RFID Based Indoor Navigation with Obstacle Detection Based on A* Algorithm for the Visually Impaired

Jayron Sanchez, Analyn Yumang, and Felicito Caluyo

***Abstract*—The visually impaired individual may use a cane, guide dog or ask for assistance from a person. This study implemented the RFID technology which consists of a low-cost RFID reader and passive RFID tag cards. The passive RFID tag cards served as checkpoints for the visually impaired. The visually impaired was guided through audio output from the system while traversing the path. The study implemented an ultrasonic sensor in detecting static obstacles. The system generated an alternate path based on A* algorithm to avoid the obstacles. Alternate paths were also generated in case the visually impaired traversed outside the intended path to the destination. A* algorithm generated the shortest path to the destination by calculating total cost of movement. The algorithm then selected the smallest movement cost as a successor to the current tag card. Several trials were conducted to determine the effect of obstacles in the time traversal of the visually impaired. Dependent-sample t-test was applied for the statistical analysis of the study. Based on the analysis, the obstacles along the path generated delays while requesting for the alternate path because of the delay in transmission from the laptop to the device via ZigBee modules.**

***Index Terms*—A* algorithm, RFID technology, ultrasonic Sensor, ZigBee module.**

I. Introduction

A visually impaired person needs a guide to be able to navigate an environment. The visually impaired person may use a cane, guide dog or ask for assistance from a person for guidance. However, the mobility and autonomy of the visually impaired person may be limited especially when he/she navigates an unfamiliar environment. The visually impaired person is usually challenged when moving from one place to another. Sounds or echoes are the guide of the visually impaired person to determine obstacles and pathways to analyze an environment [1].

Radio frequency identification (RFID) has been an emerging technology. RFID technology can be implemented for navigation, especially for the blind.

Previous navigation systems contain a navigation server, a device, and a track infrastructure [2]. The system allows the visually impaired to navigate the intended environment guided by instructions from a headphone. The system provides the user with the shortest path to reach the desired destination using a server. Some systems considered the use of tactile signals for giving information rather than acoustic signals. The environmental detection range for obstacles will increase with the use of the ultrasonic ranging system [3]. The major components of the system for navigation of the visually impaired are the following: RFID reader, tags, method of giving information and algorithm for path finding. The study adapts to the intended environment with obstacles. The information is transmitted wirelessly via ZigBee modules.

ZigBee transceivers were implemented previously for transmitting tag information [4]. The algorithm for the proposed system by the author is based on A* algorithm.

Visually impaired individuals experience difficulties compared to a person who is considered to be in good shape [4]. The proposed system helps the visually impaired person to be able to navigate indoor environments. A shortest path algorithm is implemented to be able to minimize the time of traversal from the initial position to the destination. An obstacle detection sensor is also implemented in the system to expand the detection range of the blind to avoid obstacles.

II. Methodology

This part of the study shows the necessary steps to be accomplished in order to meet the objectives of the study.

The conceptual diagram shown in Fig. 1 for the visually impaired is divided into different components. The first component of the system is the cane. The cane consists of the following: microcontroller, RFID reader, obstacle detection sensor, ZigBee module and the mp3 decoder for the headset. The appropriate location and installation of RFID tags in the intended environment is specified first by the author. Each RFID tag has a unique ID. There are two types of RFID tags: active and passive. The type of RFID tags that is implemented in the study are passive tags. Passive tags don't require power source and it cost less compared to active tags. The RFID tags are mounted on the ground in such a manner that the tags are equidistant from each other, with a distance of 0.682 m from each other, to avoid collision from the reader. Previous studies implemented RFID tags as checkpoints [5]. The destination is selected first by the person using the laptop. After selecting the destination, the user is guided to the next RFID tag by the system via headset. The user will receive instructions by hearing the basic commands for direction (forward, turn left, and turn right.) from the headset. The algorithm is implemented in the program installed in the laptop. The algorithm is applied to determine the shortest path to the selected destination. The shortest path generated by the algorithm is transmitted to the microcontroller unit using ZigBee module. ZigBee modules can communicate within 150 meters indoor range. Each time the RFID reader detects a RFID tag, the microcontroller will verify the unique ID. The ID will be checked by the microcontroller if the user is still

Manuscript received November 24, 2014; revised February 10, 2015.

The authors are with the Computer Engineering (CpE) Department at Mapua Institute of Technology, Manila, Philippines (e-mail: jayronsanchez@gmail.com).

DOI: 10.7763/IJIEE.2015.V5.572

along the path. If not, the software will generate an alternative path, using the same algorithm, for the user to traverse. In case the sensor detected an obstacle before the tag, the microcontroller will send a signal to the laptop that will request an alternative path generated by the A* algorithm. The new path for the user is transmitted to the microcontroller via ZigBee. The system will notify the user via headset when he/she reached the destination. The system learns and maps the obstacles detected.

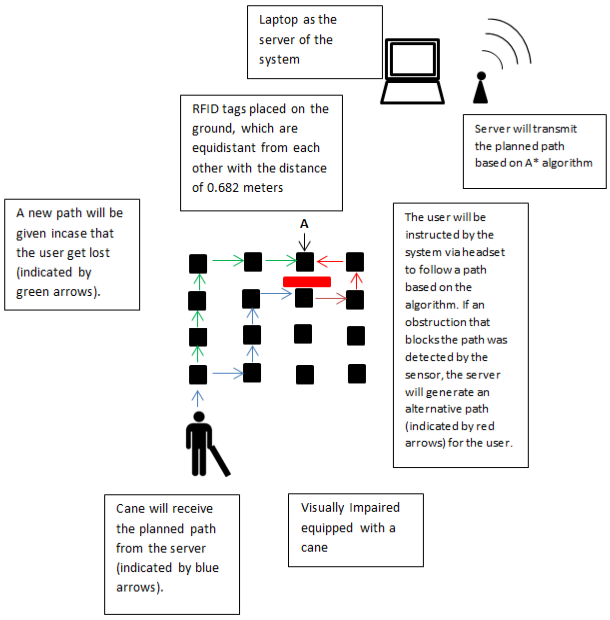

Fig. 1. Conceptual diagram of the system.

A. Cane for the Visually Impaired

The different important components of the cane are shown in Fig. 2. The ID detected by the reader is processed by the microcontroller. The ID is verified by the microcontroller. The microcontroller will receive the path from the laptop via ZigBee. The instruction is decoded by the microcontroller using the MP3 decoder. The decoded instruction, which is an audio output, is heard by the user using the headset.

There is a possibility that an obstacle will prevent the user to reach his/her destination. To solve this problem, the ultrasonic sensor, interfaced with the microcontroller unit, notifies the system via ZigBee if an obstacle was detected along the path. The software generates an alternative path to the destination for the user to avoid the obstacle.

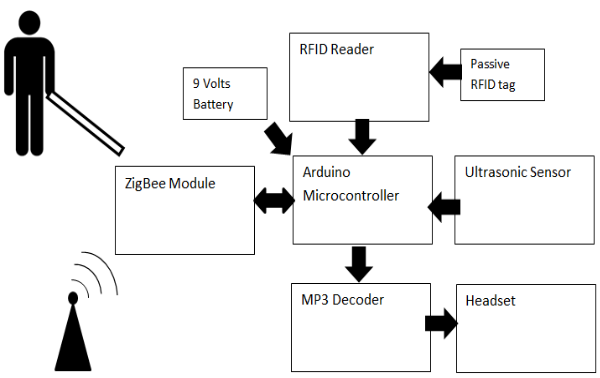

Fig. 2. Cane for visually impaired.

The ultrasonic sensor is angled at approx. 45 degrees because the white cane is inclined while being used by the visually impaired individual. It is positioned at the center part of the folding white cane to be able to detect static obstacles such as tables and chairs. The ultrasonic sensor HC-SR04 has a measuring angle of 30 degrees. The 0.5 m detection range will give enough reaction time for the visually impaired individual if a person or object is in front of him/her.

The low cost RFID reader is best positioned at the lower part of the white folding cane because the read range is only 8 – 12 cm.

B. Transmission of Shortest Path from the Server

Fig. 3 shows the laptop with the ZigBee module. The algorithm based on A* algorithm is programmed using MATLAB. The software displays and monitors the tags to be traversed by the user. The GUI (graphical user interface) is programmed using C# language. The path generated by the software is transmitted to the microcontroller via ZigBee module. In addition, the system is able to learn and adapt so that it can remember the previous obstacles that were detected.

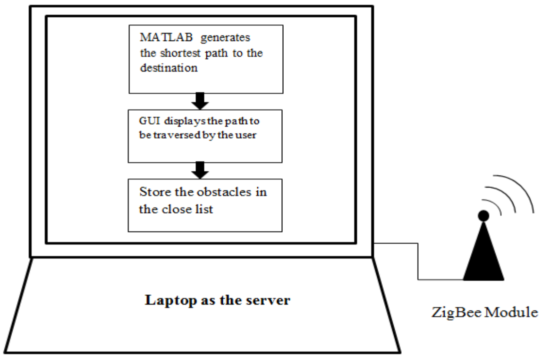

Fig. 3. Laptop with ZigBee module.

C. Shortest Path

Fig. 4 is the flowchart for determining the shortest path in the RFID tag grid based on A* algorithm. The first step in the algorithm is to include the initial or the starting tag in the open list. The system checks the open list for nodes [6]. The algorithm looks for the lowest F cost tag in the open list. This tag is labeled as the current tag and moved in the closed list. The closed list is the list of nodes that have been checked. The F cost can be computed by adding the movement cost (G cost) and the distance cost (H cost) of the tag. The next process is to check the 4 tags adjacent to the current tag. If there is an obstacle detected that blocks the adjacent tag, that tag will be included in the closed list thus it is ignored in checking for the lowest F cost. If a tag is not yet included in the open list, add it to the open list and make the current tag parent of this tag. A tag can be added in the open list if it is not yet included. The current tag will be the parent of the added tag. The F, G and H costs of the tag is recorded. If the tag is on the open list already, the algorithm checks to see if the path to that tag is better based on its G cost. Better path is considered if the G cost is lower. If it is better, the parent of the tag is changed to the current tag. The G and F scores of the tag are recalculated. The algorithm will repeat these steps until the current tag is equivalent to the destination tag. The shortest path is based on the lowest F score. It is generated by following the parent-child relationship of the tags starting from the destination until it reaches the initial tag.

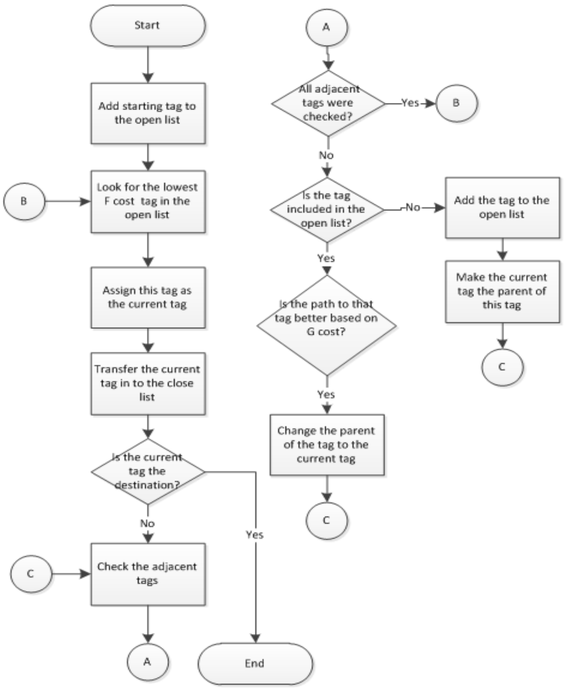

Fig. 4. Shortest path based on A* algorithm.

III. Results And Discussion

The following tables show the effect of placing the obstacle on five different positions in a given path. For this experiment, the initial position is tag A and the destination is tag P. The order of tags that the user needs to follow is: A-E-I-J-N-O-P tags in order shown in Fig. 5.

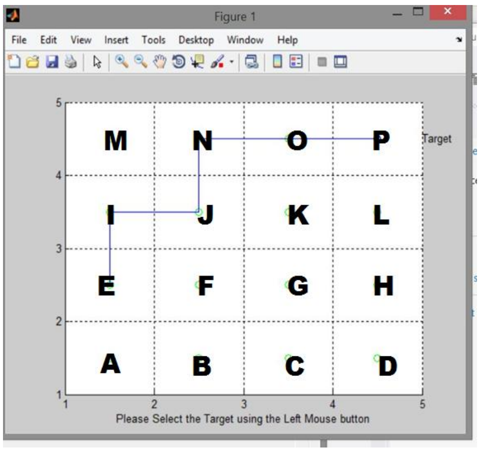

Fig. 5. Generated path based on A* algorithm.

An obstacle is placed in five different positions along the generated path by the system with initial position at tag A and the destination at tag P (see Fig. 5). The first step is the execution of the GUI in C#. The one operating the laptop pressed the "Generate Path" button in order to execute the A* algorithm in MATLAB. The user with the white folding cane is positioned for about 0.682 meters from the initial tag A. The user with the white folding cane started navigating the path after hearing the notification that the path was already sent. The time travelled, from the initial position up to the destination, of the user is measured in seconds using timer application in a smartphone. Fifteen trials for each position of the obstacle along the path are implemented. The type of obstacle used for the following tests is a chair. The dependent t-test for the statistical analysis is implemented in order to determine how the obstacles and the generated alternate paths will affect the time of traversal from initial position up to the intended destination.

TABLE I: T-Test for Obstacle at Tag E and Tag I

Trial	Obstacle at E (in seconds)	Obstacle at I (in seconds)
1.	70	67
2.	69	67
3.	70	67
4.	69	66
5.	70	68
6.	69	67
7.	69	67
8.	70	66
9.	71	67
10.	70	67
11.	69	67
12.	69	67
13.	70	67
14.	69	67
15.	70	66
Mean	69.6	66.86667
Hypothesized Mean Difference	3	
t-value	1.29291987	
p-value	0.108484293	

The hypothesized mean difference shown in Table I is equivalent to 3 because based on the computed means for obstacle at tag E and obstacle at Tag I, time travelled for obstacle at tag E is greater than time travelled for obstacle at Tag I by about 3 seconds. The null hypothesis for this experiment is that the mean for time travelled with obstacle at tag E is greater than the mean for time travelled with obstacle at Tag I. The *t*-value and *p*-value are generated using Microsoft Excel 2010. Since the *p*-value is greater than 0.05, the null hypothesis is accepted.

TABLE II: T-Test for Obstacle at Tag I and Tag J

Trial	Obstacle at I (in seconds)	Obstacle at J (in seconds)
1.	67	62
2.	67	63
3.	67	62
4.	66	62
5.	68	63
6.	67	64
7.	67	63
8.	66	63
9.	67	64
10.	67	63
11.	67	64
12.	67	63
13.	67	63
14.	67	64
15.	66	63
Mean	66.86667	63.06667
Hypothesized Mean Difference	4	
t-value	1	
p-value	0.167140972	

The hypothesized mean difference shown in Table II is equivalent to 4 because based on the computed means for obstacle at Tag I and obstacle at Tag J, time travelled for obstacle at Tag I is greater than time travelled for obstacle at

Tag J by about 4 seconds. The null hypothesis for this experiment is that the mean for time travelled with obstacle at Tag I is greater than the mean for time travelled with obstacle at Tag J. The *t*-value and *p*-value are generated using Microsoft Excel 2010. Since the *p*-value is greater than 0.05, the null hypothesis is accepted.

TABLE III: T-TEST FOR OBSTACLE AT TAG J AND TAG N

Trial	Obstacle at J (in seconds)	Obstacle at N (in seconds)
1.	62	61
2.	63	61
3.	62	60
4.	62	60
5.	63	62
6.	64	61
7.	63	62
8.	63	61
9.	64	61
10.	63	61
11.	64	60
12.	63	61
13.	63	61
14.	64	61
15.	63	60
Mean	63.06667	60.86667
Hypothesized Mean Difference	2	
t-value	0.898717034	
p-value	0.192000111	

The hypothesized mean difference shown in Table III is equivalent to 2 because based on the computed means for obstacle at Tag J and obstacle at Tag N, time travelled for obstacle at tag J is greater than time travelled for obstacle at Tag N by about 2 seconds. The null hypothesis for this experiment is that the mean for time travelled with obstacle at Tag J is greater than the mean for time travelled with obstacle at Tag N. The *t*-value and *p*-value are generated using Microsoft Excel 2010. Since the *p*-value is greater than 0.05, the null hypothesis is accepted.

TABLE IV: T-TEST FOR OBSTACLE AT TAG N AND TAG O

Trial	Obstacle at N (in seconds)	Obstacle at O (in seconds)
1.	61	75
2.	61	74
3.	60	75
4.	60	75
5.	62	74
6.	61	75
7.	62	75
8.	61	74
9.	61	75
10.	61	74
11.	60	74
12.	61	75
13.	61	75
14.	61	75
15.	60	75
Mean	60.86667	74.66667
Hypothesized Mean Difference	14	
t-value	124.9216678	
p-value	4.86978E-23	

The hypothesized mean difference shown in Table IV is equivalent to 14 seconds because based on the computed means for obstacle at Tag N and obstacle at Tag O, time travelled for obstacle at Tag O is greater than time travelled for obstacle at Tag N by about 14 seconds. The null hypothesis for this experiment is that the mean for time travelled with obstacle at Tag N is greater than the mean for time travelled with obstacle at Tag O. The *t*-value and *p*-value are generated using Microsoft Excel 2010. Since the *p*-value is lower than 0.05, the null hypothesis is rejected. The researcher accepted the alternate hypothesis where the mean of time traversal where there is an obstacle at Tag O is greater than the mean of time traversal where there is an obstacle at Tag N.

The reason why the time travelled is much longer where the obstacle is positioned at Tag O than the obstacle positioned at Tag N is because of the delay caused by transmitting the alternate path. Another reason is that the user must go around Tag O which is the tag coordinate before the destination Tag P. The alternate path generated where the obstacle is placed at Tag O consumed time. The alternate path that is generated and sent to the user when the obstacle is detected at Tag O contained four tag coordinates while the alternate path that is generated and sent to the user when the obstacle is detected at Tag N contained only three tag coordinates. Two seconds is required for the system to be able to send each coordinate of the alternate path.

IV. CONCLUSION

The study is successful in generating the shortest path to the intended destination by implementing A* algorithm in the system. The alternate paths are successfully generated in the study when an obstacle is detected or when the user got lost. Based on the experimental results of the study, the greater the number of coordinate paths in an alternate path generated by the algorithm that will be sent to the user, the greater the delay is. There is an increase in the time traversal to reach the destination when the system requested for an alternate path. The reason for the increase in time traversal is the delay in transmission of the alternate path.

ACKNOWLEDGMENT

The authors are grateful to the Department of Science and Technology’s Engineering Research and Development for Technology scholarship for providing the resources needed to be able to complete and finish the research. The scholarship grant continues to accelerate the development of researches that contributes to the development of the country.

REFERENCES

[1] J. Faria, S. Lopes, H. Fernandes, P. Martins, and J. Barroso, “Electronic white cane for blind people navigation assistance,” in *Proc. World Automation Congress (WAC)*, 2010, pp. 1-7.

[2] S. Chumkamon, P. Tuvaphanthaphiphat, and P. Keeratiwintakorn, “A blind navigation system using RFID for indoor environments,” in *Proc. 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 2008, vol. 2, pp. 765-768.

[3] D. T. Batarseh, T. N. Burcham, and G. M. McFadyen, “An ultrasonic ranging system for the blind,” in *Proc. the 1997 Sixteenth Southern Biomedical Engineering Conference*, 1997, pp. 411-413.

[4] M. Murad, A. Rehman, A. A. Shah, S. Ullah, M. Fahad, and K. M. Yahya, “RFAIDE — An RFID based navigation and object recognition assistant for visually impaired people,” in *Proc. 2011 7th International Conference on Emerging Technologies*, 2011, pp. 1-4.

[5] P. Dharani, B. Lipson, and D. Thomas, “RFID navigation system for the visually impaired," Thesis, Faculty of Worcester Polytechnic Institute, 2012.
[6] J. Sidi, L. W. Fa, and S. N. Junaini, “Shortest path simulation using interactive SVG map,” in *Proc. International Conference on Computer Technology and Development*, vol. 1, pp. 6-9, 2009.

Jayron Lester Sanchez was born in Manila, Philippines in 1992. He is a graduating student of the Computer Engineering (CpE) Department at Mapua Institute of Technology, Manila, Philippines for the degree BS computer engineering and MS computer engineering. He is expected to finish both degrees this coming March 2015.

Felicito S. Caluyo received his BS in electrical engineering and master of engineering from University of the Philippines Diliman in 1970 and 1977, respectively. He studied in France as a French government scholar and received his maitrise en electronique et communications, diplome d’etudes approfondies en electronique (D.E.A.), and Ph.D in electronics (communications optiques et microondes) from Universite de Limoges in 1985, 1986, and 1990, respectively. He was a MERALCO distinguished professorial chair on electronics and communications engineering at De La Salle University from 1990 up to 2006. He joined Mapua Institute of Technology as the dean of School of EECE from 2006 up to 2014.

Analyn N. Yumang was born in Batangas City in 1978. She received the B.S. CoE and MEng’g CpE from Mapua Institute of Technology Manila, Philippines in 2002 and 2005, respectively. Since 2002, she has been with the School of EECE, Mapua Institute of Technology, where she is currently an associate professor. She is the UNIX/.NET coordinator of the Center for Continuing Education and Special Competencies since 2005 and also the adviser of the Junior Philippines Computer Society since 2009.