Journal of King Saud University – Computer and Information Sciences 34 (2022) 7014–7024

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Dynamic indoor path planning for the visually impaired

Abir Benabid Najjar [a,*], Arwa Rashed Al-Issa [b], Manar Hosny [b]

[a] Software Engineering Department, King Saud University, Saudi Arabia
[b] Computer Science Department, King Saud University, Saudi Arabia

ARTICLE INFO

Article history:
Received 19 August 2021
Revised 29 January 2022
Accepted 3 March 2022
Available online 28 March 2022

Keywords:
Indoor navigation
Visually impaired
Ant colony algorithm
Deviation detection
Usability evaluation

ABSTRACT

Independent indoor navigation for the Visually Impaired (VI) is extremely challenging, especially in unfamiliar environments. This paper presents the development process for a navigator system for the VI that plans an optimized path to safely lead them to their desired destination. The first contribution of this research paper is proposing a path planning algorithm, based on the well-known Ant Colony Optimization (ACO) technique. The algorithm computes an optimized route considering the user's preferences while avoiding collisions with fixed obstacles. The algorithm's output consists of the moving directions for VI users, which are delivered to them via audio commands to direct their movements. A path following algorithm was also implemented so that the calculated path is dynamically updated whenever the VI diverts from the initially pre-defined path. This functionality utilizes a step and deviation detector that counts and monitors the user's steps, matching them with the expected path. Finally, a thorough usability evaluation involving both visually impaired and unimpaired users was conducted; the results indicate that the system ultimately achieves its intended goals in terms of effectiveness, efficiency and most importantly, user satisfaction.

1. Introduction

According to the World Health Organization, at least 2.2 billion people have vision impairment or blindness (World Health Organization, 2019). Recent technological developments, especially in mobile computing, can provide technical assistance to support Visually Impaired (VI) people in their everyday tasks, improve the quality of their life, and allow them to integrate into society. A wide range of assistive technologies, wearable or handheld, are currently available to aid users with impairment. The tremendous expansion of navigation systems has highly contributed to the development of assistive tools that guide the VI people, especially in outdoor environments. Yet, one of the most challenging problems faced by the VI is to safely navigate in unfamiliar indoor places. Indoor navigation finds applications in hospitals, airports, shopping centers, schools, etc. (Alqahtani et al., 2018). For such applications, most proposals focus on indoor positioning and localization (Sakpere et al., 2017) and/or obstacles detection (Veeranjaneyulu et al., 2019). In contrast, our work focuses on the navigator module. This aims to help the VI to find an optimal path in an unfamiliar indoor environment, which is vital for their independence and safety.

Thus, the main motivation of this research work is to facilitate the navigation process for the VI, through developing a useful indoor navigation mobile application that can safely lead them to the desired destination. The development process followed a User-Centered Design (UCD), as the VI user was the focus throughout the system development life cycle, from the requirements gathering, through the design of the algorithms, to the usability evaluation. The system requirements were initially distilled from a needs-finding study conducted in the local context (Hosny et al., 2015). The system design, presented in this paper, combines state-of-the-art technologies embarked in mobile devices with optimization techniques for path planning and path following. The contribution of this research work is three-fold:.

First, path planning is a significant component in any navigation system. Since the 1970 s, the path planning problem has been an interesting research area, especially in the field of robotics (Patle and G. Babu L, A. Pandey, D. R. K. Parhi, and A. Jagadeesh, , 2019). It aims to find the optimal route between two points, which saves time, effort, and resources. However, bringing these techniques from the robotics field and applying them in navigation systems

* Corresponding author.
E-mail address: abbenabid@ksu.edu.sa (A. Benabid Najjar).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2022.03.004
1319-1578/

requires further design constraints to ensure the safety of the user. Furthermore, an optimal path for the VI is often not simply the shortest path since the user's preferences and needs may vary due to the underlying path features and the environment's landscape. Thus, this paper proposes a path planning algorithm, based on Ant Colony Optimization (ACO), a technique that can find an optimized route whilst taking into consideration the VI user's preferences and avoiding collisions with obstacles. The human factors, particularly the special needs of the VI user have been integrated into each step of the algorithm and objective function.

Second, most navigation systems have a mechanism for path following to guarantee the reliability and correctness of the navigator. The systems should check if the navigating person is following the suggested path or not and notify the user if there is any deviation occurring. However, following the instructions given by the navigator is particularly challenging for the VI who entirely rely on the system to safely lead them to their desired destination. Hence, the computed path should be dynamically updated whenever the VI user diverts from the initially planned path. To achieve this, our proposed path planner is integrated with a path following module that detects any deviation from the planned path, alerts the VI user, and triggers a dynamic path re-planning.

Third, while many previous studies provide only a simulation analysis rather than real-time applications and experiments, this paper presents and discusses the results of a usability evaluation of the proposed system. The system usability has been assessed in terms of effectiveness, efficiency, and user satisfaction, involving both non-VI and VI users. Although the results are still preliminary, they indicate high accuracy and good users' acceptability to guide the VI in an indoor environment.

The remainder of this paper is organized as follows: after a review of some related works in Section 2, an overview of the system architectural design is introduced in Section 3. Section 4 presents the design and implementation of the navigator module, including the proposed ACO path planning algorithm and the Path Following module. The usability evaluation is presented in Section 5, followed by a discussion of the main contributions and limitations of this work as well as possible enhancements in Section 6. Section 7 finally concludes with some thoughts for future research directions.

2. Related works

A wide range of assistive technologies are currently available to users with impairments. The first subsection introduces some systems that were proposed to assist the VI in finding their way in different environments. Then, the second subsection focuses on some inspiring ACO path planning approaches that were proposed particularly in robotics. Finally, the third subsection reviews some path following techniques, where the computed path can be updated in case of detecting deviation due to an obstacle or not properly following the instructions.

2.1. Assistive navigation for the VI

In the 1990 s, a system developed by Meijer (Meijer, 1992), namely vOICe, offered image-to-sound rendering for the blind. The prototype was composed of a digital camera connected to conventional eyeglasses, headphones, and a computer with the required application. The proposed system uses the output of the stereo-vision algorithm and splits it into five vertical sections, corresponding to the hand's fingers. A vibration alarms the user of any obstacle in that direction corresponding to that finger.

With the development of mobile phones, Tapu et al. (Tapu et al., 2013) proposed a real-time system that detects static and dynamic obstacles and alerts the partially VI user about them, while navigating using a smartphone camera in an unknown environment. The experiments showed successful results in identifying the obstacles. However, if the direction of the camera is not stable over time it will not be accurate. So, to avoid this, they recommend using a non-invasive chest mounted harness.

Recently, the Sound of Vision (SoV) project (Caraiman et al., 2019) tackled different challenging aspects of this problem such as wearability, real-time operation, pervasiveness, usability, and cost. The stereo vision-based component performs scene reconstruction and segmentation in outdoor environments. The usability was also assessed involving VI users. Moreover, Bansal et al. (Bansal et al., 2020) proposed to combine stereo vision with ultrasonic sensors to detect the obstacles in a real-time surrounding. The proposed system uses stereo vision to recognize and track objects located at a farther distance, while ultrasonic sensors were used to detect objects located at shorter distances.

As can be noticed, the development and enhancement of computer vision-based systems have been the main research subject of many works. Few studies have suggested other alternatives, such as using Radio Frequency Identification (RFID) as proposed in RoboCart (Kulyukin et al., 2005) for robot-assisted navigation to help VI people in an indoor supermarket, where localization relied on RFID tags deployed to the objects in the environment.

As for the path planning computation, the most popular algorithm is Dijkstra's algorithm. Nicolas Tissot (Tissot, 2003) proposed a prototype for an indoor navigation system for the VI. Built for Personal Digital Assistant (PDA), the system had some predefined destinations, such as the nearest information desk. The navigator module was then responsible for computing the optimal path between two locations using Dijkstra's algorithm, taking into consideration the users' preferences, and re-planning the path when the user diverts.

Al-Ammar et al. (xxxx) proposed a design for a smartphone navigation system for the blind and VI. The proposed design was based on some assumptions such as the VI can identify obstacles and avoid unsafe places using a white cane or depend on their other senses like hearing. The navigation relied on Dijkstra's algorithm in planning the path and used a path following module, responsible for following the user while navigating and detecting any wrong turn.

Hosny et al. (Hosny et al., 2015) proposed an indoor navigation system for VI people who use an electric wheelchair. The system used the A* algorithm for path planning, taking into consideration the VI needs and preferences in the objective function of the algorithm. The route features include being free of obstacles, having few turns, being close to walls, and accommodating clues and landmarks.

Recently, Li Cai and Xiao Ling Zhu (Cai and Zhu, Nov. 2018) proposed to use the traditional walking stick mounted by an infrared emission/receiving module and a speech module. The infrared technology is used to detect obstacles, and then notify the user through the speech module depending on the distance between blind people and obstacles. The authors particularly proposed ACO and tested the system through simulation. The results showed the feasibility of using ACO as a blind guide.

2.2. Path planning

For the path planning module, many approaches have been inspired from the robotics field as discussed below. A recent detailed survey about robot path planning in dynamic environments can be found in (Cai et al., 2019); where it was stated that the reactive approaches, including ACO, are rapidly growing in the field of mobile robot navigation, as they outperform classical approaches for real-time navigation problems.

Several works proposed to use ACO to tackle the path planning problem. Yee Zi Cong and S. G. Ponnambalam (Yee and Ponnambalam, 2009) proposed an ACO for finding the path of a mobile robot. Several assumptions were made such as, in a static environment with static obstacles, one ant travels at a time and moves only to one node, and there are no diagonal movements. The experiments showed that ACO outperformed the existing approaches.

Zeng Bi et al. (Zeng et al., 2009) improved the ACO to be more efficient in finding the optimal path in a dynamic environment for mobile robots. The algorithm started by acquiring information about the environment using sensors, and then a fuzzy logic was responsible for modeling the local area of the robot. A simulation experiment proved that the improved algorithm outperforms the original one in terms of computation time and the number of iterations needed for convergence.

Michael Brand et al. (Brand et al., 2010) proposed an ACO that reroutes the robot when it faces an obstacle in a dynamic environment. The main objective was to find the shortest path with no collision in a grid network, considering two pheromone re-initialization schemes. In the experiment, they achieved the same path length when comparing the two schemes, but the local initialization outperforms the global initialization in terms of the number of iterations.

Recently, Kumar et al. (Kumar et al., 2020) tackled the problem of slow convergence of ACO by improving the evaporation rate of pheromone. They proposed a fuzzified ACO (FACO) for mobile robots, where the path pheromone update mechanism considers two types of paths, favorable and unfavorable. The performance of the FACO algorithm was assessed through simulation and showed that it overcomes the slow convergence issue.

Shiguo et al. (Li et al., 2020) tackled the problem of robot path planning by taking into consideration not only the path length but also the number of turns. They used the A* algorithm to enhance the convergence speed, and turning points were considered in the pheromone concentration. The simulation results assessed the performance of the improved ACO in terms of convergence, the number of iterations, path length, and the number of turns.

Ma and Mei (Ma and Mei, 2020) proposed an improved ACO path planning for mobile robots to address the issues of slow convergence speed, trapping into local optimum, and number of turns. They used a Jump Point Search (JPS) strategy to eliminate the less promising nodes. The performance of the algorithm was assessed by both simulation and actual mobile robot navigation experiments, showing the effectiveness and efficiency of the improved algorithm in solving path planning for mobile robots.

2.3. Path following

Different methods that aim to detect any drift or deviation from the suggested path have been proposed. These methods depend on different types of technologies such as GPS, Wi-Fi, and different sensors. Since we are considering an indoor environment and offline navigation, using sensors embedded in most smartphones is a suitable solution in detecting a deviation in our case.

Vinod Pathangay (Pathangay, 2008) introduced a method to detect the deviation of the VI user from a predefined path using a mounted camera. The path following is considered in a static indoor environment, which has predefined trained frames of the path that will be compared to the current or the testing path. The experiment's results included some false-positive results, where the same path test had some false alarms of deviation.

Serra et al. (Serra et al., 2010) proposed an indoor navigation system that relied on dead reckoning localization, where the current position is estimated based on a previously known location. The system is developed on a smartphone to make use of the embedded sensors. Steps and orientation were calculated using the compass and accelerometer respectively. The experiments showed good results in detecting both the place and orientation of the user.

Link et al. (Link et al., 2011) proposed a turn-by-turn navigation system based on the accelerometer and compass embedded in the smartphones. The matching of the expected and detected steps is completed using the sequence alignment method adopted from the Bioinformatics field. In the experiment, they made a comparison between their method and the GPS in an outdoor environment to measure the accuracy. They also tested the system in an indoor environment to show its robustness. Recently, Zekany (Zékány, 2019) adopted this technique (Link et al., 2011); as part of an indoor localization system targeting the VI.

Table 1 provides a summary of some relevant studies that were discussed above and shows the main contribution of this paper compared to previous studies. It can be noted that many existing works in navigation for the VI focused on obstacle detection using stereo vision. The few works that discussed the path planning problem for the VI, proposed to use Dijkstra or A* algorithms, except (Cai and Zhu, Nov. 2018) which proposed to use ACO. On the other hand, the ACO has proven its robustness and efficiency in the field of mobile robot navigation. Moreover, as noted in (Cai et al., 2019), most existing works provided only a simulation analysis rather than real-time applications and experiments. Bringing these techniques from the robotics field and applying them in navigation systems requires further design constraints to ensure the safety of the user. This can be achieved using a path following module. Hence, this paper proposes to combine ACO path planning and the First Fit technique (Link et al., 2011) for path following to build a complete indoor navigation system for the VI. Moreover, we present the results of the usability evaluation involving real VI users, rather than only simulation analysis.

3. System overview

Since our system is developed for people with visual disabilities, a UCD approach has been adopted to take into consideration the target user's specificities in all stages of the design process. Most assistive technologies use sensors and artifacts that are not natural for VI users. However, we chose to adapt the environment to the users' needs instead of forcing the user to adapt to the system. This paper builds on the results of the needs-finding study conducted in (Hosny et al., 2015), to gather the requirements from the VI. In (Hosny et al., 2015), the authors surveyed users to collect the VI preferences while navigating. They found that most of the respondents preferred moving in a straight line, while some preferred the shortest path or walking beside the walls. Moreover, the environment features were defined, and the path features were classified as desired or undesired. This classification was used in the ACO as discussed in section 4.1.

The environment is considered to be known and static, meaning that the locations of the obstacles are predefined, and they do not change based on the movement of the VI. Fig. 1 shows the system's architecture including four main modules and each module has a set of functions.

The first module is the Administrator interface, deployed on the server-side, and responsible for map creation and storage in a database located on a remote server. The second module, the positioner, acquires the map and locates the current position of the user using beacons or other technologies. The third module is the VI user's input/output interface, which allows the user to interact with the system, specify their preferences, select the destination, or follow the directions that will guide them to the desired

Table 1
Summary of some related works.

	Ref.	Year	Contribution
Navigation for the VI	(Tapu et al., 2013)	2013	Static and dynamic obstacles detection using a smartphone camera in an unknown environment.
	(Caraiman et al., 2019)	2019	Stereo vision for the scene reconstruction and segmentation in outdoor environments.
	(Bansal et al., 2020)	2020	Obstacle detection using stereo vision with ultrasonic sensors in a real-time surrounding.
	(Tissot, 2003)	2003	PDA-based navigation using Dijkstra's algorithm, taking into consideration the user's preferences.
	(xxxx)	2011	Smartphone navigation using Dijkstra's algorithm.
	(Hosny et al., 2015)	2015	An indoor navigation wheelchair system for VI using the A* algorithm for path planning
ACO for path planning	(Cai and Zhu, Nov. 2018)	2018	Walking stick mounted by infra-red and speech modules and using ACO for path planning
	(Yee and Ponnambalam, 2009)	2009	ACO for finding the path of a mobile robot in a known environment.
	(Zeng et al., 2009)	2009	Improved ACO for finding the path in a dynamic environment.
	(Brand et al., 2010)	2010	ACO for rerouting the robot when it faces an obstacle in a dynamic environment.
	(Kumar et al., 2020)	2020	Fuzzified ACO (FACO) for a mobile robot, addressing the problem of slow convergence of ACO.
	(Li et al., 2020)	2020	Improved ACO with A* algorithm for a mobile robot, considering both the path length and the number of turns.
	(Ma and Mei, 2020)	2020	Improved ACO with Jump Point Search (JPS) for a mobile robot, considering both the path length and the number of turns.
Path following	(Pathangay, 2008)	2008	Deviation detection by comparing pre-defined trained frames of the path to the current path.
	(Serra et al., 2010)	2010	Smartphone indoor navigation system, where steps and orientation were calculated using the compass and accelerometer respectively.
	(Link et al., 2011)	2011	Steps and deviation detection using sequence alignment method to check the matching of the expected step and detected step.
	(Zékány, 2019)	2019	Steps detection based on (Link et al., 2011) as part of an indoor localization system for the VI.
This paper		2021	Indoor navigation for the VI using ACO for path planning and (Link et al., 2011) for path following + Usability evaluation involving VI users.

destination. Finally, the fourth module, the navigator, is the core of the system and the focus of this paper. The navigator is responsible for the dynamic path planning procedure, including two main functions. First, the path planning module calculates the path using the ACO algorithm, taking as input the provided map, the required destination, and the user's preferences. The module then inputs the calculated path from the path planning component, traces the path while navigating, and detects any deviation. If deviation occurs, it will notify the path planning component of that deviation, and then compute the path again. This will be repeated until the VI user reaches the desired destination.

In this system, the environment map is modeled as a grid of equal-sized cells (0.5 m each). The building has several floors, and each floor is defined as a grid and is connected to other floors through a connecting node. Besides obstacles, the map should recognize any features such as landmarks, points of interest, locations of stairs, walls, and lifts. Moreover, nodes that have sensors such as Bluetooth beacons are also defined. Cells have characteristics such as cell type, cell weight, traversable cell or not, and cell icon. This information needs to be provided by the Administrator at the initialization phase only, and then it is saved into the database of the system to be used when planning the path.

A website has been built using the C# language to facilitate the experiments and is linked to an SQL database server. The website allows the administrator to log in to the system and begin building the maps. The website has been linked to an Android application using RESTful (Representational State Transfer) web services that collect all the required data and represent them as an XML web page. Then, this page is read in the Android application and the required data is stored into an SQLite database to be used when planning the path.

The following sections discuss the design, implementation, and usability evaluation of the navigator component.

4. Design and implementation of the navigator module

4.1. Design of the path planning module

The path planning module is based on the ACO algorithm, which has been applied in solving widespread optimization problems and has achieved successful results. ACO takes the processed map and constructs the optimal, or near-optimal path, following two main steps (Talbi, 2009) as illustrated in Fig. 2.

Step 1. The solution is constructed by each ant iteratively adding solution components to a partial solution, using the probabilistic state transition rule, until getting the complete solution. An ant may be killed if it reaches a cell that has already been visited or non-traversable neighbors. The ant chooses the next cell according to the pheromone value on the edge connecting the current and next cell. Then, an ant will move from node i consecutive to node j with the probability p_{ij} defined as follows:.

$$p_{ij} = \frac{\tau_{ij}^{\alpha} \times \eta_{ij}^{\beta}}{\sum\left(\tau_{ij}^{\alpha} \times \eta_{ij}^{\beta}\right)} \tag{1}$$

where

τ_{ij} is the amount of pheromone on the edge connecting cells i and j.

α is a parameter to control the influence ofτ_{ij}.

η_{ij} is the desirability of the edge connecting cells i and j.

β is a parameter to control the influence ofη_{ij}.

The transition rule takes into consideration the following:.

- *Pheromone trail*: which saves the properties of good, generated solutions; the greater intensity of the pheromone trail means the better the path. This value is dynamically changed by associating a pheromone value τ_{ij} with each edge connecting cells i and j.
- *Heuristic information*: this variable is problem-dependent and gives clues to the ant to decide while constructing the solution. In our approach, the value of this heuristic information has been adapted to reflect the VI's preferences in terms of the desired and/or undesired features. The heuristic information η_{ij} is defined, in our ACO, as follows:

$$\eta_{ij} = \frac{1}{\left(aC_{ij} + bD_j + cC_j^{dest}\right)} \tag{2}$$

where

C_{ij} is the cost of the edge connecting cells i and j.

D_j is the desirability of the next cell j.

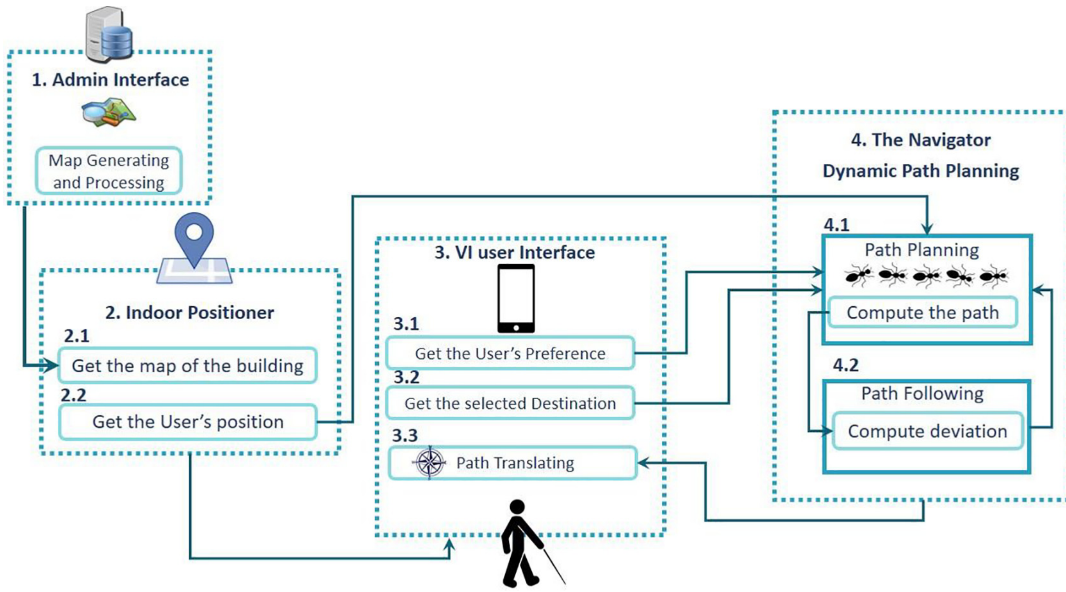

Fig. 1. Indoor navigation system architecture.

C_j^{dest} is the estimated cost to reach the desired destination starting from cell j.

a, b, and c relate to different weights that control the effect of each component.

In equation(2), the first component of the heuristic information is standard in most ACO algorithms and is intended to reduce the total travel distance to reach the destination. The second component is intended to take the preferences and the safety of the VI into consideration. More specifically, calculating the cost of the cell may vary based on the preferences of the user. The third component is inspired by the heuristic component usually used in the A* algorithm, which takes the estimated distance to reach the destination into consideration.

For the estimated cost to reach the destination, we used the Manhattan distance defined by:.

$$C_j^{dest} = |x_j - x_{dest}| + |y_j - y_{dest}| \tag{3}$$

where $\times$ and y refer to the coordinates of the cell in the grid.

Step 2. This step is named *Pheromone update* and is conducted in two phases:.

- *Evaporation phase*: in this phase, the pheromone is reduced by a certain proportion to avoid premature convergence, as given by the following function, where ρ is the pheromone evaporation rate:

$$\tau_{ij} = (1-\rho)\tau_{ij} \forall i,j \in [1,n] | \rho \in]0,1] \tag{4}$$

- *Reinforcement phase*: in this phase, the pheromone is updated based on the solution generated. Each ant will increment the pheromone associated with selected edges that belong to the best path, with a value that is proportional to the quality of the best-obtained pathπ. The amount of the pheromone on the resulting path will increase by a certain amount according to the following equation:

$$\tau_{ij} = \tau_{ij} + \Delta$$

where (i,j) is an edge belonging to the best pathπ, and $\Delta = \frac{1}{f(\pi)}$ where $f(\pi)$ is the objective function.

The objective function of a path π is defined by:.

$$f(\pi) = \left(\omega_1 \sum_{(i,j)\in\pi} C_{ij}\right) + \left(\omega_2 \sum_{j\in\pi} D_j\right) \tag{5}$$

and it represents the total cost of the path obtained by each ant. This cost is calculated based on two criteria: 1)C_{ij}, the cost of moving from cell i to cell j in the pathπ, and 2)D_j, the desirability of cell j in the pathπ, calculated by summing the desired and undesired costs of the features in the cell. ω_1 and ω_2 are weights in the range [0,1] that control the relative importance of these two criteria, such that ω_1+ω_1 = 1.0.

The algorithm works in a grid environment. The grid cells are numbered from 0 to N. To move from one cell to another, eight possible moves are allowed (horizontal, vertical, and diagonal). The cell should not be visited more than once and the start and endpoints should be specified before starting the algorithm. When the destination cell is selected, for each adjacent cell of the starting point, the heuristic information will be calculated using equation (2) as follows:.

The cost C_{ij} is equal to 10 for vertical or horizontal cells, and 14 for diagonal cells (Zhang et al., 2012).

To calculate the desirabilityD_j, values are assigned for the desired and the undesired features in each cell. This is done

Initialize the pheromone information;
Repeat
  For each Ant **Do**
    // Solution construction using pheromone trails
    S = {1, 2,..., n}; // Set of potential selected cells
    Random selection of the initial cell i;
    Repeat
      Select new cell j with the probability $p_{ij} = \frac{\tau_{ij}^{\alpha} \times \eta_{ij}^{\beta}}{\sum(\tau_{ij}^{\alpha} \times \eta_{ij}^{\beta})}$;
      S = S – {j}; i=j;
    Until S=∅
  End For
  For $i,j \in [1,n]$ **Do**
    $\tau_{ij} = (1-\rho)\,\tau_{ij}$ //Evaporation
  For $i \in [1,n]$ **Do**
    $\tau_{ij} = \tau_{i\pi} + \Delta$ // π is the best-found solution
Until no improvement for a given number of con iterations.

Fig. 2. Pseudo-code for the proposed ACO algorithm.

by giving the desired features an award (a negative penalty) to decrease their cost and punishing the undesired cells with a positive penalty, to increase their cost. When the map designer specifies the cell content, the weight is increased or decreased by 5 according to the cell desirability as shown in Table 2.

The last component is the estimated cost to the destination C_j^{dest} which is calculated using the Manhattan distance as defined in equation(3).

After initializing the variables, ants should be deployed, where each ant will construct a possible solution. Talbi (Talbi, 2009) recommends beginning with a large number of ants. If the ant finds a possible move, where there are no obstacles and the node has not been visited yet, the ant will proceed; otherwise, it will be killed, and another ant will be deployed. For each cell that the ant can move to, the transition probability will be computed and the cell with the highest probability is selected. Then the heuristic information and other relevant variables are updated.

The previous steps are repeated until the final goal is reached. When the goal is reached, the evaporation of the pheromone is conducted for all nodes, while reinforcement takes place for nodes belonging to the best path in terms of the overall cost.

4.2. Implementation and testing of the path planning module

The ACO was first developed on eclipse IDE using Java programming language to test it and tune the parameters. It was then deployed on the Android application after fixing the required parameters. These parameters must be adjusted since they can affect the overall performance of the algorithm. The ACO parameters were tuned one by one. Initially, some recommended values were assigned to each parameter; then they were increased or decreased gradually. For each value, the results of several different runs were recorded and then the best parameters values were chosen. Two different designs of the map were considered, a 20x20 and a 30x40 grid, with walls, doors, obstacles, points of interest, and landmarks. Then, the ACO algorithm was run with different values of parameters. For α, β and ρ, further to the values recommended in (Zhao and Zu, 2009); $\alpha = 1.8$, $\beta = 1.5$ and $\rho = 0.2$, we considered four more values. For the number of ants, we considered random values starting from 10 to 40. Moreover, we did not specify a fixed number of iterations to stop the algorithm; instead, we made it dynamic depending on the number of iterations without improvement. The last two parameters that were also tested are the weights used in the objective function (Equation (5)) ω_1 and ω_2. The performance of the ACO was evaluated in terms of the objective function as well as the processing time. Table 3 summarizes the selected parameters based on the parameters tuning process.

Then the algorithm was tested, using these parameters, on different maps (8x8, 20x20, 30x40, and 80x80), with 5 runs each. The results of running the algorithm on the different maps are shown in Table 4, where it is clear that the resulting objective function is stable. Except for the second map, all standard deviations of the objective function among the 5 runs were zero. Also, there is almost no fluctuation in the lengths of the computed paths. Finally, the processing time was less than one second for the relatively small to medium maps, while it was about 3 s for the biggest map (80x80).

Moreover, it turned out that the resulting path satisfies the path constraints and achieves the objective function in terms of selecting desired cells, avoiding obstacles, and walking near walls and landmarks. It can be observed in the portion of the path depicted in Fig. 3, the computed path is not necessarily the shortest, but one satisfying the preferences of the user, such as walking near the wall, next to some points of interest, and avoiding the obstacles and leaving some separating distance to them. Therefore, it seems that the objective function properly guides the algorithm in the search space and achieves its intended purpose, without colliding with obstacles.

Table 2
Desirability classification.

Desired	Undesired
1. Shortest path (total distance)	1. Obstacles (not traversable)
2. Straight-line path	2. Near obstacles (+5)
3. Near Landmark (-5)	3. Stairs/ Escalators (not traversable)
4. Near Point of Interest (POI) (-5)	4. Near stairs/escalator (+5)

4.3. Implementation of the path following module

The path following module is responsible for checking if the VI user is following the suggested path. A key challenge is to ensure high performance using the minimum number of sensors to provide more convenience for the users. Nowadays, most smartphones are embedded with sensors that can be used for navigation. The accelerometer is used to detect the user's steps, while the compass is used to identify the direction.

4.3.1. Step detection

Despite the great efforts and the number of studies conducted in walking detection and step counting, it is still challenging to ensure its accuracy and efficiency with smartphones (Kang et al., 2018). In this paper, a simple technique is implemented using only the accelerometer sensor that is embedded in most smartphones, which makes it easily accessible for step detection. It is based on the sliding window technique proposed by (Link et al., 2011) to segment the accelerometer data. This technique has been recently adopted in (Zékány, 2019) for step detection as part of an indoor localization system targeting VI people.

In (Link et al., 2011), a window (W) of five consecutive sampling points is defined, and an overlapping of 80% is utilized. If there is a difference between two consecutive points that are higher than the threshold (p), then a step is detected. Then, a timeout (t) is set to reflect the time elapsed between two steps to avoid detecting a false step. Moreover, a low pass filter (f) is set to filter the gravity noise.

Three parameters were considered for detecting the human steps from the accelerometer data. Since each user has a different walking style, we designed an interface with changeable parameters for the experiments, as shown in Fig. 4, so that the parameters can be adjusted to the user's walking style. It should be noticed that if the user changes the walking speed, the values should be modified so that the system can properly detect the steps. A future enhancement could be to implement an adaptive step detector that automatically detects the change of the walking speed or style.

4.3.2. Deviation detection

The path following module should receive the planned path as a sequence of cells and directions from the path planning module, and then verify whether the user is on the right path. The compass is used here to detect the direction of the user, by comparing the received degree with certain angles. The implemented algorithm is based on the First Fit technique (Link et al., 2011), where two methods for matching the path are considered. There are two modes: 1) a *direct matching mode*, when the detected step heading is matching the direction of the expected edge and above a certain threshold, and 2) a *lookahead matching mode* that will be triggered if there are differences when comparing the expected direction of the edge and the detected step heading in five consecutive steps.

Table 3
The ACO algorithm's fixed parameters.

α	β	ρ	Number of ants	Iteration stopping condition	ω_1	ω_2	a	b	c
0.5	0.5	0.2	35	20	0.5	0.5	0.2	0.2	0.6

Table 4
Results of running the ACO algorithm on different maps.

	Objective Function		Path length (m)		Computation time (s)	
Map Size	Avg.	Std. Deviation	Avg.	Std. Deviation	Avg.	Std. Deviation
8x8	72.5	0	2.6	0.22	0.3	0.14
20x20	272	13.39	15.2	0.76	0.9	0.14
30x40	457.5	0	21.1	0.82	1.1	0.09
80x80	282.5	0	12	0	3.2	0.34

The lookahead matching mode will look further for the matching edge direction on the expected route and map the detected steps to it. The implemented algorithm tolerates three incorrect steps then starts the lookahead matching mode, where it will start matching the last three consecutive detected steps with three coming expected steps.

5. Usability evaluation

The usability of the proposed system was evaluated using non-VI users as well as VI users. The evaluation aimed to assess mainly the path following module as its performance closely depends on the user's behavior.

5.1. Usability criteria

The (ISO 9241-11: 2018) ([30]) identifies efficiency, effectiveness, and satisfaction as major attributes of usability. For the proposed system, these three criteria can be defined as follows:.

- **Effectiveness** aims to answer the following question: "Can users' complete tasks and achieve goals with the system?" This can be evaluated by assessing the accuracy of the system in detecting the steps and deviation as follows:

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

where True Negative (TN) corresponds to no step detected when there is no actual step, True Positive (TP) corresponds to a correctly detected step, False Negative (FN) corresponds to no step detected while there is a step, and False Positive (FP) corresponds to a wrongly detected step.

- **Efficiency** aims to answer the following question: "How much effort do users require to do this?" This can be evaluated by measuring the time to task completion.
- **Satisfaction** aims to answer the following question: "What do users think about the product's ease of use?" The System Usability Scale (SUS) is an industry standard that has been used here to assess the comfort and acceptability of system use. It consists of ten questions with five response options ranging from "Strongly agree" to "Strongly disagree".

5.2. Participant's characteristics

The system was tested with 17 participants in total, 10 non-VI volunteers, and 7 VI participants. The ages varied between 18 and 37 years old. Almost 50% of the participants have previous experience in using navigation systems outdoors. However, all

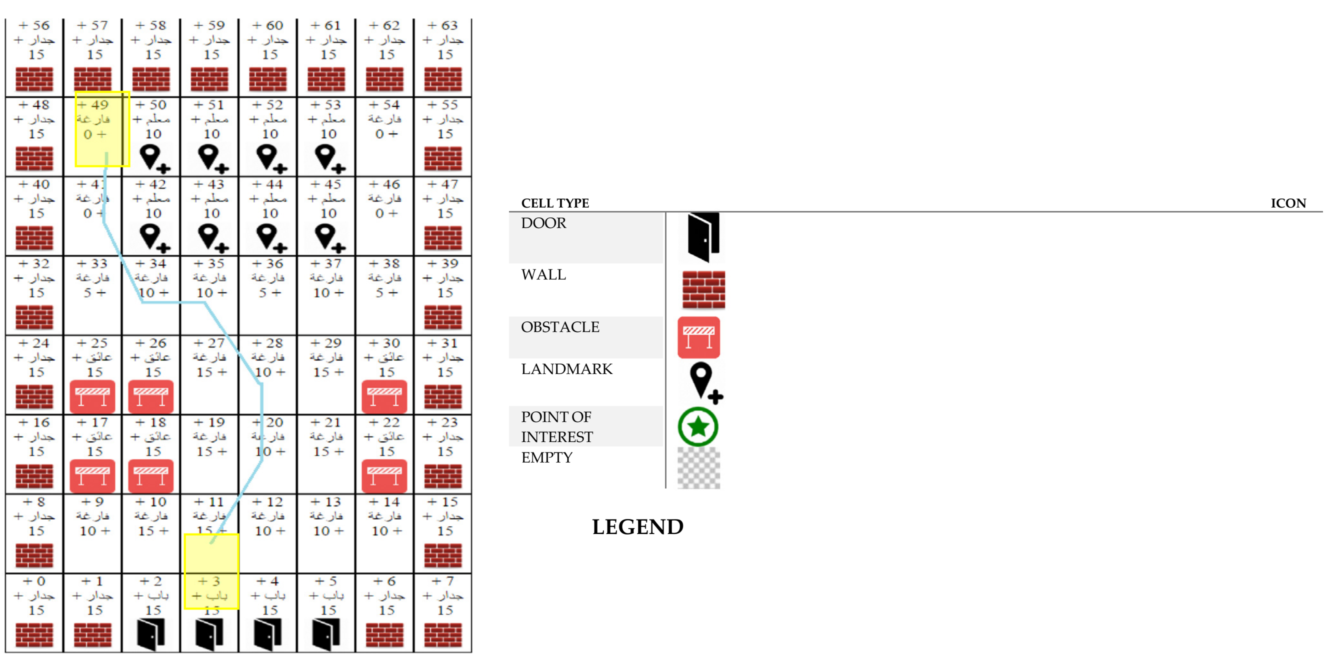

Fig. 3. Example of a computed path using ACO.

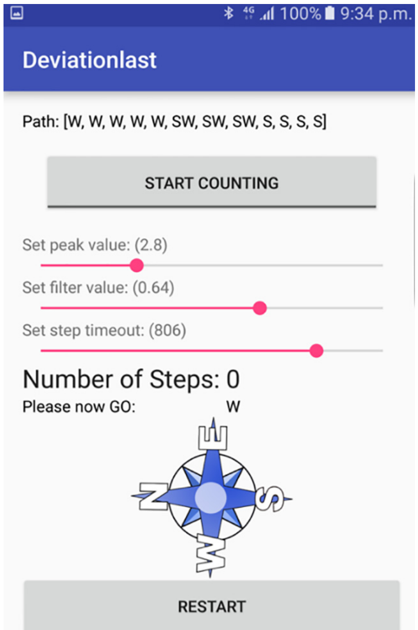

Fig. 4. Screenshot of the step detection interface.

the VI participants have no prior experience, except through participating in testing some prototypes under development.

The values of (p) and (t) depend on the user's walking style. It was noticed that male participants walked with an average value of (p) equal to 2.6 m/s which is slightly higher than female participants' with (p) equal 2 m/s.

As for the vision status, Table 5 shows the average values for height-matching 4 normal and 4 VI participants of the same gender (female) and the same age range (20 to 32 years old). The test found that the VI participants walked almost with the same velocity but with slightly higher frequency than the participants with normal vision. In fact, (Clark-Carter et al., 1986) stated that blind pedestrians would walk at a speed close to that of sighted pedestrians when accompanied by a sighted guide. However, if they walk independently, they will slow down the pace compared to their preferred walking speed.

5.3. Testing procedure

The experiments were conducted using an Android device Galaxy S6 edge model number: SM-G925I with 32 GB as total space. The testing was conducted in three phases:.

1. Pilot testing with 10 non-VI users, primarily aiming to assess the accuracy of the system before conducting the testing with the target VI users.
2. Pilot testing with three VI users who had experience in testing different navigation systems, such as Tango. This pilot testing aimed to point out any problem with the test instructions and formatting.
3. Final Testing with four VI users who had no experience with any navigation system.

Table 5
Results for height-matching normal and VI participants.

	Normal vision	VI
Avg height *cm*	157.75	157.75
Avg acceleration threshold (p) m/s^2	2.0	2.1
Avg time between steps (t) *ms*	356	307

All the testing started by first adjusting the suitable peak threshold (p) and time (t) for each user by changing these two values and asking the user to walk for a certain number of steps. When the system correctly detects the exact walked steps three times, the threshold was fixed. Then, the participants were asked to perform two tasks as shown in Table 6.

After completing the required tasks, the VI participants were requested to fill in the System Usability Scale form to assess the user's satisfaction.

5.4. Results of testing with non-VI users

The testing with non-VI users primarily aims to assess the accuracy of the system before conducting the testing with the target VI users. After adjusting the threshold, the participants were asked to perform Task 1. As shown in Fig. 6, the lowest accuracy was 90%. The American College of Sports Medicine (Schneider, 2011), considers step detection accurate if it reads between 18 and 22 steps of 20 exact steps. This is equal to 90% in terms of accuracy. Moreover, the average accuracy for each participant was higher than 96%, with an excellent overall average accuracy equal to 97.65%.

Then, they were asked to perform Task 2 by following the directions as displayed on the screen. The experimental testing results showed that any deviation from the planned path was correctly detected by the system in all the cases for all the users, which means an accuracy equal to 100% for deviation detection.

Most of the users did follow the path, and out of ten users, four users deviated, requiring the path to be re-planned. A difference between the actual and detected steps, while following the suggested path, was also noticed because the users' walking style differed compared to Task 1. The users became more careful and tried to carefully follow the directions. No tones were informing the normal user of the successful step nor directions as voice-over; this was added when testing with the VI users, as explained next.

5.5. Results of testing with VI users

After assessing the accuracy of the system through testing with non-VI users, this section reports the results of the usability testing with VI users, where the usability is assessed in terms of effectiveness, efficiency, and user satisfaction, in line with the international standard ISO 9241–11. This testing was conducted in two phases: pilot testing and final testing. Hence, two groups participated on two different days in this usability evaluation. The VI users were female adults with different educational backgrounds. The first group had experience in testing different navigation systems, while the second group had no experience with navigation systems at all.

5.5.1. Effectiveness

The testing started by calibrating the suitable thresholds and then performing Task 1. Fig. 7 and Table 7 show the average accuracy of the step detection. In the final testing with the second group, we added a beep when there is a successfully detected step,

Table 6
Tasks performed in the usability evaluation.

	Normal users (10 participants)	VI users (7 participants)
Usability factors	Effectiveness in terms of accuracy and task completion	Effectiveness, efficiency, and satisfaction
Task 1. Steps detection	walk 20 steps × 10 times	walk 8 steps × 6 times
Task 2. Deviation detection	follow the path as the directions appear on the screen	follow the path as the directions are voiced over

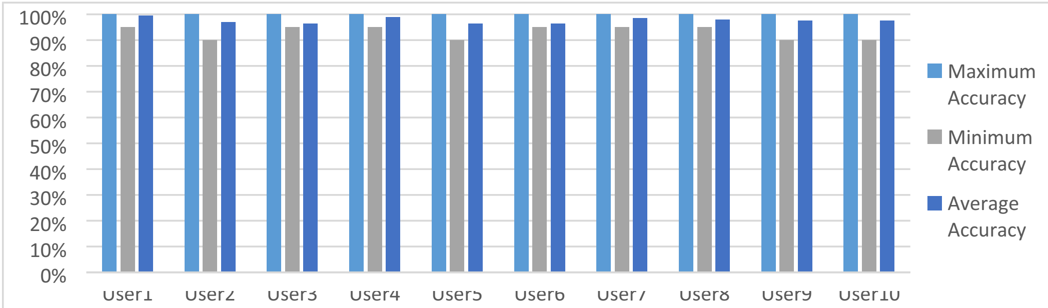

Fig. 6. Accuracy results of step detection for normal users.

as suggested by the participants in the pilot testing, which considerably improved the results of the second group.

As shown in Fig. 7, the lowest accuracy was 41% that was recorded for the first user, user1, from the first group (pilot testing), since the step detection beep was not added yet. For the second testing group (from users 4 to 7) the lowest accuracy is equal to 66%. The accuracy has been considerably improved but still not sufficient according to (Schneider, 2011). It can be noted that, for four users out of seven, a maximum accuracy equal to 100% was reached, which suggests that steps detection can be particularly accurate when users have an almost stable walking style. Table 7 shows the overall average accuracy recorded for each group, and it can be noted that the second group had a better accuracy due to the added beep for any step detected.

The participants were then asked to perform Task 2 by path following the directions from a voice-over. Most of the VI users were able to accomplish the task and the system was able to detect any deviation occurring. Like the non-VI user testing results, the experimental testing results for the VI show that any deviation from the planned path was correctly detected by the system in all the cases for all the users, which means the accuracy of 100% in deviation detection.

5.5.2. Efficiency

As for the efficiency, the time to complete Task 2 was recorded. Using the average step length for women, which approximately equals 66 cm (Thompson, 2002). Table 8 shows the walking speed of each participant for completing the task successfully. According to (Guerrero et al., 2012), the walking speed of a blind person without navigation support was 0.4 m/s. So, the VI participants in our experiments walked slower than they are supposed to do, probably because they were following the instructions given by the navigator. As shown in Table 8, the overall average walking speed was 61% of the average walking speed of a blind person without any navigation support. It can be noticed that the first VI participant was able to reach 90% of the usual walking speed, knowing that she previously participated in testing a navigation system using the Google Project Tango to detect obstacles (Jafri et al., 2017).

5.5.3. User satisfaction

Finally, the VI participants were asked to fill in the System's Usability Scale form to assess the user's satisfaction. Fig. 8 shows the VI users' satisfaction results as a score out of 100 for each tested group. The lowest score was 80%, which is considered good and above average, according to (Zaman, 2015) that states that a score above 68% is considered above average. The lowest score in the second group improved and reached 87.5%. These results indicate that the VI users were satisfied and willing to use the system.

6. Discussion and limitations

In summary, the contribution of this research work is threefold:.

1. An ACO path planning algorithm that can find an optimized route, taking into consideration the VI user's preferences, and avoids collisions with obstacles. The human factors, such as the special needs of the VI user have been incorporated in each step of the algorithm as well as in the objective function.
2. The proposed path planner is integrated with a path following module using the First Fit technique that detects any deviation from the planned path and alerts the VI users, to safely lead them to the desired destination.
3. A thorough usability evaluation involving both normal-vision and VI users was conducted to assess the usability of the proposed system in terms of effectiveness, efficiency, and satisfaction.

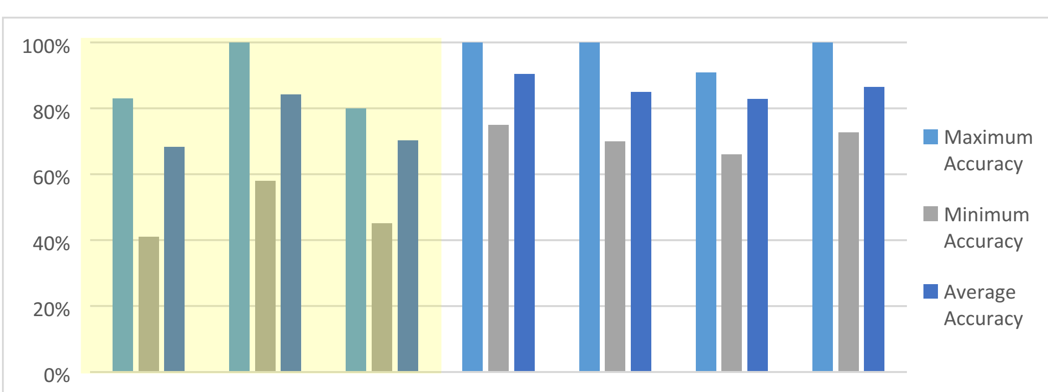

Fig. 7. Accuracy results of step detection for the VI.

Table 7
Average accuracy for VI users.

VI User	Pilot testing with the VI 1	2	3	Final testing with the VI 4	5	6	7
Average Accuracy	68.20%	84.20%	70.20%	90.50%	84.90%	82.80%	86.50%
Overall Average Accuracy	74.2%			86.175%			

Table 8
Walking speed of the VI participants to complete the path following task.

User	Average walking speed to complete the task successfully (m/s)	Percentage compared to walking speed without navigation support (0.4 m/s)
VIUser1	0.36	90%
VIUser2	0.22	55%
VIUser3	0.19	47%
VIUser4	0.22	56%
VIUser5	0.25	62%
VIUser6	0.22	54%
VIUser7	0.26	66%
Overall Average	0.24	61%

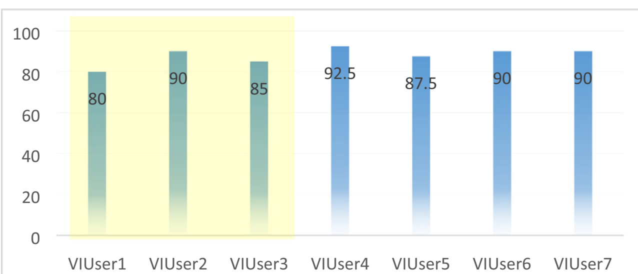

Fig. 8. VI user satisfaction results.

However, independent indoor navigation for VI people is extremely challenging, especially in unfamiliar environments, and further research studies can be conducted to improve the performance. Some of the possible enhancements are suggested below:.

1. ACO proved its robustness and efficiency in solving the path planning problem; however further design issues can still be addressed to reduce the convergence speed and computational time, improve the necessary learning phase, and deal with trapping into local optimum issues. The techniques proposed in (Li et al., 2020; Ma and Mei, 2020) can be considered to improve the convergence speed.
2. Other newly proposed metaheuristics, such as Aquila Optimization (Abualigah et al., 2021) and Arithmetic Optimization (Abualigah et al., 2021) could be applied to solve this problem. It would be interesting to investigate their performance compared to ACO and other path planning algorithms.
3. The study found that if the user changes the walking speed, the path following module should be reconfigured to properly detect the steps. An adaptive step detector can solve this issue by automatically detecting the change of the walking speed or style.
4. This paper focused on the design of the navigator component; however, an accurate positioning system and a reliable obstacle detector are also needed to build a fully functional navigating system. Moreover, a more sophisticated interface that fits the special needs of the VI, is also needed to enhance the usability of the system.
5. As for the usability evaluation, more experiments involving a larger number of participants in different environments are necessary to validate the usability of the system and ensure a high-quality experience for the VI user.

7. Conclusion

This paper presented a path planning algorithm, based on ACO designed with a specific objective function that is tailored to the needs of the VI users, and where the environmental features are considered during the path planning process. The algorithm was tested on different maps with different structures. The results in general indicate that the computed paths reflect the preferences of the VI. Moreover, a path following algorithm was developed to ensure the safety of the VI. The usability of the proposed system was assessed in terms of efficiency, effectiveness, and user satisfaction involving both non-VI and VI users. The results show a good level of accuracy along with a high level of user satisfaction.

Future work includes improving the path planning and path following modules and integrating them with an accurate positioning system, a reliable obstacle detector as well as a more sophisticated interface to build a fully functional indoor navigation system for the VI. Furthermore, this study can be extended by conducting more experiments involving a larger number of participants in different environments to have a comprehensive usability evaluation. Future research might also study the long-term use of the system to investigate its usability after a period of training.

Author Contributions

All authors have read and agreed to the published version of the manuscript.

Funding

This research project was supported by a grant from the "Research Center of College of Computer and Information Sciences", Deanship of Scientific Research, King Saud University.

The authors thank the Deanship of Scientific Research and RSSU at King Saud University for their technical support.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research project was supported by a grant from the "Research Center of College of Computer and Information Sciences", Deanship of Scientific Research, King Saud University.

References

World Health Organization, "World report on vision," 2019. https://www.who.int/publications-detail/world-report-on-vision.

E. Alqahtani, F. Alshamrani, H. Syed, and F. Al-Haidari, "Survey on Algorithms and Techniques for Indoor Navigation Systems.," in *21st Saudi Computer Society National Computer Conference (NCC), Riyadh, Saudi Arabia*, 2018, pp. 1–9, 10.1109/NCG.2018.8593096.

Hosny, M., Alsarrani, R., Najjar, A., 2015. Indoor Wheelchair Navigation for the Visually Impaired. Springer, Cham, pp. 411–417.

W. Sakpere, M. Adeyeye-Oshin, and N. B. W. Mlitwa, "A state-of-the-art survey of indoor positioning and navigation systems and technologies," South African Computer Journal, vol. 29, no. 6, pp. 145–197, 2017, 10.18489/sacj.v29i3.452.

Veeranjaneyulu, T.M.P.N., Baseer, K.K., Asha, V.S., 2019. A Systematic Literature Review on Obstacle Detection for Visually Impaired People. International Journal of Computational Science and Engineering 7 (6), 56–65.

Patle, B.K., Babu, G.L., Pandey, A., Parhi, D.R.K., Jagadeesh, A., 2019. A review: On path planning strategies for navigation of mobile robot. Defence Technology 15 (4), 582–606. https://doi.org/10.1016/j.dt.2019.04.011.

P. B. L. Meijer, "An Experimental System for Auditory Image Representations," IEEE Transactions on Biomedical Engineering, vol. 39, no. 2, 1992, 10.1109/10.121642.

R. Tapu, B. Mocanu, and T. Zaharia, "A computer vision system that ensure the autonomous navigation of blind people," *E-Health and Bioengineering Conference (EHB), Iasi, Romania,* pp. 1-4, 2013, 10.1109/EHB.2013.6707267.

S. Caraiman, O. Zvoristeanu, A. Burlacu, and P. Herghelegiu, "Stereo vision based sensory substitution for the visually impaired," *Sensors (Switzerland)*, vol. 19, no. 12, 2019, 10.3390/s19122771.

V. Bansal, K. Balasubramanian, and P. Natarajan, "Obstacle avoidance using stereo vision and depth maps for visual aid devices," *SN Applied Sciences*, vol. 2, no. 6, 2020, 10.1007/s42452-020-2815-z.

V. Kulyukin, C. Gharpure, and J. Nicholson, "RoboCart: Toward robot-assisted navigation of grocery stores by the visually impaired," *IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada,* pp. 2845-2850, 2005, 10.1109/IROS.2005.1545107.

N. Tissot, "Indoor navigation for visually impaired people-The navigation layer," *Technical Report, Swiss Federal Institute of Technology EPF Lausanne*, 2003 [Online]. Available: www. mics. ch/SumIntU03/NTissot. pdf.

M. A. Al-Ammar, H. S. Al-Khalifa, and A. S. Al-Salman, "A proposed indoor navigation system for blind individuals," *in Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services, Ho Chi Minh City, Vietnam*, pp. 527–530, 10.1145/2095536.2095647.

Cai, L., Zhu, X., Nov. 2018. "Intelligent Guide Cane design Based on Ant Colony Algorithm", *IOP Conference Series: Materials Science and Engineering, Nanchang*. China. https://doi.org/10.1088/1757-899x/423/1/012066.

K. Cai, C. Wang, J. Cheng, S. Song, C. W. de Silva, and M. Q. H. Meng, "Mobile robot path planning in dynamic environments: A survey," *Instrumentation,* 2019, vol. 6, no. 2, pp. 90-100, 10.15878/j.cnki.instrumentation.2019.02.010.

Yee, Z.C., Ponnambalam, S.G., 2009. Mobile robot path planning using ant colony optimization. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 851–856. https://doi.org/10.1109/AIM.2009.5229903.

B. Zeng, Y. Yang, and Y. Xu, "Mobile robot navigation in unknown dynamic environment based on ant colony algorithm," in *Proceedings of the 2009 WRI Global Congress on Intelligent Systems, GCIS 2009*, 2009, vol. 3, 10.1109/GCIS.2009.274.

M. Brand, M. Masuda, N. Wehner, and X. H. Yu, "Ant colony optimization algorithm for robot path planning," in *2010 International Conference on Computer Design and Applications, ICCDA 2010*, 2010, vol. 3, 10.1109/ICCDA.2010.5541300.

Kumar, S., Pandey, K.K., Muni, M.K., Parhi, D.R., 2020. Path Planning of the Mobile Robot Using Fuzzified Advanced Ant Colony Optimization. Lecture Notes in Mechanical Engineering. Springer, Singapore. https://doi.org/10.1007/978-981-15-2696-1_101.

S. Li, W. Su, R. Huang, and S. Zhang, "Mobile robot navigation algorithm based on ant colony algorithm with A∗ heuristic method," *4th International Conference on Robotics and Automation Sciences (ICRAS), Wuhan, China,* pp. 28-33, 2020, 10.1109/ICRAS49812.2020.9135055.

X. Ma and H. Mei, "The Global Path Planning of Ant Colony System Mobile Robot Based on Jump Point Search Strategy," *Jiqiren/Robot*, vol. 42, no. 4, pp. 494–502, Jul. 2020, 10.13973/j.cnki.robot.190463.

Pathangay, V., 2008. Detecting deviations in visual path following for indoor environments. In: TENCON 2008–2008 IEEE Region 10 Conference, pp. 1–5. https://doi.org/10.1109/TENCON.2008.4766554.

Serra, A., Carboni, D., Marotto, V., 2010. "Indoor pedestrian navigation system using a modern smartphone", *MobileHCI '10*. In: Proceedings of the 12th international conference on Human computer interaction with mobile devices and services, pp. 397–398. https://doi.org/10.1145/1851600.1851683.

J. Á. B. Link, P. Smith, N. Viol, and K. Wehrle, "FootPath: Accurate map-based indoor navigation using smartphones," *International Conference on Indoor Positioning and Indoor Navigation, Guimaraes, Portugal*, pp. 1-8, 2011, 10.1109/IPIN.2011.6071934.

S. Zékány, "Step-Detection Algorithm for Indoor Localization System Using Mobile Phone Sensors," *microCAD International Multidisciplinary Scientific Conference, University of Miskolc*, 23-24 May, 2019, 10.26649/musci.2019.035.

E. G. Talbi, "Metaheuristics: From Design to Implementation," *Book published by Wiley, ISBN: 978-0-470-27858-1*, 2009.

Zhang, L., Min, H., Wei, H., Huang, H., 2012. "Global path planning for mobile robot based on A∗ algorithm and genetic algorithm", in. IEEE International Conference on Robotics and Biomimetics (ROBIO) 2012, 1795–1799. https://doi.org/10.1109/ROBIO.2012.6491228.

Y. Zhao and W. Zu, "Real-time obstacle avoidance method for mobile robots based on a modified particle swarm optimization," in *Proceedings of the 2009 International Joint Conference on Computational Sciences and Optimization, CSO2009*, 2009, vol. 2, 10.1109/CSO.2009.196.

Kang, X., Huang, B., Qi, G., 2018. A novel walking detection and step counting algorithm using unconstrained smartphones. Sensors (Switzerland) 18 (1), 297. https://doi.org/10.3390/s18010297.

. ISO 9241–112018(E), no. November.

D. D. Clark-Carter, A. D. Heyes, and C. I. Howarth, "The efficiency and walking speed of visually impaired people," *Ergonomics*, vol. 29, no. 6, 1986, 10.1080/00140138608968314.

Schneider, P., 2011. Selecting and effectively-using a pedometer. ACSM's Consumer Information Committee, Brochure reprinted with permission of the American College of Sports Medicine.

Dave Thompson, "Stride Analysis," University of Oklahoma Health Sciences Center, 2002, [online] https://ouhsc.edu/bserdac/dthompso/web/gait/knmatics/stride.htm (accessed Jul. 09, 2020).

Guerrero, L.A., Vasquez, F., Ochoa, S.F., 2012. An indoor navigation system for the visually impaired. Sensors (Switzerland) 12 (6), 8236–8258. https://doi.org/10.3390/s120608236.

Jafri, R., Campos, R.L., Ali, S.A., Arabnia, H.R., 2017. Visual and Infrared Sensor Data-Based Obstacle Detection for the Visually Impaired Using the Google Project Tango Tablet Development Kit and the Unity Engine. IEEE Access 6, 443–454. https://doi.org/10.1109/ACCESS.2017.2766579.

Zaman, H.B. et al., 2015. "Advances in Visual Informatics", *4th International Visual Informatics Conference, IVIC 2015 Bangi, Malaysia*. Lecture Notes in Computer Science 9429.

L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, "Aquila Optimizer: A novel meta-heuristic optimization algorithm," Computers & Industrial Engineering, vol. 157, 2021, 10.1016/j.cie.2021.107250.

L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The Arithmetic Optimization Algorithm," *Computer Methods in Applied Mechanics and Engineering,* vol. 376, 2021, 10.1016/j.cma.2020.113609.