights, we designed and developed our first map-less navigation system prototype. The system uses a map-less navigation algorithm that navigates the user to the next intersection or end of the hallway. It also uses a sign recognition algorithm to read out directional and textual signs to make the user more confident in their navigation. Then we went through a design iteration to improve the interface with five blind people. Based on the comments obtained through the study, we updated how the intersections and signs are read out and the button layout of the handle interface. Also, a “Take-me-back” functionality was added to PathFinder. Finally, we conducted the main study, asking the participants to navigate an unfamiliar building using PathFinder. Using the system, all the participants were able to navigate to the destination with increased confidence and less cognitive load compared to their usual navigation aid. Although participants generally rated the topline system, *i.e.*, a system which uses a prebuilt map, higher than PathFinder, all participants agreed that it is still an acceptable aid as it can be used in more buildings, and two of them considered the controllability of PathFinder as an advantage. Overall, PathFinder could be used as an “in-between” option between the regular navigation aid and the topline system, as it provides a good trade-off in terms of usable area and functionality. For future work, we aim to redesign the system by improving functionalities and considering the balance between automation and controllability of the system.

ACKNOWLEDGMENTS

This work was supported by JST-Mirai Program (JPMJMI19B2).

REFERENCES

- [1] Mouna Afif, Yahia Said, Edwige Pissaloux, Mohamed Atri, et al. 2020. Recognizing signs and doors for Indoor Wayfinding for Blind and Visually Impaired Persons.

- In 2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSiP). IEEE, Los Alamitos, CA, USA, 1–4. <https://doi.org/10.1109/ATSiP49331.2020.9231933>
- [2] David Alejo, Gonzalo Mier, Carlos Marques, Fernando Caballero, Luis Merino, and Paula Alvito. 2019. SIAR: A Ground Robot Solution for Semi-autonomous Inspection of Visitable Sewers. *Advances in Robotics Research: From Lab to Market: ECHORD++: Robotic Science Supporting Innovation* 132 (2019), 275. https://doi.org/10.1007/978-3-030-22327-4_13
- [3] Mrouj Almuhanji and Ching Y. Suen. 2022. A Complete Framework for Shop Signboards Detection and Classification. In 2022 26th International Conference on Pattern Recognition (ICPR). IEEE, Piscataway, NJ, USA, 4671–4677. <https://doi.org/10.1109/ICPR56361.2022.9956399>
- [4] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7. <https://doi.org/10.1201/9781498710411-35>
- [5] Hsuan-Eng Chen, Yi-Ying Lin, Chien-Hsing Chen, and I-Fang Wang. 2015. Blind-Navi: A navigation app for the visually impaired smartphone user. In *Proceedings of the 33rd annual ACM conference extended abstracts on human factors in computing systems*. ACM, New York, NY, USA, 19–24. <https://doi.org/10.1145/2702613.2726953>
- [6] Qiang Chen, Yinong Chen, Jinhui Zhu, Gennaro De Luca, Mei Zhang, and Ying Guo. 2020. Traffic light and moving object detection for a guide-dog robot. *The Journal of Engineering* 2020, 13 (2020), 675–678.
- [7] Qihe Chen, Luyao Wang, Yan Zhang, Ziang Li, Tingmin Yan, Fan Wang, Guyue Zhou, and Jianguo Gong. 2022. Can Quadruped Navigation Robots be Used as Guide Dogs? <https://doi.org/10.48550/ARXIV.2210.08727>
- [8] Xiaoxue Chen, Lianwen Jin, Yuanzhi Zhu, Canjie Luo, and Tianwei Wang. 2021. Text Recognition in the Wild: A Survey. *ACM Comput. Surv.* 54, 2, Article 42 (2021), 35 pages. <https://doi.org/10.1145/3440756>
- [9] Zhichao Chen and Stanley T Birchfield. 2009. Qualitative vision-based path following. *IEEE Transactions on Robotics* 25, 3 (2009), 749–754. <https://doi.org/10.1109/TRO.2009.2017140>
- [10] Angela Constantinescu, Karin Müller, Monica Haurilet, Vanessa Petrausch, and Rainer Stiefelhagen. 2020. Bring the Environment to Life: A Sonification Module for People with Visual Impairments to Improve Situation Awareness. In *Proceedings of the 2020 International Conference on Multimodal Interaction*. ACM, New York, NY, USA, 50–59. <https://doi.org/10.1145/3382507.3418874>
- [11] Guilherme N DeSouza and Avinash C Kak. 2002. Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence* 24, 2 (2002), 237–267. <https://doi.org/10.1109/34982903>
- [12] Andrés A Diaz-Toro, Sixto E Campaña-Bastidas, and Eduardo F Caicedo-Bravo. 2021. Vision-Based System for Assisting Blind People to Wander Unknown Environments in a Safe Way. *Journal of Sensors* 2021 (2021). <https://doi.org/10.1159/2021/6685686>
- [13] Barzin Doroodgar, Yugang Liu, and Goldie Nejat. 2014. A learning-based semi-autonomous controller for robotic exploration of unknown disaster scenes while searching for victims. *IEEE Transactions on Cybernetics* 44, 12 (2014), 2719–2732. <https://doi.org/10.1109/TCYB.2014.2314294>
- [14] Christian Engel, Karin Müller, Angela Constantinescu, Claudia Loitsch, Vanessa Petrausch, Gerhard Weber, and Rainer Stiefelhagen. 2020. Travelling More Independently: A Requirements Analysis for Accessible Journeys to Unknown Buildings for People with Visual Impairments. In *ASSETS '20*. ACM, New York, NY, USA. <https://doi.org/10.1145/3373625.3417022>
- [15] Glenn Jocher et al. 2021. *ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support*. Zenodo. <https://doi.org/10.5281/zenodo.5563715>
- [16] Navid Fallah, Ilias Apostopoulos, Kostas Bekris, and Eelke Folmer. 2012. The user as a sensor: navigating users with visual impairments in indoor spaces using tactile landmarks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 425–432. <https://doi.org/10.1145/2207676.2207735>
- [17] Alexander Fiannaca, Ilias Apostopoulos, and Eelke Folmer. 2014. Headlock: a wearable navigation aid that helps blind cane users traverse large open spaces. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility*. ACM, New York, NY, USA, 19–26. <https://doi.org/10.1145/2661334.2661453>
- [18] German Flores and Roberto Manduchi. 2018. Easy return: an app for indoor backtracking assistance. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–12. <https://doi.org/10.1145/3173574.3173591>
- [19] Adriano Garcia, Edward Mattison, and Kanad Ghose. 2015. High-speed vision-based autonomous indoor navigation of a quadcopter. In *2015 international conference on unmanned aircraft systems (ICUAS)*. IEEE, Los Alamitos, CA, USA, 338–347. <https://doi.org/10.1109/ICUAS.2015.7152308>
- [20] Adriano Garcia, Sandeep S Mittal, Edward Kiewra, and Kanad Ghose. 2019. A convolutional neural network vision system approach to indoor autonomous quadrotor navigation. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, Los Alamitos, CA, USA, 1344–1352. <https://doi.org/10.1109/ICUAS.2019.8798183>
- [21] João Guerreiro, Daisuke Sato, Saki Asakawa, Huixu Dong, Kris M Kitani, and Chieko Asakawa. 2019. CaBot: Designing and Evaluating an Autonomous Navigation Robot for Blind People. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, New York, NY, USA, 68–82. <https://doi.org/10.1145/3308561.3353771>
- [22] A Hong, O Igbaroro, Yugang Liu, Farzad Niroui, Goldie Nejat, and Beno Benhabib. 2019. Investigating human-robot teams for learning-based semi-autonomous control in urban search and rescue environments. *Journal of Intelligent & Robotic Systems* 94, 3 (2019), 669–686. <https://doi.org/10.1007/s10846-018-0899-0>
- [23] Hochul Hwang, Tim Xia, Ibrahim Keita, Ken Suzuki, Joydeep Biswas, Sunghoon I. Lee, and Donghyun Kim. 2022. System Configuration and Navigation of a Guide Dog Robot: Toward Animal Guide Dog-Level Guiding Work. <https://doi.org/10.48550/ARXIV.2210.13368>
- [24] Hiroki Ishida, Kouichi Matsutani, Miho Adachi, Shingo Kobayashi, and Ryusuke Miyamoto. 2019. Intersection Recognition Using Results of Semantic Segmentation for Visual Navigation. In *Computer Vision Systems*. Springer, New York, NY, USA, 153–163. https://doi.org/10.1007/978-3-030-34995-0_15
- [25] Watthanarak Jeamwatthanachai, M. Wald, and G. Wills. 2019. Indoor navigation by blind people: Behaviors and challenges in unfamiliar spaces and buildings. *The British Journal of Visual Impairment* 37 (2019), 140 – 153. <https://doi.org/10.1177/0264619619833723>
- [26] Robert K Katzschnmann, Brandon Araki, and Daniela Rus. 2018. Safe local navigation for visually impaired users with a time-of-flight and haptic feedback device. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26, 3 (2018), 583–593. <https://doi.org/10.1109/TNSRE.2018.2800665>
- [27] Seita Kayukawa, Keita Higuchi, João Guerreiro, Shigeo Morishima, Yoichi Sato, Kris Kitani, and Chieko Asakawa. 2019. BBeeP: A sonic collision avoidance system for blind travellers and nearby pedestrians. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–12. <https://doi.org/10.1145/3290605.3300282>
- [28] Seita Kayukawa, Tatsuya Ishihara, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. 2020. Guiding Blind Pedestrians in Public Spaces by Understanding Walking Behavior of Nearby Pedestrians. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–22. <https://doi.org/10.1145/3411825>
- [29] Seita Kayukawa, Daisuke Sato, Masayuki Murata, Tatsuya Ishihara, Akihiro Kosugi, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. 2022. How Users, Facility Managers, and Bystanders Perceive and Accept a Navigation Robot for Visually Impaired People in Public Buildings. In *Proceedings of the 31st IEEE International Conference on Robot & Human Interactive Communication (IEEE RO-MAN 2022)*. IEEE, Piscataway, NJ, USA, 8 pages.
- [30] Sulaiman Khan, Shah Nazir, and Habib Ullah Khan. 2021. Analysis of Navigation Assistants for Blind and Visually Impaired People: A Systematic Review. *IEEE Access* 9 (2021), 26712–26734. <https://doi.org/10.1109/ACCESS.2021.3052415>
- [31] Jee-Eun Kim, Masahiro Bessho, Shinsuke Kobayashi, Noboru Koshizuka, and Ken Sakamura. 2016. Navigating Visually Impaired Travelers in a Large Train Station Using Smartphone and Bluetooth Low Energy. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*. ACM, New York, NY, USA, 604–611. <https://doi.org/10.1145/2851613.2851716>
- [32] Shinji Kotani, Hideo Mori, and Noriaki Kiyohiro. 1996. Development of the robotic travel aid "HITOMI". *Robotics and Autonomous Systems* 17, 1-2 (1996), 119–128. [https://doi.org/10.1016/0921-8890\(95\)00067-4](https://doi.org/10.1016/0921-8890(95)00067-4)
- [33] Vladimir Kulyukin, Chaitanya Gharpure, Pradyna Sute, Nathan De Graw, John Nicholson, and S Pavithran. 2004. A robotic wayfinding system for the visually impaired. In *Proceedings of the National Conference on Artificial Intelligence*. AAAI Press, Palo Alto, California, USA, 864–869. <https://doi.org/10.5555/1597321.1597337>
- [34] Bineeth Kuriakose, Raju Shrestha, and Frode Eika Sandnes. 2020. Tools and Technologies for Blind and Visually Impaired Navigation Support: A Review. *IETE Technical Review* 0, 0 (2020), 1–16. <https://doi.org/10.1080/02564602.2020.1819893>
- [35] Masaki Kurabayashi, Seita Kayukawa, Hironobu Takagi, Chieko Asakawa, and Shigeo Morishima. 2021. LineChaser: A Smartphone-Based Navigation System for Blind People to Stand in Lines. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–13. <https://doi.org/10.1145/3411764.3445451>
- [36] Masaki Kurabayashi, Seita Kayukawa, Jayakorn Vongkulbhisan, Chieko Asakawa, Daisuke Sato, Hironobu Takagi, and Shigeo Morishima. 2022. Corridor-Walker: Mobile Indoor Walking Assistance for Blind People to Avoid Obstacles and Recognize Intersections. *Proceedings of the ACM on Human-Computer Interaction* 6, MHCI (2022), 1–22. <https://doi.org/10.1145/3546714>
- [37] Gerard Lacey and Shane MacNamara. 2000. Context-aware shared control of a robot mobility aid for the elderly blind. *The International Journal of Robotics Research* 19, 11 (2000), 1054–1065. <https://doi.org/10.1177/02783640022067968>
- [38] Johan Larsson, Mathias Broxvall, and Alessandro Saffiotti. 2008. Laser based intersection detection for reactive navigation in an underground mine. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Piscataway, NJ, USA, 2222–2227. <https://doi.org/10.1109/IROS.2008.4650911>

- [39] Young Hoon Lee and Gerard Medioni. 2014. Wearable RGBD indoor navigation system for the blind. In *European Conference on Computer Vision*. Springer, New York, NY, USA, 493–508. https://doi.org/10.1007/978-3-319-16199-0_35
- [40] Bing Li, J Pablo Munoz, Xuejian Rong, Jizhong Xiao, Yingli Tian, and Aries Arditi. 2016. ISANA: wearable context-aware indoor assistive navigation with obstacle avoidance for the blind. In *European Conference on Computer Vision*. Springer, New York, NY, USA, 448–462. https://doi.org/10.1007/978-3-319-48881-3_31
- [41] Chen-Lung Lu, Zi-Yan Liu, Jui-Te Huang, Ching-I Huang, Bo-Hui Wang, Yi Chen, Nien-Hsin Wu, Hsueh-Cheng Wang, Laura Giarré, and Pei-Yi Kuo. 2021. Assistive Navigation Using Deep Reinforcement Learning Guiding Robot With UWB/Voice Beacons and Semantic Feedbacks for Blind and Visually Impaired People. *Frontiers in Robotics and AI* 8 (2021), 1–23 pages. <https://doi.org/10.3389/frobt.2021.654132>
- [42] Kanah Manjari, Madhushri Verma, and Gaurav Singal. 2020. A Survey on Assistive Technology for visually impaired. *Internet of Things* 11 (2020), 100188. <https://doi.org/10.1016/j.iot.2020.100188>
- [43] Ronja Möller, Antonino Furnari, Sebastiano Battiatto, Aki Härmä, and Giovanni Maria Farinella. 2021. A survey on human-aware robot navigation. *Robotics and Autonomous Systems* 145 (2021), 103837. <https://doi.org/10.1016/j.robot.2021.103837>
- [44] Piyush Mukhija, Siddharth Tourani, and K Madhava Krishna. 2012. Outdoor intersection detection for autonomous exploration. In *2012 15th International IEEE Conference on Intelligent Transportation Systems*. IEEE, Piscataway, NJ, USA, 218–223. <https://doi.org/10.1109/ITSC.2012.6338647>
- [45] Karin Müller, Christin Engel, Claudia Loitsch, Rainer Stieffelhagen, and Gerhard Weber. 2022. Traveling More Independently: A Study on the Diverse Needs and Challenges of People with Visual or Mobility Impairments in Unfamiliar Indoor Environments. *ACM Trans. Access. Comput.* 15, 2 (2022), 1–44. <https://doi.org/10.1145/3514255>
- [46] Saeid Nahavandi, Roohallah Alizadehsani, Darius Nahavandi, Shady Mohamed, Navid Mohajer, Mohammad Rokonuzzaman, and Ibrahim Hossain. 2022. A Comprehensive Review on Autonomous Navigation. <https://doi.org/10.48550/ARXIV.2212.12808>
- [47] Anh-Tu Nguyen, Jagat Jyoti Rath, Chen Lv, Thierry-Marie Guerra, and Jimmy Lauber. 2021. Human-machine shared driving control for semi-autonomous vehicles using level of cooperativeness. *Sensors* 21, 14 (2021), 4647. <https://doi.org/10.3390/s2114467>
- [48] Suraj R. Pardeshi, Vikul J. Pawar, Kailas D. Kharat, and Sachin Chavan. 2021. Assistive Technologies for Visually Impaired Persons Using Image Processing Techniques – A Survey. In *Recent Trends in Image Processing and Pattern Recognition*, K. C. Santosh and Bharti Gawali (Eds.). Springer Singapore, Singapore, 95–110. https://doi.org/10.1007/978-981-16-0507-9_9
- [49] Xavier Perrin, Ricardo Chavarriaga, Francis Colas, Roland Siegwart, and José del R Millán. 2010. Brain-coupled interaction for semi-autonomous navigation of an assistive robot. *Robotics and Autonomous Systems* 58, 12 (2010), 1246–1255. <https://doi.org/10.1016/j.robot.2010.05.010>
- [50] Giorgio Presti, Dragan Ahmetovic, Mattia Ducci, Cristian Bernareggi, Luca Ludovico, Adriano Baratè, Federico Avanzini, and Sergio Mascetti. 2019. WatchOut: Obstacle sonification for people with visual impairment or blindness. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, New York, NY, USA, 402–413. <https://doi.org/10.1145/3308561.3353779>
- [51] Pablo-Alejandro Quinones, Tammy Greene, Rayoung Yang, and Mark Newman. 2011. Supporting visually impaired navigation: a needs-finding study. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1645–1650. <https://doi.org/10.1145/1979742.1979822>
- [52] Santhosh K Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. 2020. Occupancy anticipation for efficient exploration and navigation. In *Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, New York, NY, USA, 400–418. https://doi.org/10.1007/978-3-030-58558-7_24
- [53] David A Ross and Alexander Lightman. 2005. Talking braille: a wireless ubiquitous computing network for orientation and wayfinding. In *Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility*. ACM, New York, NY, USA, 98–105. <https://doi.org/10.1145/1090785.1090805>
- [54] Manaswi Saha, Alexander J Fianinaca, Melanie Kneisel, Edward Cutrell, and Meredith Ringel Morris. 2019. Closing the gap: Designing for the last-few-meters wayfinding problem for people with visual impairments. In *The 21st international acm sigaccess conference on computers and accessibility*. ACM, New York, NY, USA, 222–235. <https://doi.org/10.1145/3308561.3353776>
- [55] Nuzhat Gooda Sahib, Tony Stockman, Anastasios Tombros, and Ouussama Metathla. 2013. Participatory design with blind users: a scenario-based approach. In *IFIP Conference on Human-Computer Interaction*. Springer, New York, NY, USA, 685–701. https://doi.org/10.1007/978-3-642-40483-2_48
- [56] Daisuke Sato, Urash Oh, João Guerreiro, Dragan Ahmetovic, Kakuya Naito, Hironobu Takagi, Kris M Kitani, and Chiaki Asakawa. 2019. NavCog3 in the wild: Large-scale blind indoor navigation assistant with semantic features. *ACM Transactions on Accessible Computing (TACCESS)* 12, 3 (2019), 14. <https://doi.org/10.1145/3340319>
- [57] Hiroaki Seki, S Kobayashi, Yoshitsugu Kamiya, Masatoshi Hikizu, and Hisanao Nomura. 2000. Autonomous/semi-autonomous navigation system of a wheelchair by active ultrasonic beacons. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, Vol. 2. IEEE, Piscataway, NJ, USA, 1366–1371. <https://doi.org/10.1109/ROBOT.2000.844788>
- [58] Mahendran Subramanian, Noyan Songur, Darrell Adjei, Pavel Orlov, and A Aldo Faisal. 2019. A Eye Drive: Gaze-based semi-autonomous wheelchair interface. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, Piscataway, NJ, USA, 5967–5970. <https://doi.org/10.1109/EMBC.2019.8856608>
- [59] Haobin Tan, Chang Chen, Xinyu Luo, Jiaming Zhang, Constantin Seibold, Kailun Yang, and Rainier Stieffelhagen. 2021. Flying guide dog: Walkable path discovery for the visually impaired utilizing drones and transformer-based semantic segmentation. In *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, Piscataway, NJ, USA, 1123–1128. <https://doi.org/10.1109/ROBIO54168.2021.9739520>
- [60] Hongru Tang, Xiaosong Cao, Aiguo Song, Yan Guo, and Jiatong Bao. 2009. Human-robot collaborative teleoperation system for semi-autonomous reconnaissance robot. In *2009 International Conference on Mechatronics and Automation*. IEEE, Piscataway, NJ, USA, 1934–1939. <https://doi.org/10.1109/ICMA.2009.5246589>
- [61] N Veeranjaneyulu. 2019. A Meta-Analysis on Obstacle Detection for Visually Impaired People. *i-manager's Journal on Pattern Recognition* 6, 1 (2019), 40–62. <https://doi.org/10.26634/jpr.6.1.15523>
- [62] Liyang Wang, Jinxin Zhao, and Liangjun Zhang. 2021. Navdog: robotic navigation guide dog via model predictive control and human-robot modeling. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*. New York, NY, USA, ACM, 815–818. <https://doi.org/10.1145/3412841.3442098>
- [63] Yue Wang and Fumin Zhang. 2017. *Trends in control and decision-making for human-robot collaboration systems*. Springer, New York, NY, USA. <https://doi.org/10.1007/978-3-319-40533-9>
- [64] Anxing Xiao, Wenzhe Tong, Lizi Yang, Jun Zeng, Zhongyu Li, and Koushil Sreenath. 2021. Robotic guide dog: Leading a human with leash-guided hybrid physical interaction. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, Piscataway, NJ, USA, 11470–11476. <https://doi.org/10.1109/ICRA506.2021.9561786>
- [65] Yutaro Yamanaka, Seita Kayukawa, Hironobu Takagi, Yuichi Nagaoka, Yoshimune Hiratsuka, and Satoshi Kurihara. 2021. One-Shot Wayfinding Method for Blind People via OCR and Arrow Analysis with a 360-degree Smartphone Camera. In *Proceedings of the 18th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (Beppu, Japan) (MobiQuitous '21)*. Springer, New York, NY, USA, 19 pages. https://doi.org/10.1007/978-3-030-94822-1_9
- [66] Fan Yang, Dung-Han Lee, John Keller, and Sebastian Scherer. 2021. Graph-based topological exploration planning in large-scale 3d environments. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, Piscataway, NJ, USA, 12730–12736. <https://doi.org/10.1109/ICRA48506.2021.9561830>
- [67] Chris Yoon, Ryan Louie, Jeremy Ryan, MinhKhang Vu, Hyegi Bang, William Derkens, and Paul Ruvolo. 2019. Leveraging Augmented Reality to Create Apps for People with Visual Disabilities: A Case Study in Indoor Navigation. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, New York, NY, USA, 210–221. <https://doi.org/10.1145/3308561.3353788>
- [68] Limin Zeng, Björn Einert, Alexander Pitkin, and Gerhard Weber. 2018. Hapticine: Design and development of an interactive haptic rein for a guidance robot. In *International Conference on Computers Helping People with Special Needs*. Springer, New York, NY, USA, 94–101. https://doi.org/10.1007/978-3-319-94274-2_14
- [69] Kai Zhu and Tao Zhang. 2021. Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Science and Technology* 26, 5 (2021), 674–691. <https://doi.org/10.26599/TST.2021.9010012>
- [70] Quanwen Zhu, Long Chen, Qingquan Li, Ming Li, Andreas Nüchter, and Jian Wang. 2012. 3d lidar point cloud based intersection recognition for autonomous driving. In *2012 IEEE Intelligent Vehicles Symposium*. IEEE, Piscataway, NJ, USA, 456–461. <https://doi.org/10.1109/IVS.2012.6232219>
- [71] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. 2017. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, Piscataway, NJ, USA, 3357–3364. <https://doi.org/10.1109/ICRA.2017.7989381>