University of Nevada, Reno

Integrating Minimalistic Localization and Navigation for People with Visual Impairments

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science
with a major in Computer Science.

by

Ilias Apostolopoulos

Dr. Kostas E. Bekris, Thesis Advisor

May 2011

University of Nevada, Reno

Statewide • Worldwide

We recommend that the thesis prepared under our supervision by

Ilias Apostolopoulos

entitled

Integrating Minimalistic Localization and Navigation for People with Visual Impairments

be accepted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Dr. Kostas E. Bekris, Ph.D., Advisor

Dr. Eelke Folmer, Ph.D., Committee Member

Dr. Dwight Egbert, Ph.D., Committee Member

Dr. Daniel Cook, Ph.D., Graduate School Representative

Marsha H. Read, Ph.D., Associate Dean, Graduate School

May 2011

Abstract

Indoor localization and navigation systems for individuals with visual impairments (VI) typically rely upon extensive augmentation of the physical space or expensive sensors; thus, few systems have been adopted. This work describes a system able to guide people with VI through buildings using inexpensive sensors, such as accelerometers, which are available in portable devices like smart phones. This approach introduces some challenges due to the limited computational power of the portable devices and the highly erroneous sensors. The method takes advantage of feedback from the human user, who confirms the presence of landmarks. The system calculates the location of the user in real time and uses it to provide audio instructions on how to reach the desired destination. A first set of experiments suggested that the accuracy of the localization depends on the type of directions provided and the availability of good transition and observation models that describe the user's behavior. During this initial set of experiments, the system was not executed in real time so the approach had to be improved. Towards an improved version of the method, a significant amount of computation was transferred offline in order to speed up the system's online execution. Inspired by results in multi-model estimation, this work employs multiple particle filters, where each one uses a different assumption for the user's average step length. This helps to adaptively estimate the value of this parameter on the fly. The system simultaneously estimates the step length of the user, as it varies between different people, from path to path, and during the execution of the path. Experiments are presented that evaluate the accuracy of the location estimation process and of the integrated direction provision method. Sighted people, that were blindfolded, participated in these experiments.

Acknowledgements

This work was supported by internal funds by UNR.

Contents

Abstract **i**

Acknowledgements **ii**

List of Figures **iv**

List of Tables **v**

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Objective . . . 2
1.3 Challenges . . . 2
1.4 System Overview . . . 3

2 Background **6**
2.1 Navigation and Cognitive Mapping . . . 6
2.2 Indoor Navigation Systems for Users with VI . . . 7
2.2.1 Localization . . . 8
2.2.2 Path planning . . . 8
2.2.3 Providing location information . . . 8
2.2.4 Interaction . . . 9
2.3 Localization Techniques . . . 9
2.3.1 Dead-Reckoning . . . 9
2.3.2 Beacon-based . . . 9
2.3.3 Sensor-based . . . 9
2.4 Bayesian methods . . . 10
2.5 Path planning . . . 11
2.6 Multi-model representation . . . 11

3 Initial Approach **13**
3.1 Objectives . . . 13
3.2 High-level operation . . . 13
3.2.1 Direction Provision . . . 14
3.3 Localization . . . 16
3.3.1 Particle Filter . . . 18

3.3.2 Transition Model . . . 18
3.3.3 Observation Model . . . 19
3.3.4 Sampling . . . 19
3.4 Experiments . . . 20
3.4.1 Setup . . . 20
3.4.2 Participants . . . 21
3.4.3 Ground Truth . . . 21
3.4.4 Parameters . . . 22
3.4.5 Success Ratio of Direction Provision . . . 22
3.4.6 Localization Accuracy . . . 23

4 Improved Approach **26**
4.1 Overview . . . 26
4.2 Offline Process . . . 26
4.3 Direction Provision . . . 27
4.4 Localization . . . 28
4.4.1 Transition Model . . . 28
4.4.2 Observation Model . . . 29
4.4.3 Sampling . . . 29
4.5 Experiments . . . 30
4.5.1 Setup . . . 30
4.5.2 Participants . . . 30
4.5.3 Ground Truth . . . 32
4.5.4 Parameters . . . 32
4.5.5 Success Ratio of Direction Provision . . . 33
4.5.6 Off course correction . . . 34
4.5.7 Computational overhead . . . 35

5 Conclusion **37**
5.1 Summary . . . 37
5.2 Future work . . . 38

List of Figures

1.1 An individual with visual impairments testing the system. 2
1.2 A communication graph between the components of the system. . . . 5

3.1 The map of the environment and the paths traversed during the experimental section. 15
3.2 An example path starting from 255 and finishing at 231 16
3.3 An illustration of the particle reseting process. The particle's position is moved close to the nearest landmark of the type that the user confirmed 20
3.4 a) Ground truth vs. dead-reckoning vs. particle filtering. b) Error graph. . 24

4.1 The map of the first floor 31
4.2 The map of the second floor 31
4.3 A graph showing the localization error during the execution of a path. The different lines represent the distance of the different particle filters from the actual location of the user 35

List of Tables

1.1 Examples of model parameters 3

2.1 Indoor Navigation Systems for Users with VI 7

3.1 Table of parameters used in the case studies 22
3.2 Average distance between destination and the user's position upon completion (m) 23
3.3 Average path duration (sec). 23
3.4 Average error of dead reckoning in final location (m). 24
3.5 Average error of the proposed interactive localization process (m). . . 25
3.6 Standard deviation for the error of the proposed interactive localization process (m). 25

4.1 Distance between destination and the user's position upon completion (m) 33
4.2 Distance between ground truth and destination (m) 34
4.3 Profiling data (msec) 36

Chapter 1 Introduction

The motivation of this work is presented here, along with the objectives, the challenges faced and a high-level system overview. The following chapters include some background work, a description of an initial approach to the problem of guiding people with visual impairments, an improved approach based on the challenges faced during the initial implementation and, finally, a discussion section about the overall results and future work.

1.1 Motivation

Sighted people can navigate environments by primarily relying upon their visual senses to find their way. Individuals with visual impairments (VI) have to rely upon their compensatory senses (e.g. touch, sound) for way-finding, resulting in reduced mobility. Unfamiliar environments are especially challenging as it is difficult to build a mental map of the surroundings using non-visual feedback [45].

To increase the mobility of individuals with VI various navigation systems have been developed. While there are many solutions for outdoor navigation systems, indoor alternatives are more difficult to develop. Outdoor navigation systems typically use GPS, however GPS signals cannot be received in indoor environments. Existing indoor navigation systems typically rely upon augmenting physical infrastructure with identifiers such as RFID tags [41, 10, 5]. While RFID tags might be cheap, a large amount of them is required to cover a whole building. Often, RFID tags are installed under carpets on the floor. Although this is possible, hallways or large open spaces with concrete floor or tiles render the installation of these tags more difficult. Other solutions employ laser sensors [30, 29] or cameras [81]. While these solutions often lead to sophisticated algorithms, they can be expensive, cumbersome, and computationally demanding alternatives.

Figure 1.1: An individual with visual impairments testing the system.

1.2 Objective

This research describes an inexpensive solution that does not require physical infrastructure and depends on cheap, light-weight sensors, such as an accelerometer and a compass, that are already available on popular devices, such as smart phones. Instead of depending on a physical infrastructure or expensive and cumbersome sensors, the system presented here only needs a virtual infrastructure, that can be created and updated very fast with low cost, and lightweight inexpensive sensors that can found in an everyday handheld device.

1.3 Challenges

The proposed system has to deal with uncertainty at multiple levels of its operation. Uncertainty arises from:

- The behavior of the user: e.g., how quickly does the person move, how accurately

does one person turn when instructed to do so, how good is the person at identifying landmarks. For instance, while users can easily identify landmarks of different types, they cannot readily distinguish landmarks of the same type. When a user confirms a door, and there are a number of doors close one to each other, it is possible that the user did not confirm the correct one. The system has to take into consideration the possibility that the user confirmed an adjacent landmark.

- The environment: The model of the environment may lead to an uncertain representation, as the annotation of the map, such as the actual location or the type of the landmarks, might be incorrect.

- The sensors: Sensors used in mobile phones usually have low accuracy. The error due to these sensors must also be taken into consideration.

The core of the research effort regarding the localization component is devoted to the definition and online learning of appropriate observation and transition models for individual users. Table 1.1 provides examples of potential parameters for these models. It is important for the models to be able to differentiate between users. This is especially important for this application, as different users will also have different types and degrees of visual impairments.

Transition Model	**Observation Model**
Average Step Length	Landmark Identification Accuracy
Step Detection Accuracy	Distance from Landmark upon Confirmation
Turning Accuracy	Confirmation Efficiency

Table 1.1: Examples of model parameters

1.4 System Overview

Tactile landmarks, such as doors, hallway intersections or floor transitions, play an important role in the cognitive mapping of indoor spaces by users with VI [36, 9].

By incorporating the unique sensing capabilities of users with VI, the system aims to provide guidance in spaces for which the user does not have a prior cognitive map. The system assumes the availability of a 2D map with addressing information (room numbers) and landmark locations. Then, it follows these steps:

1. A user specifies a start and a destination room number to travel to.

2. Given landmarks identifiable by users with VI on the map, the system computes the shortest path using A* and identifies landmarks along the path.

3. The user presses a button on the phone after successfully executing each direction.

4. Directions are provided iteratively upon the confirmation of each landmark, or when the user is presumed to be lost. The phone's built-in speaker is used for the transmission of the direction.

Figure 1.2 lists a high-level overview of the four different components of the system: (1) the cyber-representation component stores annotated models of indoor environments; (2) the localization component provides a location estimate of the user with VI; (3) the direction provision component provides directions to a user specified location; and (4) the interface component interacts with the user. All components have physical models of the users with VI with the exception of the cyber-representation component which explicitly models sighted users annotating the models.

The landmarks used from the system are features that can be found in most buildings. Doors, hallway intersections, floor transitions, water coolers, ramps, stairs and elevators are incorporated to guide the user around the building. These landmarks are easily recognizable from users with VI by using touch and sound, thus creating no need for additional physical infrastructure.

This research proposes a system that takes into consideration the sources of uncertainty previously mentioned and provides an integration of localization and path planning primitives. Multiple particle filters are employed in order to deal with the

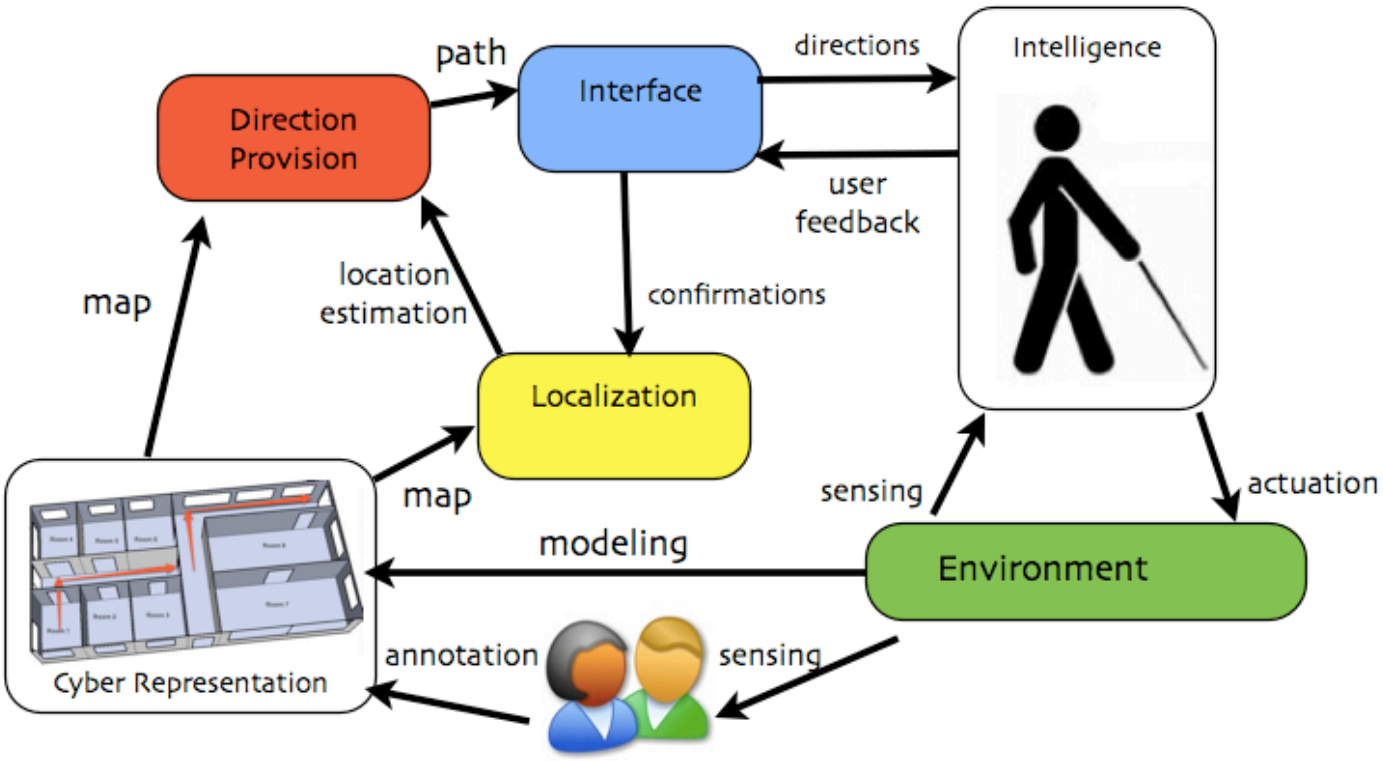

Figure 1.2: A communication graph between the components of the system.

highly non-linear process of localization as well as learning and updating a model of the user's behavior. To provide directions to the user, a path is created from start to goal using the A* algorithm. Then, turn-to-turn directions are provided based on the location estimation of the user provided by localization. Results show that this system is capable of successfully locating and guiding the user to the desired destination when a path with unique landmarks is provided.

Chapter 2 Background

There is a lot of work related to navigation systems for people with visual impairments, localization techniques, path planning, bayesian methods and multi-model estimation.

2.1 Navigation and Cognitive Mapping

Navigation relies on a combination of mobility and orientation skills [21]. People employ either path integration, where they orient themselves relative to a starting position using proprioceptive date, or landmark-based navigation, where they rely upon perceptual cues together with an external or cognitive map [53, 22, 51]. Path integration allows for exploring unfamiliar environments in which users may build a cognitive map by observing landmarks [82, 53]. Studies show small difference in path integration ability between sighted and individuals with VI [51], but cognitive mapping is significantly slower for users with VI [46, 71]. Cognitive mapping of outdoor environments has been extensively studied [21, 68, 28] and has reported to primarily rely upon landmarks that can be recognized by touch [71] in the users' immediate space [11], such as curbs or traffic lights. Pedestrian navigation systems for sighted users have been developed where users are localized by reporting visual landmarks, such as escalators [56] or churches [32]. Recently, the cognitive mapping of indoor spaces by people with VI has been studied [36, 77, 26]. Tactile landmarks easily sensed through touch, such as doors, hallway intersections and floor transitions, play an important role in the cognitive mapping of indoor spaces [36, 83]. Sounds and smells also play a minor role [83]. The use of virtual environments has been shown to aid cognitive mapping of users with VI [46, 71, 57].

Table 2.1: Indoor Navigation Systems for Users with VI

Authors	**Localization**	**Directions**	**Information**	**Feedback**
1998 Sonnenblick [76]	IR	-	Room name	Speech
2001 May [54]	barcode	-	-	Braille
2002 Ross & Blasch [68]	IR, RFID	-	-	Audio, Speech, Haptic
2003 Coroama and Rothenbacher [17]	RF	Objects	Central	-
2004 Ran et al [62]	Ultrasound	Objects	Room layout, objects	Speech
Hub et al [33]	Wifi, Camera	-	Object name	Speech
Amemiya et al [5]	RFID	-	-	Haptic (braille)
2005 Ross & Lightman [69]	IR, RF, Audio	Locations	Objects	Speech, braille
Willis & Helal [85]	RFID	Objects	Room layout, objects.	Haptic (braille)
2006 Gifford et al [25]	RFID	-	Room layout, objects	Speech
2008 Bessho et al [10]	RFID, IR	Stations	Layout of Station	Speech
Riehle et al [64]	WiFi	Rooms	-	Speech
Rajamaki et al [61]	WiFi	Rooms	-	Speech
D'Atri et al [18]	RFID	-	-	Speech

2.2 Indoor Navigation Systems for Users with VI

Navigation systems for users with VI aim to allow safe navigation in unfamiliar environments. This includes locating the user and optionally providing directions to a desired destination and/or describing the surroundings, such as obstacles or landmarks. Navigation systems can be differentiated into outdoor and indoor systems. Outdoor systems [52, 69, 68] typically use GPS for localization. Relatively few indoor navigation systems exist, as GPS signals cannot be received indoors, and alternative localization techniques must be used. Table 2.1 provides an overview of existing indoor navigation systems for VI and lists the specific techniques used for localization, path planning, providing location information and interaction with the system.

2.2.1 Localization

By augmenting the physical infrastructure with identifiers, people can be localized when an identifier is sensed. Different technologies have been used, such as infrared (IR) [76, 68, 69], wireless [34, 33, 64, 61], ultrasound [62], or radio frequency identifier (RFID) tags [5, 68, 25, 10, 18]. There are limitations in this approach, as IR requires line of sight and the environment or the user may interfere with RFID readings [68]. Wireless based systems often suffer from multi-path effects [85] or cannot be used in certain spaces, e.g., hospitals. A number of vision-based systems require little physical augmentation [33, 58, 35]. Some of them may utilize wireless signals or RFID tags, or a virtual environment representation. There are also approaches that utilize magnetic compasses [78]. Magnetic compasses are not very reliable when used in an indoors environment due to the noise created by the infrastructure. Although, there is a work that prerecords the signatures of different disturbances in a building and tries to match these signatures later in order to get a localization estimation. A recent approach uses a laser-range finder combined with odometry readings [31]. Relative to the last methods, the proposed approach aims to further reduce sensing requirements and avoid any environment augmentation with identifiers.

2.2.2 Path planning

Only three systems [62, 69, 10] provide global path planning where paths are computed using A* [70] and where the system updates the user's position dynamically.

2.2.3 Providing location information

Information on a user's location varies from providing the name of a room [76] to detailed descriptions of the room's layout, such as objects in that room [25, 62]. Information is either stored locally [76, 25], centrally [62, 5, 33, 64] or distributedly [85, 10].

2.2.4 Interaction

Users receive feedback using audio such as speech [76, 25, 10] or audio cues [68] to haptic solutions such as a tapping interface [68], pager belts [85] or haptic gloves [5].

2.3 Localization Techniques

Certain navigation devices focus on local hazard detection to provide obstacle-avoidance capabilities to users with VI [72, 86]. Most navigation systems, however, are able to locate the user and provide directions to a user-specified destination. Outdoor navigation systems [52, 67] mainly use GPS to localize the user. Indoor systems cannot use GPS signals, as these are blocked by buildings. To surpass this issue, alternative localization techniques have been developed.

2.3.1 Dead-Reckoning

Dead-Reckoning techniques integrate measurements of the human's motion. Accelerometers [14] and radar measurements [84] have been used for this purpose. Without any external reference, however, the error in dead-reckoning grows unbounded.

2.3.2 Beacon-based

Beacon-based approaches augment the physical space with identifiers. Such beacons could be retro-reflective digital signs detected by a camera [81], infrared [67] or ultrasound identifiers [62]. A popular solution involves RFID tags [41, 10, 5]. Nevertheless, locating identifiers may be hard, as beacons may require line of sight or close proximity to the human. Other beacons, such as wireless nodes [64, 44, 43], suffer from multi-path effects or interference. Another drawback is the significant time and cost spent installing and calibrating beacons.

2.3.3 Sensor-based

Sensor-based solutions employ sensors, such as cameras [40], that can detect pre-existing features of indoor spaces, such as walls or doors. For instance, a multi-camera

rig has been developed to estimate the 6 DOF pose of people with VI [20]. A different camera system matches physical objects with objects in a virtual representation of the space [33]. Nevertheless, cameras require good lighting conditions, and may impose a computational cost prohibitive for portable devices. An alternative makes use of a 2D laser scanner [30, 29]. This method achieves 3D pose estimation by integrating data from an IMU unit, the laser scanner, and knowledge of the 3D structure of the space. While laser scanners can robustly detect low-level features, this method has led to sophisticated algorithms for 3D pose estimation and it depends on relatively expensive and heavy sensors.

The proposed approach is also a sensor-based solution. It employs the user as a sensor together with information from light-weight, affordable devices, such as a pedometer. These sensors are available on smart phones and it is interesting to study the feasibility of using such popular devices to (i) interact effectively with a user with VI; and (ii) run in real-time localization primitives given their limited resources. To achieve this objective under the minimalistic and noisy nature of the available sensors, this work utilizes probabilistic tools that have been shown to be effective in robotics and evaluates their efficiency for different forms of direction provision.

2.4 Bayesian methods

Bayesian methods for localization work incrementally, where given the previous belief about the agent's location, the new belief is computed using the latest displacement and sensor reading. A transition model is used in order to advance the movement of the system and an observation model in order to compare the sensor readings with the state estimation. An important issue is how to represent and store the belief distribution. One method is the Extended Kalman filter (EKF) [37, 75], which assumes normal distributions. Its purpose is to use measurements observed over time, containing noise (random variations) and other inaccuracies, and produce values that tend to be closer to the true values of the measurements and their associated calculated values. While Kalman filters provide a compact representation and return

the optimum estimate under certain assumptions, a normal distribution may not be a good model, especially for multi-modal distributions. An alternative is to use particle filters [27, 80, 47, 63, 60, 79], which sample estimates of the agent's state. Particle filters keep a number of different estimations called particles. Each particle holds a state estimation and a weight. Particle filters are able to represent multi-modal distributions at the expense of increased computation. Such distributions arise often in this research's application, such as when a door is confirmed, where the belief increases in front of all of the doors in the vicinity of the last estimate. Thus, particle filters appear as an appropriate solution. This research shows that it is possible to achieve a sufficient, real-time solution with a particle filter.

2.5 Path planning

Popular path planning techniques include search methods, such as A* on grid-based maps [59, 66, 70], the visibility graph [24, 19], the Voronoi graph or the medial axis [48, 8], cell decomposition techniques [16, 3] or potential field approaches [39, 65]. Recent work has focused on solving complex high-dimensional challenges, giving rise to sampling-based methods [38, 4, 12, 73, 23]. These methods sample collision-free configurations in order to construct a roadmap, a graph representing the connectivity of the obstacle-free space. While these methods are not complete, the probability the roadmap reflects the obstacle-free space connectivity increases exponentially fast to 1 [42]. Planning under uncertainty can be modeled by Partially Observable Markov Decision Process (POMDP) [50, 13].

2.6 Multi-model representation

The user's step length changes dynamically and these changes have to be taken into consideration during path execution. To achieve this, this project is building upon work in multi-model state estimation [1, 55, 49, 74]. Multi-model estimation is commonly used to calculate both the state and the model of the system, when it is changing [15]. Multi-model estimation systems though are also used to track the

changes in the environment itself [2].

In the first case, multi-model estimation is commonly used to both track the location of the system and the value of a system variable that might be discrete [15] or continuous [49]. In the second case, the system tries to determine the noise due to the environment. By doing so, the system can dynamically adapt in an environment with random properties [2]. Most multi-model estimation systems use multiple Kalman filters to determine the system's model.

Multi-model estimations utilize multiple bayesian methods in order to estimate a model variable. The bayesian methods can be either multiple Extended Kalman filters or multiple particle filters. Each method holds a unique value of a variable that needs to be approximated along with localization. After a few transitions and observations, the estimations of the variable converge to a value if the system model is stable. In case the system model changes, the variable estimations adopt dynamically to the correct values in order to match with the current observations of the system.

The proposed system is trying to determine the system model by estimating the user's step length. To achieve multi-model estimations, the system is maintaining multiple particle filters, where each one has a different estimation for the step length of the user.

Chapter 3 Initial Approach

An initial approach was implemented to address the problem of indoor navigation. Experiments were executed to test this approach localization accuracy and the effect of different types of directions to the success rate and the execution time.

3.1 Objectives

The goal of this initial approach was to test if it possible to successfully localize the user and decide which type of directions is better based on accuracy and speed. This initial approach is not executed on the phone. Sensing data from the users are gathered during the execution of the experiments and are then processed offline to determine the location of the user.

3.2 High-level operation

Tactile landmarks, such as doors, intersections or floor transitions, play an important role in the cognitive mapping of indoor spaces by users with VI [36, 9]. By incorporating the unique sensing capabilities of users with VI, the system aims to provide guidance in spaces for which the user does not have a prior cognitive map. The system assumes the availability of a 2D map with addressing information (room numbers) and landmark locations. Then, it follows these steps:

1. A user specifies a start and destination room number to travel to.

2. The system computes the shortest path using A* and finds landmarks along the path.

3. Directions are provided iteratively upon completion through the phone's built-in speaker. The user presses a button on the phone after successfully executing each direction.

3.2.1 Direction Provision

The type of directions significantly effects the efficiency and reliability of navigation. Reliability is high when the user is required to confirm the presence of every single landmark along a path but this is detrimental to efficiency. Conversely, when the system solely relies on odometry, users have a smaller cognitive load but a high chance of getting lost, due to the inherent propagation of errors associated with dead reckoning. To gain a better insight in these tradeoffs two different types of direction provisions were tested:

1. **Landmark** based directions, e.g., "move forward until you reach a hallway on your left ". No distance to a landmark is provided. Directions were subdivided based on the maximum distance between landmarks: (a) 30ft, (b) 50ft and (c) unlimited. Wall following and door counting strategies were employed for the first 2 cases (i.e., "Follow the wall on your left until you reach the third door"). For the last case no wall following or door counting strategies were used for directions leading to a hallway.

2. **Metric** based directions, e.g., "Walk x steps until your reach a landmark on your left/right". Within this approach the maximum distance between landmarks was also varied with 30ft, 50ft and unlimited. For example: "Walk 23 steps until you reach a door on your right" for the 30ft limit.

Both types of instructions contain a second type of direction with an action on a landmark, for example, "Turn right into the hallway".

The directions provided to the user were hardcoded into the system for each path. This initial approach did not generate automatically the instructions. Instead, the paths were predefined and the ability of the users to follow these instructions was tested. Here are some examples of instructions of different types generated to guide the user along the path in Figure 3.4.

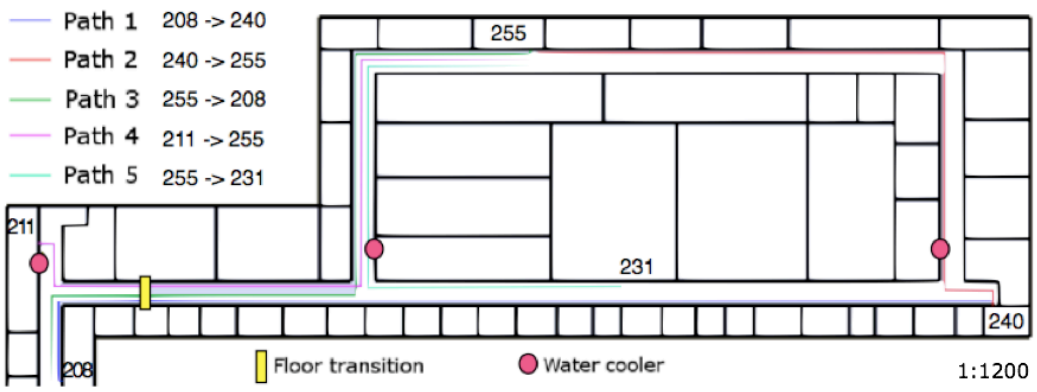

Figure 3.1: The map of the environment and the paths traversed during the experimental section.

Landmark No Max Threshold

- Exit the room then turn right
- Move forward until you reach a hallway on your left
- Turn left to the hallway
- Move forward until you reach a water cooler on your left
- Turn left to the hallway
- Follow the wall on your left until you reach the third door
- You have reached your destination

Metric 15 Meters Threshold

- Exit the room then turn right
- Walk x steps until you reach a hallway on your left
- Turn left to the hallway
- Walk x steps until you reach a door on your left
- Walk x steps until you reach a water cooler on your left

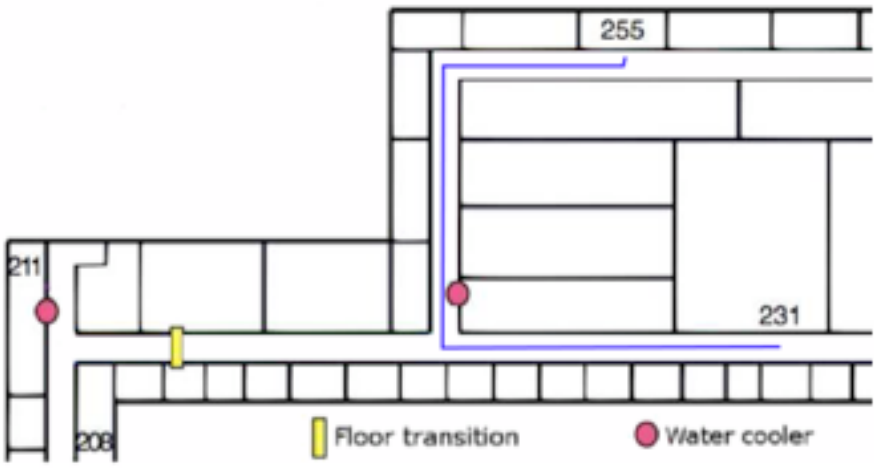

Figure 3.2: An example path starting from 255 and finishing at 231

- Turn left to the hallway
- Walk x steps until you reach a door on your right
- Walk x steps until you reach a door on your left
- You have reached your destination

3.3 Localization

The data we get from the sensors are raw data from the accelerometer which are then filtered to detect a step. We are also receiving sensory data from the user when a landmark is confirmed. The system expects a specific kind of landmark to be confirmed, depending on the last instruction, and when it gets confirmed it searches for this kind of landmark on the data from the mini-map.

Consider a planar system moving among n static landmarks. The system is a human with VI, and the landmarks corresponds to tactile features of indoor spaces. Let $\xi = (x, y, \theta)$ denote the state of the system. The map m of the world is available and stores a function which returns whether each(x, y) is occupied by an obstacle or not. The map also stores the n landmarks present in the world. The landmarks belong to k different types $\{L_1, \ldots, L_k\}$, such as doors, hallway intersections or floor transitions

(most often $k < n$). Landmarks l^i in the same class L_j are indistinguishable to the human user.

The data $d_T = (o(0{:}T), u(0{:}T-1))$ available to the system up to time T are transitions $u(0{:}T-1)$ and observations $o(0{:}T)$. A transition $u_t = (u_t^f, u_t^\theta)$ at time t corresponds to a motion where the agent acquires the global orientation u_t^θ and moves forward u_t^f. This transition determines the kinematic transition model of the system:

$$(x_{t+1}, y_{t+1}, \theta_{t+1}) = (x_t + u_t^f \cdot cos(u_t^\theta), y_t + u_t^f \cdot sin(u_t^\theta), u_t^\theta) \tag{3.1}$$

In this application the translation is measured from a pedometer and the orientation with a compass. An observation o_t^j of a landmark type L_j from state $\xi_t = (x_t, y_t, \theta_t)$ implies:

$$\exists\ l^i \in L_j : ||(x_t, y_t), (x^i, y^i)|| < R_{obs} \tag{3.2}$$

The above observation model specifies that a user can sense a landmark type L_j in their vicinity, only if such a landmark $l^i(x^i, y^i) \in L_j$ is within a predefined observation distance R_{obs} from the current coordinates of the system (x_t, y_t).

The objective is to be able to incrementally estimate the user's state ξ_T at time T. The general Bayes filter computes a belief distribution $B_T = P(\xi_T | d_T)$ at time T over ξ_T given the data d_T. The computation requires:

a) an initialization B_0,

b) a transition model $P(\xi' | u, \xi)$, describing the probability that the user is at location ξ' if it was previously at ξ and transitioned by u, and

c) the observation model $P(o|\xi, m)$ describing the likelihood of observing o when the user is at ξ and given the map m. The map is assumed to be static and correct in this work.

Then given a normalization factor η the belief distribution can be updated as follows:

$$B_T = \eta \cdot P(o_T | \xi_T, m) \int P(\xi_T | u_{T-1}, \xi_{T-1}) \cdot B_{T-1} \cdot d\xi_{T-1} \tag{3.3}$$

The computational cost of integrating over all states renders the explicit computation of the above equation inefficient. Most online algorithms simplify the problem by approximating Eq. 1. This work follows a Particle Filter approximation.

3.3.1 Particle Filter

It is possible to represent B_T through a set of P particles $p^i = (\xi^i, w^i)$ $(i \in [1, N])$. Each particle stores a state estimate ξ^i together with a weight w^i, representing the probability of ξ^i being the true state. As the number of particles approaches infinity, the better the particle filter represents the belief distribution. In order to update the particle filter given a new transition and an observation, this work follows an approach similar to importance sampling [27]. At each time step T, given a particle population $\{p_T^1, \ldots, p_T^P\}$, a transition u_T and an observation o_{T+1}, the following steps are executed:

A. For each particle $p_T^i = (\xi_T^i, w_T^i)$

   i. Employ the transition model $P(\xi_{T+1}^i | u_T, \xi_T^i)$ to acquire: ξ_{T+1}^i.

   ii. Employ the observation model to compute the new weight $w_{T+1}^i = P(o_{T+1} | \xi_{T+1}, m)$.

B. Sample a new population of P particles given the weights w_{T+1}^i

3.3.2 Transition Model

The approach collects all the sensor readings that have been produced by the sensors during the last time step: (i) orientations from the compass and (ii) step counts from the pedometer. Typically within a single time step (in the order of 150ms-300ms), the compass provides multiple orientation estimates. These are averaged to acquire u_t^θ. The pedometer typically returns either zero or one step measured. This value has to be translated into a distance estimate. To compute the length of a step, the implementation employs a short training session for each user. During this session the user traverses a couple of paths between two landmarks with known distance. The pedometer computes the number of steps during the execution of these paths

and the device estimates the average length of a step. Based on this estimate and the number of steps measured by the pedometer online, the approach constructs u_t^f.

Given u_t^f and u_t^θ, different levels of noise are added for the application of the transition model to each particle. The noise parameters for particle p^i are drawn from a normal distribution: (i) $(u_t^f)^i = \mathcal{N}(u_t^f, \sigma_f^2)$ and (ii) $(u_t^\theta)^i = \mathcal{N}(u_t^\theta, \sigma_\theta^2)$. The resulting values are used in Eq. 3.1 to acquire the new state ξ_{T+1}^i. The corresponding transition from ξ_T^i to ξ_{T+1}^i is then checked on the map to compute whether it corresponds to a path that collides with obstacles. If it does, then the sampling of $(u_t^f)^i$ and $(u_t^\theta)^i$ is repeated until either a collision free transition is found or a certain number of attempts has been tested.

3.3.3 Observation Model

There are two cases for computing the weights w_{T+1}^i of the particles. If there was no landmark confirmation by the user during the last step, then all of the weights are equal to 1. If the user confirmed the presence of a landmark of type L_j, then the approach prunes particles not in the vicinity of such landmarks. In particular, for every p^q the method finds all l^i so that $||(x_t^q, y_t^q), (x^i, y^i)|| < R_{obs}$. If none of the l^i is of the type L_j, then $w_{T+1}^i = 0$. Otherwise, the weight is inversely proportional to the distance $||(x_t^q, y_t^q), (x^i, y^i)||$, where l^i is the closest landmark of the correct type.

3.3.4 Sampling

The algorithm samples with higher probability particles with higher weights. It might happen, however, that all particles get a weight of 0 ("particle impoverishment"). This is why, the "Mixture MCL" method [80] samples a certain number of particles from the observation, while the "sensor resetting" approach [47] samples from the observation only when it deviates substantially from the previous distribution. The approach implemented by this work follows a similar idea. When all of the particles happen to get a weight of 0, which typically occurs when the user confirms a landmark l^i and the filter has failed to progress the particles to the vicinity of that landmark,

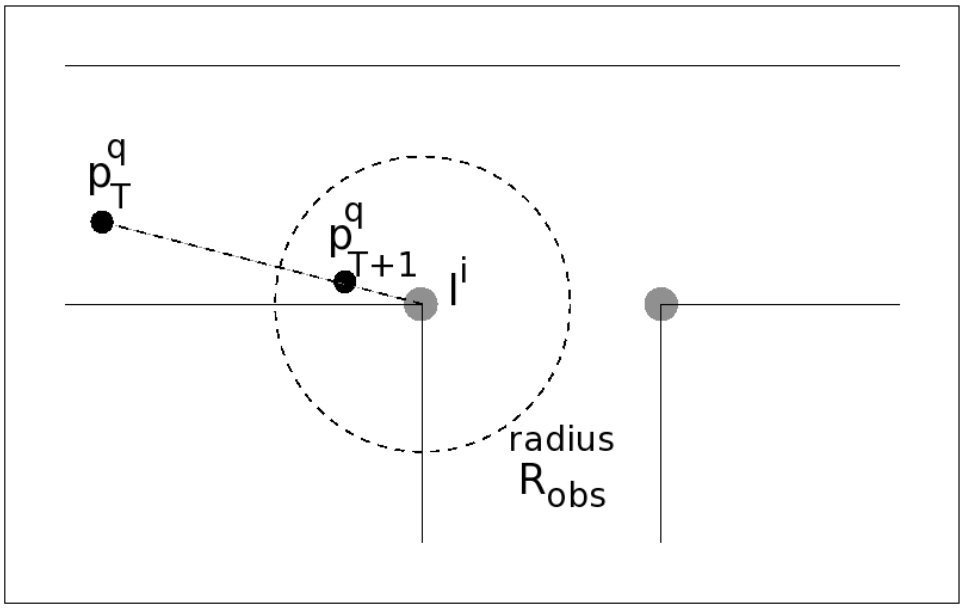

Figure 3.3: An illustration of the particle reseting process. The particle's position is moved close to the nearest landmark of the type that the user confirmed

then the particles are sampled from the observation as shown in the figure to the right. For each particle p_T^q, the method computes the landmark of the confirmed type that is closer to p_T^q. For the closest such landmark l^i, the line between p_T^q and l^i is computed. If there is line of sight between the particle and the landmark, then the new particle is sampled along the line segment $[p_T^q, l^i]$ and within the radius R_{obs}, which represents the greatest distance from which a landmark can be sensed. The line segment is introduced in the computation so as to guarantee that the new particle will not cross into a room or into a different corridor.

3.4 Experiments

In order to test our system's localization accuracy and the effect of different types of direction provision a number of experiments were executed.

3.4.1 Setup

The system has been implemented as a Java application for the open-source Google Android smart phone (ver. 1.6). A map of a building's floor on the campus of the

University of Nevada, Reno was created in the Keyhole Markup Language (KML) and loaded to the application (Fig. 3.1). The map was manually augmented with the following landmarks: *(i) 3 water coolers, (ii) 1 floor transition marked by a metal strip, (iii) 3 hallway intersections, (iv) 2 hallway turns and (v) 72 doors.* Five different paths were defined along the corridors of the building. For each path, there are two alternatives for directions, with three levels of granularity each, as specified in Sec. 4.3. Overall, six different ways to provide directions were tested per path. The application communicated the directions using text to speech software. The user was able to confirm the completion of an instruction by pressing the tactile scroll button on the smart phone or could ask for a direction to be repeated by tapping on the phone's screen. These experiments were executed with the initial version of the system.

3.4.2 Participants

Ten volunteers were involved in the experimental session. Users held the phone in their hand while holding a cane in their other (Figure 1). One of the volunteers was legally blind and assisted in the setup of the experiments. This individual pointed out landmarks, such as a metal strip on the floor, which sighted people typically ignore. Nine more volunteers were involved that were sighted users and who were blindfolded during the experiments. Typically, sighted users perform worse than people with VI when they navigate without visual cues. Some of the users had visited the building in the past and were aware of its structure, while others didn't. This discrepancy did not seem to considerably influence the efficiency of users in reaching the desired destination. Each user executed ten traversals, which corresponded to two traversals per path using different types of directions.

3.4.3 Ground Truth

To measure the true position of the user, an observer was recording the user's motion. This was achieved by placing markers on the floor every two meters. Every time the

user was crossing a marker, the observer was recording the time on a second smart phone. To recreate the true path, the assumption was that the user moves with constant speed between markers. Thus, the resolution of the ground truth is two meters.

3.4.4 Parameters

The following table provides the parameters of the results presented here. A relatively high standard deviation for the orientation parameter in the transition model was chosen because of the unreliable nature of the compass. A very small number of particles (20) was used to achieve real-time performance, while being able to save output files at the same time. Recording the status of the application (e.g., saving all the measurements, landmark confirmations and the particle filter state) takes three times longer than the actual estimation by the particle filter. Thus, in a real application the particle filter can run with at least 3 times the number of particles.

Number of Particles P	20
Landmark radius R_{obs}	1 meter
Standard Deviation in Orientation σ_θ	30°
Standard Deviation in Forward Motion σ_f	0.2 meters
Maximum Number of Tries To Find a Collision Free Transition	5

Table 3.1: Table of parameters used in the case studies

3.4.5 Success Ratio of Direction Provision

Table 4.1 provides the average distance between the destination and the actual position achieved by the user over all experiments of the same type. This table shows that most of the paths were completed successfully. In particular, in 84% of the experiments the distance between the desired destination and the achieved position was less than 2 meters, which is the resolution of the ground truth. In 92% of the experiments the error was less than 3.5 meters. It also turns out that landmark-based directions result in smaller errors and higher success ratios. Table 4.3 provides the average duration of a path until completion. The users were able to complete paths

quicker when they were not asked to confirm a larger number of landmarks, which was the expected result ("No Max" case in direction provision).

Distance from Destination	Path 1 (98.14m)	Path 2 (69.49m)	Path 3 (72.54m)	Path 4 (67.66m)	Path 5 (54.25m)
Landmark No Max	0.46	1.83	0	2.44	1.83
Landmark 9 Meters	0	1.22	0.46	2.19	1.83
Landmark 15 Meters	0.91	0.91	1.83	1.83	2.29
Metric No Max	2.74	0.61	0	2.29	1.83
Metric 9 Meters	3.05	2.74	1.22	0.91	1.22
Metric 15 Meters	4.57	0	0	2.74	1.83

Table 3.2: Average distance between destination and the user's position upon completion (m)

Path Duration	Path 1 (98.14m)	Path 2 (69.49m)	Path 3 (72.54m)	Path 4 (67.66m)	Path 5 (54.25m)
Landmark No Max	155.75	123.67	135.67	119.67	111.25
Landmark 9 Meters	201.33	177.00	212.00	192.50	138.75
Landmark 15 Meters	265.00	155.25	156.50	226.67	110.50
Metric No Max	136.25	180.00	137.50	129.50	108.50
Metric 9 Meters	242.67	252.75	173.67	219.00	169.00
Metric 15 Meters	264.00	173.67	247.00	180.00	147.33

Table 3.3: Average path duration (sec).

3.4.6 Localization Accuracy

Tables 3.4 and 3.6 provide the errors for dead reckoning and the proposed particle filter based approach. In particular it specifies the average error in meters between the final true location of the user and the estimate by the corresponding technique. The estimate from the particle filter corresponds to the particle which was closer to the average state of all particles at the last iteration. It is important to note that in most cases there were particles closer to the true position than the "average" particle.

The comparison between the two tables shows that the particle filter approach improves considerably over the result acquired just by integrating the sensor readings. The improvement ranges from a factor of 10 to a factor of 2 for different paths and

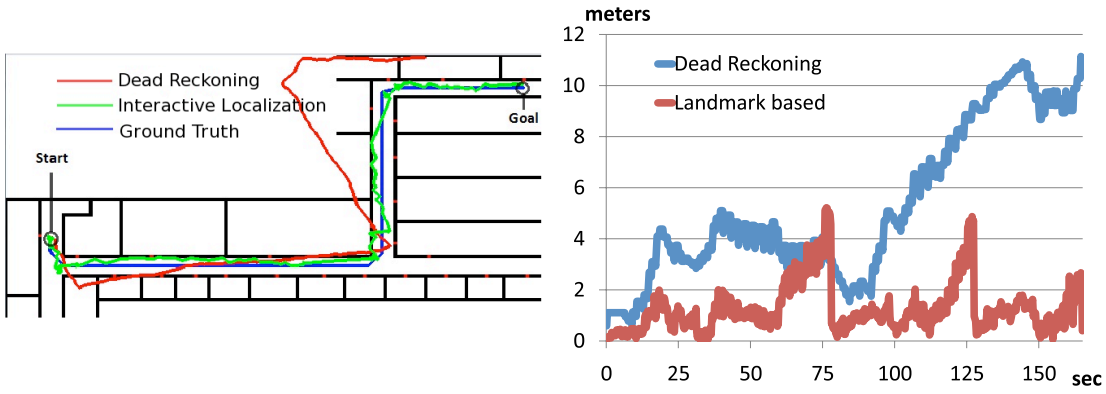

Figure 3.4: a) Ground truth vs. dead-reckoning vs. particle filtering. b) Error graph.

direction provisions. This despite the very small number of particles employed by the approach. The important point, however, is the considerable effect that the direction provision process has on the efficiency of the particle filtering algorithm. The average error in meters in the final location for the "Landmark 9 meters" approach is approx. 9.5 meters, while it goes down to 2.1 meters for the "Landmark 15 meters" approach, which also appears to be the best solution to the problem. The errors were lower for paths that contained distinctive landmarks such as hallways (in the order of 1.2-2.5m) and considerably higher for paths that corresponded to long straight line paths where all the landmarks were the same (doors). Figure 3.4 provides an error graph for a specific path/direction provision combination for dead reckoning and the particle filter approach. The expectation is that as the computational power of portable devices increases, it will be possible to run the same algorithm for a larger number of particles and thus further improve accuracy.

Dead-Reckoning	Path 1 (98.14m)	Path 2 (69.49m)	Path 3 (72.54m)	Path 4 (67.66m)	Path 5 (54.25m)
Landmark No Max	20.79	25.53	10.83	9.82	13.79
Landmark 9 Meters	10.19	32.50	17.87	8.59	13.43
Landmark 15 Meters	26.81	26.89	16.29	13.12	8.89
Metric No Max	14.91	25.69	15.49	13.41	11.65
Metric 9 Meters	18.00	28.99	23.84	5.89	7.53
Metric 15 Meters	19.55	20.95	31.44	3.88	7.90

Table 3.4: Average error of dead reckoning in final location (m).

Interactive Localization	Path 1 (98.14m)	Path 2 (69.49m)	Path 3 (72.54m)	Path 4 (67.66m)	Path 5 (54.25m)
Landmark No Max	18.80	15.51	1.32	1.14	5.31
Landmark 9 Meters	10.12	25.95	1.34	3.02	7.35
Landmark 15 Meters	5.47	4.02	3.03	3.63	2.33
Metric No Max	12.95	11.33	3.98	3.75	5.38
Metric 9 Meters	3.20	11.92	0.84	3.05	4.24
Metric 15 Meters	10.43	5.25	3.06	1.48	3.87

Table 3.5: Average error of the proposed interactive localization process (m).

Interactive Localization	Path 1 (98.14m)	Path 2 (69.49m)	Path 3 (72.54m)	Path 4 (67.66m)	Path 5 (54.25m)
Landmark No Max	14.24	16.98	0.77	2.14	7.48
Landmark 9 Meters	3.44	7.94	0.29	1.84	1.55
Landmark 15 Meters	1.12	5.91	0.15	2.49	1.18
Metric No Max	4.41	7.50	3.67	2.81	2.49
Metric 9 Meters	2.35	5.81	0.46	2.02	2.70
Metric 15 Meters	18.11	6.69	0.12	0.69	0.59

Table 3.6: Standard deviation for the error of the proposed interactive localization process (m).

Note that the errors in tables 4.1 and 3.6 are not comparable, since the first corresponds to how close the user reached the desired destination and the last two tables correspond to localization accuracy. The current method for guiding the user does not depend on the localization process and this explains why it is possible for the error in localization to be higher than the distance between the true user location and the desired one.

Chapter 4 Improved Approach

The initial approach did not address all the issues. Over the next sections, improvements are introduced and new experiments are executed to test the improved system's efficiency and localization accuracy.

4.1 Overview

The previous approach was processing offline the data collected from the sensors. Results suggested that the system could not be executed on a portable device in real time. Also, the accuracy of the system needs to be improvement even more. The improvements mainly target to the decrease of the system's computational needs and increase the localization accuracy.

4.2 Offline Process

Because of the limited processing power of the phone, we have moved some of the most computationally expensive operations to an offline process. We use the offline process to create the maps for our system and integrate additional data in order to avoid computationally expensive operations during the execution of our system.

The offline process, given the map and some accessible points, finds the accessible spaces of the map by using an eight neighbor flooding algorithm. The algorithm creates a 2D map of the space with information about the accessibility of every space. After that, the accessibility map is used in order to create particles for each landmark in the accessible space. We store these particles as part of the landmarks. When the map is saved the particles are being stored in map for each accessible landmark. These particles are going to be used later if all particle filters die during the execution of the system. If they die, the particles will be reset with the values of the particles of the closest landmark. By storing these particles on the map, we avoid searching later for accessible spaces to reset the states of the landmarks as this might become

computationally cumbersome if the number of particles is big.

After calculating the landmarks, the offline process creates a mini-map of the actual map. The mini-map is a 2D map of the actual map with lower resolution. Each cell of the mini-map corresponds to an area of 3x3 tiles of the map. The tiles represent a unique part of the space as their covering regions do not overlap each other. We distinguish, then, using the accessibility map which tiles are accessible and which are not. For each accessible tile, we perform an exhaustive search in order to find the closest to the specific tile landmarks of each type. Depending on the availability of each type of landmarks on the map, we store from one to five landmarks of every type in every tile. After storing the landmarks, we assign to them a weight depending on their distance of the center of the tile. Closer landmarks get higher weights, while landmarks that are more than 18 meters away from the center of the tile gets a weight of zero. When all accessible tiles are created, they get saved in the file where the map is saved. The coordinates of the center of the tile with all the lists of the landmarks are stored for each accessible landmark.

By storing all this data, we can avoid exhaustive searches during the execution of the system. When the user has confirmed a landmark, we can look up fast on the closest landmark for the current location of each particle. This data is also used in the case that all particles died during the confirmation of a landmark. When this happens, the system can find fast close landmarks to reset the particles.

4.3 Direction Provision

Number of landmarks and frequency of provided commands can affect the performance and usability of the system. It might appear that using all the landmarks along a path can result in a more accurate localization. However, the results of our previous experiments [6] show that using more landmarks and consequently providing commands more frequently results in slower execution of the path. Success rate and execution duration indicated that the best result is achieved when there is no upper limit for the distance between two consecutive commands. For landmarks such as

hallways or ramps the command is required as the user has to take an action at that point such as turning at hallways or walking over the ramp.

To guide the user to destination, the navigation system plans a path from current location to the desired destination. Navatar uses a 2D grid model to represent environment map and A* is a good candidate for path planning. The path is planned based on different requirements such as the shortest path or shortest travel time. For this project we have other objectives for the path that have higher priority. The path must be easy to follow, reduces localization uncertainty, and ensures user safety. The result of our preliminary study shows that uncertainty can be reduced using more distinctive and unique landmarks such as hallway intersections rather than doors which can easily be missed. Planning a path that guides the user along walls and avoid open areas will help the user to maintain their orientation and lower the uncertainty as well. To ensure users' safety, the path avoids obstacles or hazardous areas such as ramps with no handrail. The landmarks provided in directions are safe to touch as the user will confirm their existence using cane or hand. To simplify the process of following the path, similar to reducing uncertainty, we use landmarks that are distinctive and can be found easily such as hallway intersections and floor transitions.

4.4 Localization

In order to achieve better localization accuracy, a number of improvements were executed in the localization component of our system.

4.4.1 Transition Model

In the initial approach, the step length of the user was calculate before executing the experiments with a training application in order to have a better transition model for each user. However, this approach did not provide a good transition model for the user due to variations of the user's step length along the execution of a path or among different paths. Instead of keeping a single step length, we have various step lengths. In order to represent these step lengths, a multi-model estimation approach is utilized.

Instead of a single particle filter P, a number of particle filters P^k ($k \in [1, N]$) are executed simultaneously where each one keeps a different estimation of the user's step length. Each particle filter progresses its particles based on the estimation that keeps locally. The initial values of the step lengths assigned to the particle filters start from 0.37 meters and increase by 0.06 meters for each particle filter.

After some experiments, compass proved to be unreliable for localization. Due to a lot of noise produced from the buildings' infrastructure, the compass produces readings with great deviation. For this reason, the compass is not used in the later version. Since our approach does not use compass readings, we assume that the user completes successfully turns every time the phone provides turn directions. When this happens, the orientation θ^{ik} of each particle p_T^{ik} changes based on the direction the user was instructed to turn by 90 degrees. The approach collects all the pedometer readings that have been produced by the accelerometer during the last time step. The pedometer typically returns either zero or one step measured. This value has to be translated into a distance estimate.

4.4.2 Observation Model

In order, though, to have a better estimation, we, now, prune each particle p_T^{ik} based on the direction the landmark l^i of type L_j is. The directions provided to the user specifies in which direction the landmark l^i will be found. We assign a weight $w_{T+1}^{ik} = 0$ to the particles that detect the landmark on the wrong side based on their orientation θ^{ik} and the angle their are facing the landmark.

4.4.3 Sampling

At the beginning of the sampling process the particle filter P^K with the highest weight is calculated. This happens by summing the weights of all particles p_T^{ik} for each particle filter P^k and choosing the maximum sum. The algorithm processes each particle filter P^k separately. Particles p_T^{ik} from different particle filters P^k cannot be sampled. For each particle filter P^k, we sample the particles p_T^{ik} with the higher weights w_T^{ik}. In

case that all particles p_T^{ik} in a particle filter P^k die, the whole particle filter copies the particle filter P^K with the highest weight. It, then, changes its step length based on a gaussian distribution whose mean is the step length of the best particle filter P^K and standard deviation of 0.03 meters. By doing that, the estimation of the user's step length is becoming more accurate. In the special case that all particle filters die, their particles get reset to the closest landmark of the type that was confirmed by the user [6]. The process of particle resetting is the same as in the initial approach with one addition. The estimations of the step lengths are also reset to their default values.

4.5 Experiments

A new set of experiments had to be executed in order to test the new system and compare it with the previous approach.

4.5.1 Setup

The system was implemented in Java for the Android mobile platform. Maps of the first and second floor of the SEM building of University of Nevada, Reno were created with the use of the offline process. The maps include the following landmarks: *(i) 211 doors, (ii) 15 hallway intersections, (iii) 6 water coolers, (iv) 1 floor transition, (v) 2 elevators, (vi) 8 sets of stairs and (vii) 1 ramp.* For the experiments, 10 paths were tested. The application provided the instructions through a text to speech synthesis library embedded on the Android platform. The user could confirm the landmarks by pressing anywhere on the screen. This set of experiments was executed with the improved version of the system.

4.5.2 Participants

Eight volunteers participated in the experiments. The volunteers were sighted people who were blindfolded. Some of the participants had visited the building before while others had never visited the building before. The volunteers were provided with a

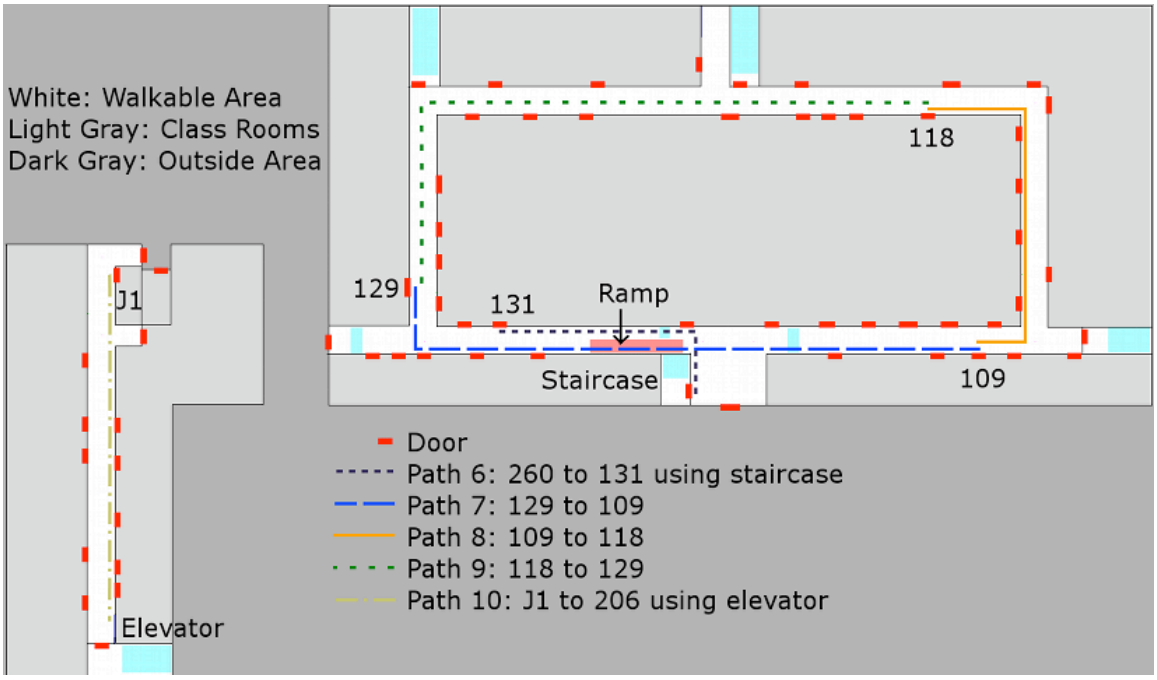

Figure 4.1: The map of the first floor

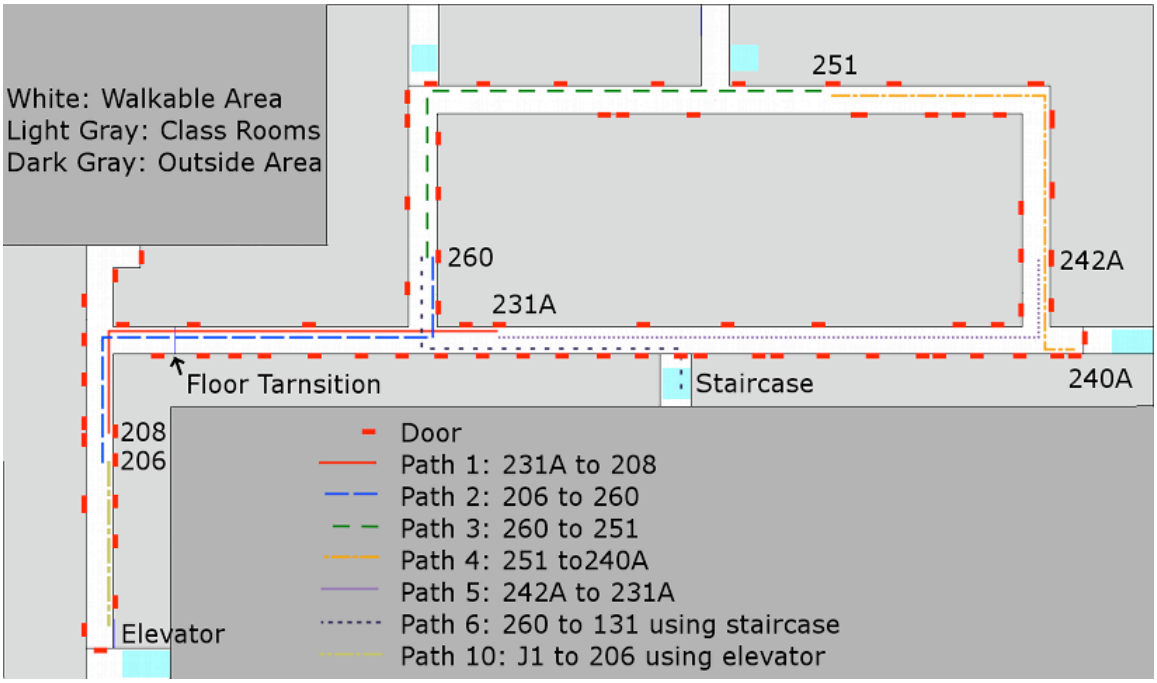

Figure 4.2: The map of the second floor

white cane and the phone. Before the actual experiments the users executed a training path to familiarize themselves with the cane and the inability to see. This was proven to be helpful for the users as they could navigate easier with the cane in the actual experiments.

4.5.3 Ground Truth

A commercial robot localization system called Hagisonic StarGazer was used to capture the ground truth. The product is consisting of (1) set of passive landmarks with unique ID mounted on the ceiling; (2) an infrared camera that outputs ID, distance, and angle of the closest landmark; and (3) an open source library to operate the camera. A map of landmarks is created using the software developed for this purpose. To use the system for human tracking, the camera and a battery are installed on a belt that the user wears during experiments. The user carries a tablet that records the landmark information received from the camera and calculates the user location from the map based on the relative landmark location. The accuracy of the camera is 2cm when installed on a flat surface; but the camera is sensitive to tilt and might result in incorrect reading when installed on a belt. To reduce the noise the ground truth is smoothed using an outliers detection algorithm.

4.5.4 Parameters

The following table holds the parameters that the system used to run the experiments. We chose 7 particle filters in order to have a wide variety of step lengths. The step length starts from 0.37 meters for the first particle filter and is increased by 0.06 meters for the next one. These values represent the mean values of a Gaussian distribution with standard deviation of 0.03 meters.

Number of Particle Filters	7
Number of Particles per Particle Filter P	50
Landmark radius R_{obs}	1.5 meters
Standard Deviation in Forward Motion σ_f	0.03 meters

Distance from Destination	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8
Path 109-118	0.82	1.53	0.71	2.00	1.03	0.99	1.63	1.77
Path 118-129	24.66	1.55	2.04	0.46	0.89	2.30	0.75	1.35
Path 129-109	0.63	3.11	3.55	2.75	3.84	0.66	1.54	4.83
Path 206-260	1.39	0.44	1.34	1.19	0.95	1.29	1.58	0.71
Path 231A-208	0.37	1.78	0.19	2.15	2.05	1.57	0.56	1.19
Path 242A-231A	22.18	22.57	12.65	3.33	16.58	16.67	1.12	16.51
Path 251-240A	0.76	0.97	1.28	1.26	1.83	1.25	1.61	1.36
Path 260-131	3.68	2.91	0.96	10.35	3.94	3.06	1.21	6.32
Path 260-251	1.24	1.34	0.74	1.21	0.93	1.19	1.29	1.46
Path J1-206	1.03	1.51	2.72	3.49	0.97	1.69	3.84	1.99

Table 4.1: Distance between destination and the user's position upon completion (m)

4.5.5 Success Ratio of Direction Provision

Table 4.1 provides the average distance between the destination and the actual position achieved by the user. This table shows that most of the paths were completed successfully. In particular, in 79% of the experiments the system guided the user to the desired destination within an average distance of 1.5 meters. In 9% of the experiments the error the system guided the user to a location adjacent to the desired destination with an average distance of 2.75 meters. The calculation of the error was implemented by comparing the ground truth position with the location estimation of the system.

To calculate the state estimation of the system the K-Means clustering algorithm was used. The algorithm clusters all the particles, not taking into consideration if they belong in different particle filters. The clusters are circles in the state space with maximum radius of 3 meters. The K-Means algorithm starts with 3 clusters and adds more clusters if the particles are scattered. Respectively, when the particles gather together, the clusters merge. The cluster with the most amount of particles is chosen and its mean is considered to be the state estimation. Fig. 4.3 shows an example of the localization error during the progression of a path. Each graph represents the progression of one of the particle filters. The particle filters that do not have a correct estimation of the user's step length have higher localization error. It is clear that the

Distance from Destination	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8
Path 109-118	0.98	1.14	0.76	2.38	0.98	1.05	1.05	1.01
Path 118-129	16.79	1.75	1.35	1.47	1.99	1.97	1.27	1.20
Path 129-109	0.54	3.10	1.42	2.81	2.69	0.96	1.39	1.32
Path 206-260	1.55	1.03	0.40	1.29	1.43	1.07	1.13	1.29
Path 231A-208	0.65	0.41	1.21	0.86	0.87	0.91	0.99	0.77
Path 242A-231A	8.66	4.95	6.97	2.00	0.71	5.80	1.40	2.17
Path 251-240A	0.88	0.49	0.55	0.70	0.67	0.19	0.52	0.10
Path 260-131	1.23	1.36	0.45	0.62	5.49	0.49	0.65	6.11
Path 260-251	0.48	1.18	0.54	1.15	0.27	0.82	0.77	0.73
Path J1-206	1.09	1.05	2.61	0.73	0.27	2.13	0.93	0.86

Table 4.2: Distance between ground truth and destination (m)

localization error is getting higher as we progress the path, but drops instantly when the user confirms a landmark. When a landmark is confirmed, the particle filters that die are resampled with a better step length estimation.

4.5.6 Off course correction

In some of the experiments users might miss a landmark or move further away from the landmark confirmed. In this case the system adapts the instructions depending on what the location estimation of the user is. For example, in a path the user had to follow this set of instructions:

1. "Follow the wall to your left until you reach a hallway intersection"
2. "Turn left"
3. "Follow the wall to your right until you reach a ramp"
4. "Take the ramp down"
5. "Follow the wall to your right until you reach a hallway intersection"
6. "Move straight until you pass the hallway"

However the user advanced further away from the end of the ramp and reached the hallway intersection. In this case the phone provided the following instructions:

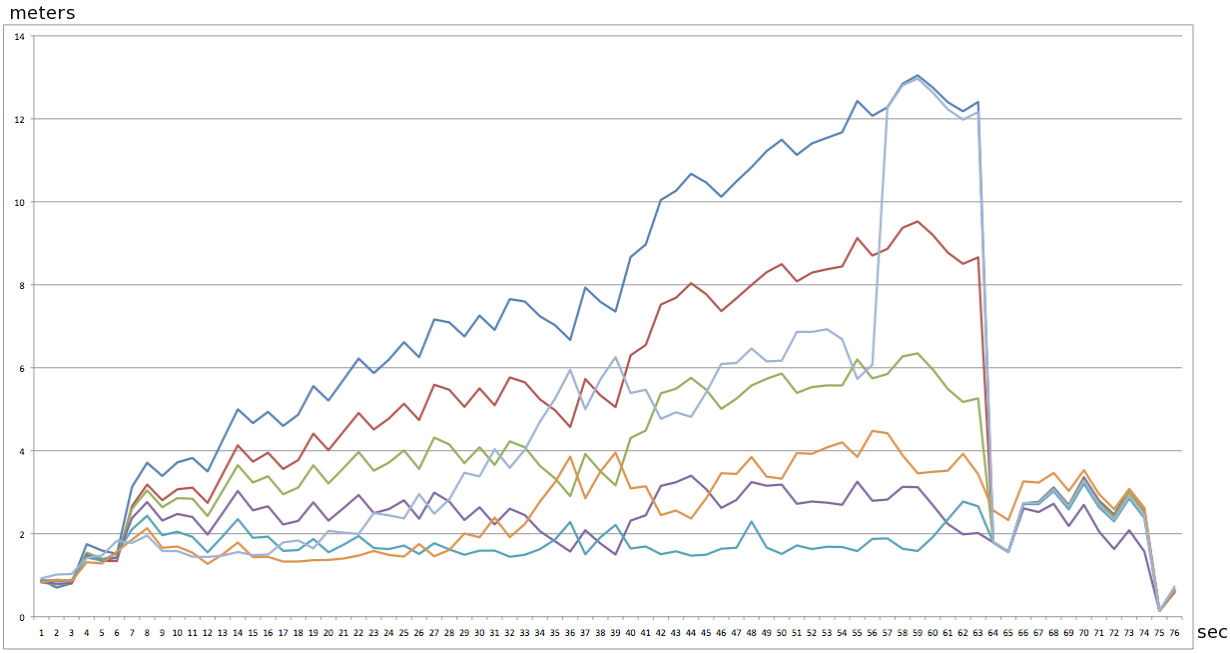

Figure 4.3: A graph showing the localization error during the execution of a path. The different lines represent the distance of the different particle filters from the actual location of the user

1. "Follow the wall to your left until you reach a hallway intersection"
2. "Turn left"
3. "Follow the wall to your right until you reach a ramp"
4. "Take the ramp down"
5. "Move straight until you pass the hallway"

We see in this case that the phone provided a more suitable instruction since the user had already advanced in the path.

4.5.7 Computational overhead

The system managed to run all the experiments smoothly without running into issues due to the limited computational power of the phone. The system must be able to process all the data it receives from the sensors and the user in real time. That means that it must run a full localization loop in under 1 second. To achieve this, we used

Profiling	Avg total time	Avg time per step	Time std	Max step time	Avg steps	Avg time for transition	Avg time for observation	Avg time for sampling
Path 1	2808.13	39.14	22.11	95.13	71.63	1281.75	2048.13	39.88
Path 2	4332.13	37.72	21.47	99.75	114.75	1970.88	3116.75	58.25
Path 3	5237.13	39.88	23.03	106.25	131.5	2220.38	3764.75	64
Path 4	3444.13	39.23	23.40	104.88	87.88	1556.75	3764.75	35.25
Path 5	3114.75	40.01	22.29	106.38	77.25	1409.38	2241.63	44.38
Path 6	4387.88	39.41	23.15	114.13	111.38	2007.5	3158.5	40.5
Path 7	3397.88	39.02	22.04	99.5	86.38	1490.25	2358.38	43.25
Path 8	4861.86	41.25	24.16	118	117.86	2118.57	3549.57	69.71
Path 9	3953.5	40.39	23.24	105.13	97.13	1811.75	2883.75	46.88
Path 10	3684.38	38.69	21.79	92.5	95.5	1676.13	2629	39.13

Table 4.3: Profiling data (msec)

a small number of particles (350), however, profiling data during the execution of the experiments suggest that we can increase the number of particles. An average particle filter step takes about 40 ms while the maximum step time reached 114.13 ms. These data indicate we can increase the number of particles at least 8 times.

Chapter 5 Conclusion

After testing the initial and the improved approach this section provides an overall summary of the work and discusses possible future steps.

5.1 Summary

The first part of this work presented a study on the feasibility of navigating a user with VI through an indoor environment using a minimalistic and interactive sensing approach achievable with a smart phone. The sensors used in the experiments are inexpensive and available on popular portable devices. Nevertheless, they are also highly erroneous. For instance, compass sensors, especially cheap ones, perform very poorly in indoor environments due to metal structures and electro-magnetic noise. This was also the case in the building where the experiments presented in this work were executed. Despite this challenge, it was still possible to track a human user who does not have any visual feedback with sufficient accuracy through an interactive localization process.

The latest user studies navigated in more complex environments that involve a larger variety of landmarks, for instance, multiple floors that involved elevators and ramps. Furthermore, in the current system the next direction is provided automatically based on the user's confirmation of successfully executing the previous instruction and the location estimation. The overall approach combines manual (upon user confirmations) and automatic direction provisions (based on localization estimates).

The user studies have shown that the success of the path depends mostly on the type of the path that is chosen for the user. Paths that have few and distinctive landmarks are more successful than paths that require door counting. Paths where the user had to confirm more than 2 doors in a row usually failed to localize the user accurately. In the future, the path planning approach should take into consideration the type of landmarks the user will have to confirm along each path and select the

path that will lead the user to the target destination with the highest probability.

This line of research opens the door to exciting new applications for methods from robotics in the area of human-centered autonomous intelligent systems. For instance, minimalistic approaches could be employed to improve localization accuracy while maintaining a low computational overhead. Similarly, it is interesting to investigate how to automatically plan alternative paths that lead along a larger number of landmarks or along more distinguishable landmarks, such as preferring a hallway confirmation over a door. Such planning under uncertainty tools may significantly boost chances of the user successfully arriving at the destination and the localization estimate being more accurate.

5.2 Future work

Although a system that can successfully navigate a user with visual impairments was presented, there is still room for many improvements. First, and most important, the path planning process should be changed. So far, the A* algorithm was choosing the shortest path for the user to follow. Although this might decrease the time it takes to follow the path, it might compromise the success of the path. A different objective function must be implemented for the path planning process. The objective function should take into consideration the number of distinctive landmarks along the execution of the path in order to provide higher probability of successful completion. In this manner, a longer path might be selected for the user to follow, but it will ensure the highest probability of successfully guiding the user to the target destination.

Also, the highly erroneous sensors can be replaced with more accurate sensors. The phone accelerometer can be replaced with an external wireless accelerometer that can provide more accurate data about the steps of the user.

2D maps of an indoor environment, as used here, can be acquired from architectural blueprints. Nevertheless, it may be more useful to use richer types of representations. 3D virtual models can be employed to more accurately represent indoor environments with multiple levels and features like low ceilings, ramps, uneven floors

and rails, which are impediments to navigation for users with VI. It is interesting to investigate how to extract landmarks such as doors or staircases automatically from the geometry of such models in order to utilize them in navigation and localization tools for individuals with VI. 3D models may be easier to annotate than 2D maps as users can more easily visualize the space.

Finally, it is also possible to make use of more realistic models of human motion [7]. Now the transition model representing the human motion is a simple unicycle-like system. This model can later be improved to represent the more complex model of the human motion.

Bibliography

[1] *Stochastic Models, Estimation and Control*, volume 2 of *Mathematics in Science and Engineering*. Maybeck, P. S., 1982.

[2] G. A. Ackerson and K. S. Fu. On state estimation in switching environments. *IEEE Transactions on Automatic Controls*, AC-15(1), 1970.

[3] H. Alt, R. Fleischer, M. Kaufmann, K. Mehlhorn, S. Näher, S. Schirra, and C. Uhrig. Approximate motion planning and the complexity of the boundary of the union of simple geometric figures. pages 281–289, 1990.

[4] N. M. Amato, O. B. Bayazit, L. K. Dale, C. Jones, and D. Vallejo. OBPRM: An obstacle-based PRM for 3D workspaces. pages 155–168, 1998.

[5] Tomohiro Amemiya, Jun Yamashita, Koichi Hirota, and Michitaka Hirose. Virtual leading blocks for the deaf-blind: A real-time way-finder by verbal-nonverbal hybrid interface and high-density rfid tag space. In *Proc. of the IEEE Virtual Reality*, Washington, DC, USA, 2004.

[6] E. Apostolopoulos, N. Fallah, E. Folmer, and K. E. Bekris. Feasibility of interactive localization and navigation of people with visual impairments. In *11th IEEE Intelligent Autonomous Systems (IAS-10)*, Ottawa, CA, August 2010 2010.

[7] G. Arechavaleta, J.-P. Laumond, H. Hicheur, and A. Berthoz. An optimality principle governing human walking. In *IEEE Transactions on Robotics*, volume 24, 2008.

[8] F. Aurenhammer. Voronoi diagrams: A survery of a fundamental geometric data structure. *ACM Computing Surveys*, 23:345–405, 1991.

[9] A Parush B Tsuji, G Lindgaard. Landmarks for navigators who are visually impaired. In *Proceedings International Cartography Conference*, 2005.

[10] M. Bessho, S. Kobayashi, N. Koshizuka, and K. Sakamura. Assisting mobility of the disabled using space-identifying ubiquitous infrastructure. In *Proc. of the 10th Int. ACM SIGACCESS Conf. on Computers and Accessibility (ASSETS)*, pages 283–284, Halifax, Nova Scotia, Canada, 2008. ACM.

[11] M. Brambing. *Speech, place and action*, chapter Language and geographic orientation for the blind, pages 203–218. John Wiley & Sons, New York.

[12] M. S. Branicky, S. M. LaValle, K. Olson, and L. Yang. Quasi-randomized path planning. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1481–1487, 2001.

[13] E. Brunskill, L. Kaelbling, T. Lozano-Perez, and N. Roy. Continuous-state pomdps with hybrid dynamics. In *Proc. of the Tenth Inter. Symposium on Artificial Intelligence and Mathematics*, Fort Lauderdale, FL, 2008.

[14] F. Cavallo, A. M. Sabatini, and V. Genovese. A step towards gps/ins personal navigation systems: real-time assissment of gait by foot inertial sensing. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1187–1191, Edmonton, Canada, August 2-6 2005.

[15] C.B. Chang and M. Athans. State estimation for discrete systems with switching parameters. *IEEE Transactions on Aerospace and Electronic Systems*, 1978.

[16] B. Chazelle. Approximation and decomposition of shapes. In J. T. Schwartz and C. K. Yap, editors, *Algorithmic and Geometric Aspects of Robotics*, pages 145–185. Lawrence Erlbaum Associates, Hillsdale, NJ, 1987.

[17] Vlad Coroama and Felix Rothenbacher. The chatty environment – providing everyday independence to the visually impaired. In *Workshop on Ubiquitous Computing for Pervasive Healthcare Applications at Ubicomp*, 2003.

[18] Edoardo D'Atri, Carlo Maria Medaglia, Alexandru Serbanati, Ugo Biader Ceipidor, Emanuele Panizzi, and Alessandro D'Atri. A system to aid blind people in the mobility: A usability test and its results. In *ICONS '07: Proceedings of the Second International Conference on Systems*, page 35, Washington, DC, USA, 2007. IEEE Computer Society.

[19] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications, 2nd Ed.* Springer-Verlag, Berlin, 2000.

[20] F. Dellaert and S. Tariq. A multi-camera pose tracker for assisting the visually impaired. In *1st IEEE Workshop on Computer Vision Applications for the Visually Impaired*, June 20-25 2005.

[21] E. Foulke. chapter Perception, cognition, and mobility of blind pedestrians., pages 55–76. Spatial abilities: Development and physiological foundations, New York, 1982.

[22] C. R. Gallistel. *The organization of learning.* MIT Press, Cambridge, MA, 1990.

[23] R. Geraerts and M. Overmars. Sampling techniques for probabilistic roadmap planners. In *Proceedings International Conference on Intelligent Autonomous Systems*, 2004.

[24] S. K. Ghosh and D. M. Mount. An output-sensitive algoithm for computing vsibility graphs. *SIAM Journal of Computing*, 20:888–910, 1991.

[25] Scott Gifford, Jim Knox, Jonathan James, and Atul Prakash. Introduction to the talking points project. In *Assets '06: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*, pages 271–272, New York, NY, USA, 2006. ACM.

[26] Nicholas A Giudice, Jonathan Z Bakdash, and Gordon E Legge. Wayfinding with words: spatial learning and navigation using dynamically updated verbal descriptions. *Psychol Res*, 71(3):347–58, May 2007.

[27] N. J. Gordon, D. J. Salmond, and A. F. M. Smithm. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings on Radar and Signal Processing*, 140(2):107–113, 1993.

[28] D.A. Guth, E.W. Hill, and J.J. Rieser. Tests of blind pedestrians' use of traffic sounds for street-crossing alignment. *Journal of Visual Impairment & Blindness*, 83(9):461–468, 1989.

[29] J. A. Hesch, F. M. Mirzaei, G. L. Mariottini, and S. I. Roumeliotis. A 3d pose estimator for the visually impaired. In *Proc. of the IEEE/RSJ Intern. Conference on Intelligent Robots and Systems (IROS)*, pages 2716–2723, Oct. 11-15 2009.

[30] J. A. Hesch and S. I. Roumeliotis. An indoor localization aid for the visually impaired. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3545–3551, Roma, Italy, April 10-14 2007.

[31] Joel A. Hesch and Stergios I. Roumeliotis. Design and analysis of a portable indoor localization aid for the visually impaired. *The International Journal of Robotics Research*, 29(11):1400–1415, 2010.

[32] Harlan Hile, Ramakrishna Vedantham, Gregory Cuellar, Alan Liu, Natasha Gelfand, Radek Grzeszczuk, and Gaetano Borriello. Landmark-based pedestrian navigation from collections of geotagged photos. In *MUM '08: Proceedings of the 7th International Conference on Mobile and Ubiquitous Multimedia*, pages 145–152, New York, NY, USA, 2008. ACM.

[33] A. Hub, J. Diepstraten, and T. Ertl. Design and development of an indoor navigation and object identification system for the blind. In *Proc. of the 6th Int. ACM SIGACCESS Conf. on Computers and Accessibility (ASSETS)*, pages 147–152, Atlanta, GA, USA, 2004. ACM.

[34] Yutaka Inoue, Akio Sashima, and Koichi Kurumatani. Indoor positioning system using beacon devices for practical pedestrian navigation on mobile phone. In *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing*, UIC '09, pages 251–265, Berlin, Heidelberg, 2009. Springer-Verlag.

[35] T. Ishikawa, M. Kourogi, T. Okuma, and T. Kurata. Economic and synergistic pedestrian tracking system for indoor environments. In *Soft Computing and Pattern Recognition, 2009. SOCPAR '09. International Conference of*, pages 522 –527, 2009.

[36] A. A. Kalia, G. E. Legge, and N. A. Giudice. Learning building layouts with non-geometric visual information: the effects of visual impairment and age. *Perception*, 37(11):1677–99, 2008.

[37] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(35-45), 1960.

[38] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars. Probabilistic roadmaps for path planning in high dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.

[39] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. 5(1):90–98, 1986.

[40] O. Koch and S. Teller. Wide-area egomotion estimation from known 3d structure. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, June 17-22 2007.

[41] V. Kulyukin, C. Gharpure, J. Nicholson, and S. Pavithran. Rfid in robot-assisted indoor navigation for the visually impaired. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1979–1984, Sendai, Japan, Sept. 28-Oct.2 2004.

[42] A. Ladd and L. E. Kavraki. Measure theoretic analysis of probabilistic path planning. 20(2):229–242, 2004.

[43] A. M. Ladd, K. E. Bekris, A. Rudys, L. E. Kavraki, and D. S. Wallach. On the feasibility of using wireless ethernet for indoor localization. *IEEE Transactions on Robotics and Automation*, 20(3):555–559, June 2004.

[44] A. M. Ladd, K. E. Bekris, A. Rudys, G. Marceau, L. E. Kavraki, and D. S. Wallach. Robotics-based location sensing using wireless ethernet. In *Eight ACM International Conference on Mobile Computing and Networking (MOBICOM 2002)*, Atlanta, GA, September 2002. ACM Press, ACM Press.

[45] O. Lahav and D. Mioduser. Multisensory virtual environment for supporting blind persons' acquisition of spatial cognitive mapping – a case study. In *Proc. of the World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pages 1046–1051, 2001.

[46] Orly Lahav and David Mioduser. Multisensory virtual environment for supporting blind persons' acquisition of spatial cognitive mapping – a case study. In Craig Montgomerie and Jarmo Viteli, editors, *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2001*, pages 1046–1051, Norfolk, VA, 2001. AACE.

[47] S. Lenser and M. Veloso. Sensor resetting localization for poorly modelled mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics and Automtion (ICRA)*, San Francisco, April 2000.

[48] D. Leven and M. Sharir. Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams. *Discrete and Computational Geometry*, 2:9–31, 1987.

[49] X. R. Li, X. Zhi, and Y. Zhang. Multiple-model estimation with variable structure, part iii: model-group switching algorithm. In *IEEE Trans. Aerosp. Electron. Syst.,*, volume 35, pages 225–241, 1999.

[50] M. Littman, L. Kaelbling, and A. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[51] J M Loomis, R L Klatzky, and R G Golledge. Navigating without vision: basic and applied research. *Optom Vis Sci*, 78(5):282–9, May 2001.

[52] Jack M. Loomis, Reginald G. Golledge, and Roberta L. Klatzky. Navigation system for the blind: Auditory display modes and guidance. *Presence: Teleoper. Virtual Environ.*, 7(2):193–203, 1998.

[53] Kevin Lynch. *The Image of the city*. MIT Press., Cambridge, Ma.,, 1960.

[54] M. May. Wayfinding technologies for people with visual impairments: Research and development of an integrated platform., 2001.

[55] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in target tracking: A survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34:103–123, 1998.

[56] Alexandra Millonig and Katja Schechtner. Developing landmark-based pedestrian-navigation systems. *Intelligent Transportation Systems, IEEE Transactions on*, 8(1):43–49, March 2007.

[57] David Mioduser and Orly Lahav. Anticipatory cognitive mapping of unknown spaces by people who are blind using a virtual learning environment. In *ICLS '04: Proceedings of the 6th international conference on Learning sciences*, pages 334–341. International Society of the Learning Sciences, 2004.

[58] A. Mulloni, D. Wagner, I. Barakonyi, and D. Schmalstieg. Indoor positioning and navigation with camera phones. *Pervasive Computing, IEEE*, 8(2):22 –31, 2009.

[59] N. J. Nilsson. A mobile automaton: An application of artificial intelligence techniques. In *1st International Conference on Artificial Intelligence*, pages 509–520, 1969.

[60] S. M. Oh, S. Tariq, B. Walker, and F. Dellaert. Map-based priors for localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004.

[61] Jyri Rajamäki, Petri Viinikainen, Julius Tuomisto, Thomas Sederholm, and Miika Säämänen. Laureapop indoor navigation service for the visually impaired in a wlan environment. In *Proceedings of the 6th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications*, pages 96–101, Stevens Point, Wisconsin, USA, 2007. World Scientific and Engineering Academy and Society (WSEAS).

[62] L. Ran, S. Helal, and S. Moore. Drishti: An integrated indoor/outdoor blind navigation system and service. *IEEE International Conference on Pervasive Computing and Communications*, page 23, 2004.

[63] I. M. Rekleitis. A particle filter tutorial for mobile robot localization. Technical Report TR-CIM-04-02, Centre for Intelligent Machines, McGill Univrsity, Montreal, Quebec, Canada, 2002.

[64] T. H. Riehle, P. Lichter, and N. A. Giudice. An indoor navigation system to support the visually impaired. In *30th Annual International Conference of the IEEE Conference on Engineering in Medicine and Biology Society (EMBS)*, pages 4435–4438, Aug. 2008.

[65] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential fields. 8(5):501–518, October 1992.

[66] D. Rina and J. Pearl. Generalized best-first search strategies and the optimality of a*. *Journal of the ACM*, 32(3):505–536, 1985.

[67] D. A. Ross and B. B. Blasch. Development of a wearable computer orientation system. *Personal Ubiquitous Comput.*, 6(1):49–63, 2002.

[68] David A. Ross and Bruce B. Blasch. Development of a wearable computer orientation system. *Personal Ubiquitous Comput.*, 6(1):49–63, 2002.

[69] David A. Ross and Alexander Lightman. Talking braille: a wireless ubiquitous computing network for orientation and wayfinding. In *Assets '05: Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility*, pages 98–105, New York, NY, USA, 2005. ACM.

[70] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, chapter 4, pages 97–104. Prentice Hall, 2003.

[71] S. K. Semwal. Wayfinding and navigation in haptic virtual environments. *Multimedia and Expo, IEEE International Conference on*, 0:143, 2001.

[72] S. Shoval, J. Borenstein, and Y. Koren. Auditory guidance with the navbelt - a computerized travel aid for the blind. *IEEE Transactions on Systems, Man and Cybernetics*, 28(3):459–467, August 1998.

[73] T. Siméon, J.-P. Laumond, and C. Nissoux. Visibility based probabilistic roadmaps for motion planning. *Advanced Robotics*, 14(6), 2000.

[74] S. Skaff, A. Rizzi, H. Choset, and M. Tesch. Context identification for efficient multiple-model state estimation of systems with cyclical intermittent dynamics. *IEEE Transactions on Robotics*, 27(1), February 2011.

[75] R. Smith, M. Self, and P. Cheeseman. *Estimating Uncertain Spatial Relationships in Robotics*. Springer Verlag, 1990.

[76] Yehuda Sonnenblick. An indoor navigation system for blind individuals. In *Proceedings of the 13th annual Conference on Technology and Persons with Disabilities*, 1998.

[77] Brian J Stankiewicz and Amy A Kalia. Acquisition of structural versus object landmark knowledge. *J Exp Psychol Hum Percept Perform*, 33(2):378–90, Apr 2007.

[78] S. Suksakulchai, S. Thongchai, D. M. Wilkes, and K. Kawamura. Mobile robot localization using an electronic compass for corridor environment. In *Proceedings of the IEEE International Conference of Systems, Man and Cybernetics, Nashville, Tennessee*, 2000.

[79] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, September 2005.

[80] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2000.

[81] B. Tjan, P. Beckmann, N. Giudice, and G. Legge. Digital sign system for indoor wayfinding for the visually impaired. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition - Workshop on Computer Vision Applications for the Visually Impaired*, San Diego, CA, June 20-25 2005.

[82] E. C. Tolman. Cognitive maps in rats and men. *Psychol Rev*, 55(4):189–208, Jul 1948.

[83] B. Tsuji, G. Lindgaard, and A. Parush. Landmarks for navigators who are visually impaired. In *Proceedings of the International Cartography Conference*, Coruna, Spain, July 9-16 2005.

[84] P. Van Dorp and C. A. Groen, F. Human walking estimation with radar. *IEE Proceedings, Radar, Sonar and Navigation*, 150(5):356–365, 2003.

[85] S. Willis and S. Helal. Rfid information grid for blind navigation and wayfinding. In *ISWC '05: Proc. of the Ninth IEEE International Symposium on Wearable Computers*, pages 34–37, Washington, DC, USA, 2005. IEEE Computer Society.

[86] D. Yuan and R. Manduchi. Dynamic environment exploration using a virtual white cane. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 243–249, San Diego, CA, June 20-25 2005.