Visual Assistance Software for the Visually Impaired

Vasu Gupta

MacEwan University

Abstract

The proposed project involves the creation of a visual assistance software for the visually impaired to identify people in their surroundings to avoid potential collisions. Object detection technique using TensorFlow Lite Object Detection API is performed to identify the people in the view of a visually impaired person. Vibrations of different lengths were used to notify the visually impaired about the location of the person in the view. The software is tested in various conditions to verify its working.

Introduction

Design of urban areas is complex and requires one not only to design for the masses but to incorporate issues surrounding gender, diversity, and inclusion. According to World Health Organization in 2011, there are 285 million visually impaired people which include weak sighted (246 million) and blind (39 million) [1]. Major challenges with daily life for the visually impaired include localization and navigation [2], object detection, and recognition [3], developing social relationships [4], and reading and writing problems [4]. Traditionally, the visually impaired tend to sway their white cane to navigate themselves and detect their surroundings [5]. They also make use of guide-dogs to perform their daily basic activities.

Several products have been created to aid the visually impaired. To solve the problem of navigation and localization, researchers like Bai, Lian, Liu, Wang, and Liu created a smart guiding device which helps the visually impaired navigate and helps avoiding obstacles [5]. To gather depth information of the surroundings, they made use of depth camera, and ultrasonic sensor to find out the obstacle distance [5]. In another study, Augmented Reality (AR) was used to recognize objects indoors and converted the data to an audio form [6]. To read the signs and text, a prototype was created which involves text extraction using Tesseract Optical Character Recognition (OCR) [7].

The navigation and navigation setup created by Bai, Lian, Liu, Wang, and Liu worked well to avoid collisions, but does not provide information about the surroundings to the visually impaired. Moreover, the study using AR to detect objects was limited to camera connected to a laptop. Furthermore, the text extraction was less accurate and further work is needed to make it work ideally.

Therefore, none of these products deal with image classification to gather information about the surroundings. For instance, if the sensors detect the obstacle, the user is then notified about it's presence, but is still not aware of the obstacle.

Figure 1: EPSON Moverio BT-300 Smart Glasses

Methods

The proposed method consists of EPSON Moverio BT-300 smart eyeglasses for the visually impaired to avoid collisions. The glasses are attached to a controller running on Android™ 5.1 powered by Intel® Atom x5 1.44GHz Quad Core CPU [8] (*Figure 1*). The device just weighs 69g has a 5 Megapixel camera connected to it [8] which captures visual input at 1280x720 pixels. The glasses have an in-built 0.43" wide Si-OLED display on which the Android OS is projected [8]. This display is used for debugging for the most part and will not be useful for the user. The device supports earphone connectivity with 4 pin mini jack [8].

As the controller is an android device, we created an android app which uses the device camera to perform real-time object detection on the eyeglasses. This app uses TensorFlow Lite, an open source deep learning framework to carry out numerical computations using data flow graphs. With a small binary size and low latency, TensorFlow makes on-device machine learning inference possible [9]. "TensorFlow Lite supports hardware acceleration with the Android Neural Networks API" [10]. The app detects people in real-time in the images captured by the on-device camera "using a quantized MobileNet SSD model trained on the COCO dataset" [11].

The on-device camera is used to give input image to the app. The app passes the image to the TensorFlow Lite Object Detection API, it performs a series of operations on the data and returns the objects detected in the image, their location in the image, number of objects detected and the confidence score of detections. As the model is trained on COCO dataset, the objects identified are large in number. These detections give many false positives/negatives; therefore, we are limiting the detection to only identification of people to avoid giving false information to the user. After performing the detection, the app draws rectangles around the people detected with a confidence of more than 50%. Applying this technique to the video stream in real-time gives an augmented reality (AR) like experience.

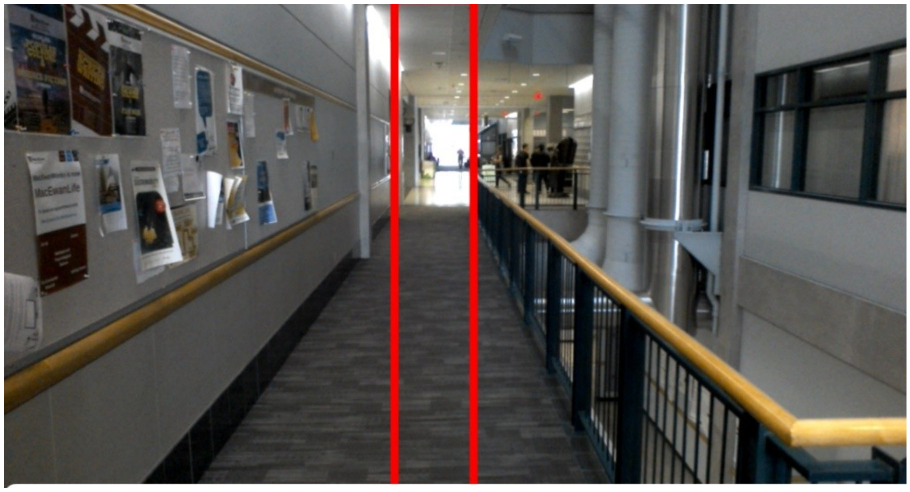

Figure 2: Application view

To represent the line of sight of the user, the application has a rectangle in the middle (*Figure 2*). The application has three types of vibrations (denoted by A, B and C) to notify the users about the people in their surroundings. Vibration A is a 500ms long buzz in the controller which indicates the presence of a person the line of sight of the user. Vibration B conveys the absence of people in the frame(scene) by giving a 100ms vibration. Vibration C, a 250ms buzz indicates the presence of a person in the frame (scene), but not in the line of sight to alert the user.

To produce a positive user experience, every notification is only given once after the detection. For example, if a person entered the line of sight of the user, then the controller vibrates with Vibration A and will not vibrate again until there is a change in the situation. If there is a person in the line of sight, then the user is alarmed and would be cautious to move forward. If the person moves his/her head toward the right or left and way is clear with no people, the user is notified with Vibration B or C (Depending on the situation). To notify the user about the possible collisions, we notify the user with Vibration C indicating that there is a person in the frame, but not in the line of sight; hence be cautious while walking. However, if that person detected moves into the line of sight, then the user is notified by Vibration A. If the person moves out of the frame, then the user is given Vibration B. It would still act the same if more people enter the frame as it already gave a notification about people who originally entered.

No. of people (4 situations)	Position in the frame (2 situations)	Orientation (3 situations)	Motion (3 situations)
0	-	-	-
1	side/ middle	side/front/back	Moving/SSt/SSi
2	side/ middle	side/front/back	Moving/SSt/SSi
2+	side/ middle	side/front/back	Moving/SSt/SSi

Table 1.1: Testing Scenarios

There are various situations in which this device needs to be tested (*Table 1.1*). The situations include the position of person detected is at the center of the frame, or on the sides of the frame. The orientation of the person can also vary; hence, the device is also tested by capturing front, back and side angles of the human being. Situations like moving subject and the user, stationary and standing (SSt) subject, and stationary and sitting (SSi) subject are also tested. Moreover, the device is also tested in a scenario having no person in the frame. This results in a total of 55 scenarios for testing in different possible cases.

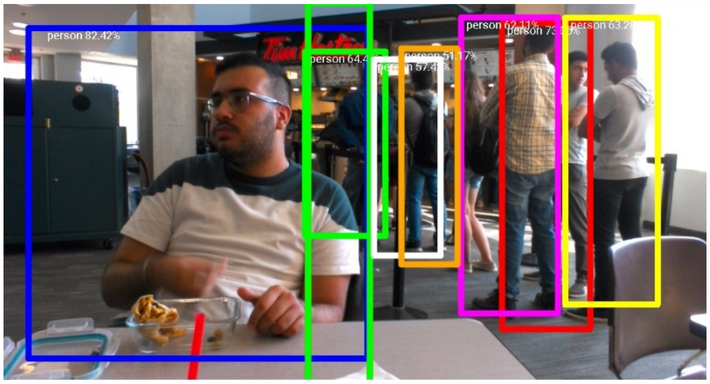

Figure 3

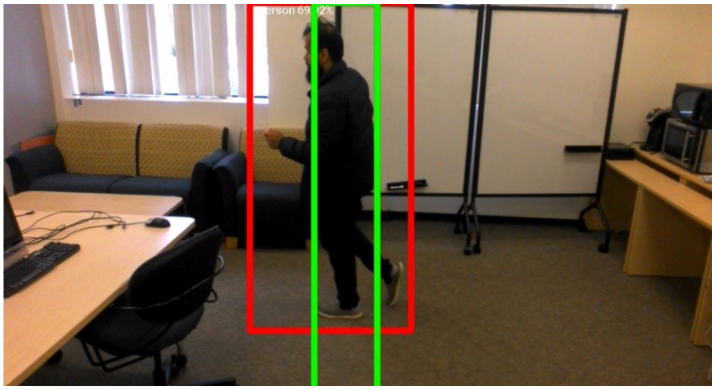

Figure 4

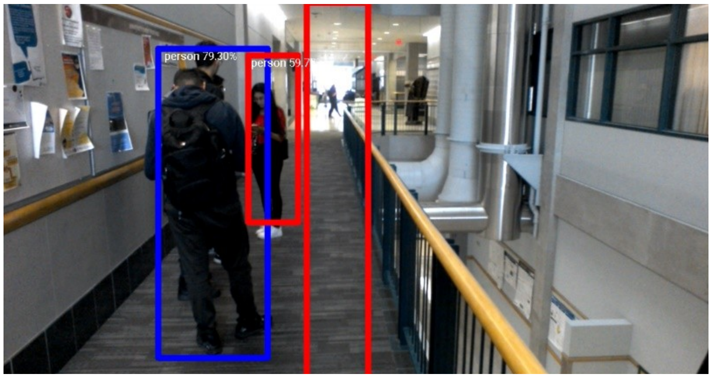

Figure 5

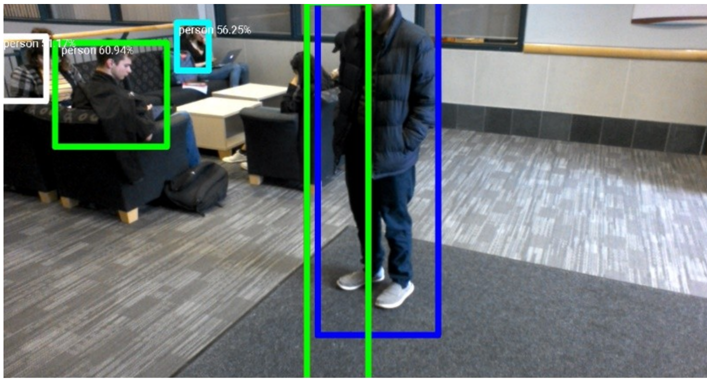

Figure 6

Result

We have developed a new type of navigation assistant for the visually impaired. The software is designed and implemented to detect people using EPSON Moverio BT-300 glasses as the hardware. The software is designed to produce vibration output to notify the visually impaired about the different scenarios possible. When the device gets tested in the above mentioned 55 situations, it gives accurate results in the form vibrations to the user. The software successfully recognizes a person or multiple people from different angles i.e. front, back and side (*Figure 3*). Whenever, a person is detected, a rectangle is drawn around the person to identify its position. If the detected person is in the center of the screen or overlaps with the center box, the user is given Vibration A (*Figure 4*). If the user is detected on the sides, the vibration is C (*Figure 5*). However, if there is one person on the side, and one in the center, then Vibration A is given because of its higher priority (danger) (*Figure 6*). Detecting people in an environment when the surrounding people move works accurately and gives vibrations according to their position in the frame. Similarly, the test for different orientations gives the same vibration depending on their position in the frame. When there is no person found in the frame, then user is notified by Vibration B (*Figure 2*). The software is tested around the MacEwan campus including library, cafeteria and hallways. Overall, the application does a good job on detecting people with a great accuracy and gives vibrations as required. The software is robust and can be easily handled by the visually impaired as it does not involve any inputs from the user. The software ensures a safe mobility for the user as well.

Conclusion

In this work, a set of electronic smart glasses are proposed for the visually impaired community to help them navigate. This device is used to make the visually impaired feel independent to move around. The controller connected with the eyeglasses notifies the user by three vibrations which are easily comprehensible. The future advancements on the device are needed to make it identify distance to their obstacles and detecting more objects with high accuracy. The end goal is to make the visually impaired navigate just like a sighted person.

References

[1] B. Soveny, G. Kovacs, and Z. T. Kardkovacs, “Blind guide - A virtual eye for guiding indoor and outdoor movement,” *2014 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom)*, pp. 343–347, 2014.

[2] A. Hub, J. Diepstraten, and T. Ertl, “Design and development of an indoor navigation and object identification system for the blind,” *ACM SIGACCESS Accessibility and Computing*, no. 77-78, pp. 147–152, Jan. 2003.

[3] A. Benabidvww and M. Alzuhair, “User involvement in the development of indoor navigation system for the visually impaired: A needs-finding study,” *2014 3rd International Conference on User Science and Engineering (i-USEr)*, pp. 97–102, 2014.

[4] W. R. Otyola, G. M. Kibanja, and A. M. Mugagga, “Challenges faced by visually impaired students at Makerere and Kyambogo Universities,” *Makerere Journal of Higher Education*, vol. 9, no. 1, p. 75, 2017.

[5] J. Bai, S. Lian, Z. Liu, K. Wang, and D. Liu, “Smart guiding glasses for visually impaired people in indoor environment,” *IEEE Transactions on Consumer Electronics*, vol. 63, no. 3, pp. 258–266, 2017.

[6] J. Sánchez and A. Tadres, “Augmented reality application for the navigation of people who are blind,” *International Journal on Disability and Human Development*, vol. 10, no. 1, Jan. 2011.

[7] M. Rajesh, B. K. Rajan, A. Roy, K. A. Thomas, A. Thomas, T. B. Tharakan, and C. Dinesh, “Text recognition and face detection aid for visually impaired person using Raspberry PI,” *2017 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, 2017.

[8] Ismail, “The Moverio BT-300 (AR/DEV Edition) ... IS... EPIC.,” *Smart Glasses | Smart Glasses | For Home | Epson Canada*, 19-Jun-2018. [Online]. Available: https://epson.ca/For-Home/Smart-Glasses/Smart-Glasses/Moverio-BT-300-Smart-Glasses-(AR-Developer-Edition)-/p/V11H756020.

[9] “TensorFlow Lite guide: TensorFlow,” *TensorFlow*. [Online]. Available: https://www.tensorflow.org/lite/guide.

[10] S. Penchikala, “TensorFlow Lite Supports On-Device Conversational Modeling,” *InfoQ*, 29-Nov-2017. [Online]. Available: https://www.infoq.com/news/2017/11/tensorflow-lite/.

[11] Tensorflow, “Tensorflow,” *GitHub*, 26-Feb-2019. [Online]. Available: https://github.com/tensorflow/examples/blob/master/lite/examples/object_detection/android/README.md.