An AI-Based Visual Aid With Integrated Reading Assistant for the Completely Blind

Muiz Ahmed Khan, Pias Paul, Mahmudur Rashid, *Student Member, IEEE*, Mainul Hossain, *Member, IEEE*, and Md Atiqur Rahman Ahad, *Senior Member, IEEE*

***Abstract*—Blindness prevents a person from gaining knowledge of the surrounding environment and makes unassisted navigation, object recognition, obstacle avoidance, and reading tasks a major challenge. In this work, we propose a novel visual aid system for the completely blind. Because of its low cost, compact size, and ease-of-integration, Raspberry Pi 3 Model B+ has been used to demonstrate the functionality of the proposed prototype. The design incorporates a camera and sensors for obstacle avoidance and advanced image processing algorithms for object detection. The distance between the user and the obstacle is measured by the camera as well as ultrasonic sensors. The system includes an integrated reading assistant, in the form of the image-to-text converter, followed by an auditory feedback. The entire setup is lightweight and portable and can be mounted onto a regular pair of eyeglasses, without any additional cost and complexity. Experiments are carried out with 60 completely blind individuals to evaluate the performance of the proposed device with respect to the traditional white cane. The evaluations are performed in controlled environments that mimic real-world scenarios encountered by a blind person. Results show that the proposed device, as compared with the white cane, enables greater accessibility, comfort, and ease of navigation for the visually impaired.**

***Index Terms*—Blind people, completely blind, electronic navigation aid, Raspberry Pi, visual aid, visually impaired people, wearable system.**

I. Introduction

BLINDNESS or loss of vision is one of the most common disabilities worldwide. Blindness, either caused by natural means or some form of accidents, has grown over the past decades. Partially blind people experience a cloudy vision, seeing only shadows, and suffer from poor night vision or tunnel vision. A completely blind person, on the other hand, has no vision at all. Recent statistics from the World Health Organization estimate the number of visually impaired or blind people to be about 2.2 billion [1]. A white cane is used traditionally by the blind people to help them navigate their surroundings, although use of the white cane does not provide information for moving obstacles that are approaching from a distance. Moreover, white canes are unable to detect raised obstacles that are above the knee level. Trained guide dogs are another option that can assist the blind. However, trained dogs are expensive and not readily available. Recent studies have proposed several types [2]–[9] of wearable or hand-held electronic travel aids (ETAs). Most of these devices integrate various sensors to map the surroundings and provide voice or sound alarms through headphones. The quality of the auditory signal, delivered in real-time, affects the reliability of these gadgets. Many ETAs, currently available in the market, do not include a real-time reading assistant and suffer from a poor user interface, high cost, limited portability, and lack of hands-free access. These devices are, therefore, not widely popular among the blind and require further improvement in design, performance, and reliability for use in both indoor and outdoor settings.

In this article, we propose a novel visual aid system for completely blind individuals. The unique features, which define the novelty of the proposed design, include the following.

1) Hands free, wearable, low power, and compact design, mountable on a pair of eyeglasses, for the indoor and outdoor navigation with an integrated reading assistant.
2) Complex algorithm processing with a low-end configuration.
3) Real-time, camera-based, accurate distance measurement, which simplifies the design and lowers the cost by reducing the number of required sensors.

The proposed setup, in its current form, can detect both stationary and moving objects in real time and provide auditory feedback to the blind. In addition, the device comes with an in-built reading assistant that is capable of reading text from any document. This article discusses the design, construction, and performance evaluation of the proposed visual aid system and is organized as follows. Section II summarizes the existing literature on blind navigation aids, highlighting their benefits and challenges. Section III presents the design and the working principle of the prototype, while Section IV discusses the experimental setup for performance evaluation. Section V summarizes the results using appropriate statistical analysis. Finally, Section VI concludes the article.

Manuscript received March 31, 2020; revised July 23, 2020; accepted September 6, 2020. Date of publication October 20, 2020; date of current version November 12, 2020. This article was recommended by Associate Editor Z. Yu. *(Corresponding author: Mainul Hossain.)*

Muiz Ahmed Khan, Pias Paul, and Mahmudur Rashid are with the Department of Electrical and Computer Engineering, North South University, Dhaka 1229, Bangladesh (e-mail: muiz.khan@northsouth.edu; paul.pias@northsouth.edu; mahmudur.rashid@northsouth.edu).

Mainul Hossain is with the Department of Electrical and Electronic Engineering, University of Dhaka, Dhaka 1000, Bangladesh (e-mail: mainul.eee@du.ac.bd).

Md Atiqur Rahman Ahad is with the Department of Electrical and Electronic Engineering, University of Dhaka, Dhaka 1000, Bangladesh, and also with the Department of Intelligent Media, Osaka University, Suita 565-0871, Japan (e-mail: atiqahad@du.ac.bd).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2020.3027534

II. Relevant Work

The electronic aids for the visually impaired can be categorized into three different subcategories, ETAs, electronic orientation aids, and positional locator devices. ETAs provide object detection, warning, and avoidance for safe navigation [10]–[12]. ETAs work in few steps; sensors are used to collect data from the environment, which are then processed through a computing device to detect an obstacle or object and give the user a feedback corresponding to the identified object. The ultrasonic sensors can detect an object within 300 cm by generating a 40 kHz signal and receiving reflected echo from the object in front of it. The distance is calculated based on the pulse count and time-of-flight (TOF). Smart glasses [2], [9] and boots [12], mounted with ultrasonic sensors, have already been proposed as an aid to the visually impaired. A new approach by Katzschmann *et al.* [13] uses an array of infrared TOF distance sensors facing in different directions. Villanueva and Farcy [14] combine a white cane with near-IR LED and a photodiode to emit and detect the IR pulses reflected from obstacles, respectively. Cameras [15], [16] and binocular vision sensors [17] have also been used to capture the visual data for the blind.

Different devices and techniques are used for processing the collected data. Raspberry Pi 3 Model B+, with open computer vision (OpenCV) software, has been used to process the images captured from the camera [18]. Platforms such as Google tango [3] have also been used. A cloud-enabled computation enables the use of wearable devices [2]. A field-programmable gate array is also another option to process the gathered data [19]. The preprocessing of captured images is done to reduce noise and distortion. Images are manually processed by using the Gaussian filter, gray scale conversion, binary image conversion, edge detection, and cropping [20]. The processed image is then fed to the Tesseract optical character recognition (OCR) engine to extract the text from it [21]. The stereo image quality assessment [17] employs a novel technique to select the best image, out of many. The best image is then fed to a convolutional neural network (CNN), which is trained on big data and runs on a cloud device. The audio feedback in most devices is provided through a headset or a speaker. The audio is either a synthetic voice [20] from the text-to-speech synthesis system [22] or a voice user interface [23] generating a beep sound. Vibrations and tactile feedback are also used in some systems.

Andò *et al.* [24] introduced a haptic device, similar to the white cane, with an embedded smart sensing strategy and an active handle, which detects an obstacle and produces vibration mimicking a real sensation on the cane handle. Another traditional white cane like system, guide cane [13], rolls on wheels and has steering servo motors to guide the wheels by sensing the obstacles from ultrasonic sensors. The backdrop of this system is that the user must always hold the device by their hand, whereas, many systems, which provide a hands-free experience, are readily available. NavGuide [12] and NavCane [25] are assistive devices that use multiple sensors to detect obstacles up to the knee level. Both NavGuide and NavCane are equipped with wet floor sensors. NavCane can be integrated into the white cane systems and offers a global positioning system (GPS) with a mobile communication module.

A context-aware navigation framework is demonstrated by Xiao *et al.* [4], which provides visual cues and distance sensing along with location-context information, using GPS. The platform can also access geographic information systems, transportation databases, and social media with the help of Wi-Fi communication through the Internet. Lan *et al.* [26] proposed a smart glass system, which can detect and recognize road signs, such as public toilets, restaurants, and bus stops, in the cities in real time. This system is lightweight, portable, and flexible. However, reading out the road signage alone may not carry enough information for a blind user to be comfortable in an outdoor environment. Since public signs can be different in different cities, therefore, if a sign is not registered in the database of the system, the system will not be able to recognize it. Hoang *et al.* [20] designed an assistive system using mobile Kinect and a matrix of electrodes for obstacle detection and warning. However, the system has a complex configuration and an uncomfortable setup because the sensors are always placed inside the mouth during navigation. Furthermore, it is expensive and has less portability.

Islam *et al.* [27] presented a comprehensive review of sensor-based walking assistants for the visually impaired. The authors identified key features that are essential for an ideal walking assistant. These include low cost, simple, and lightweight design with a reliable indoor and outdoor coverage. Based on the feedback from several blind user groups, software developers, and engineers, Dakopoulos and Bourbakis [10] also identified 14 structural and operational features that describe an ideal ETA for the blind.

Despite numerous efforts, many existing systems do not incorporate all features to the same satisfactory level and are often limited by cost and complexity. Our main contribution here was to build a simple, low cost, portable, and hands-free ETA prototype for the blind, with text-to-speech conversion capabilities for basic, everyday indoor and outdoor use. While, the proposed system, in its present form, lacks advanced features, such as the detection of wet floors and ascending staircases, reading of road signs, use of GPS, or mobile communication module, the flexible design presents opportunities for future improvements and enhancements.

III. Design of the Proposed Device

We propose a visual aid for completely blind individuals, with an integrated reading assistant. The setup is mounted on a pair of eyeglasses and can provide real-time auditory feedback to the user through a headphone. Camera and sensors are used for distance measurement between the obstacle and the user. The schematic view in Fig. 1 presents the hardware setup of the proposed device, while Fig. 2 shows a photograph of the actual device prototype.

For the object detection part, multiple techniques have been adopted. For instance, TensorFlow object detection application programming interface (API), frameworks, and libraries, such as OpenCV and Haar cascade classifier, are used for detecting faces and eyes and implement distance measurement. Tesseract, which is a free OCR engine, for various operating systems, is

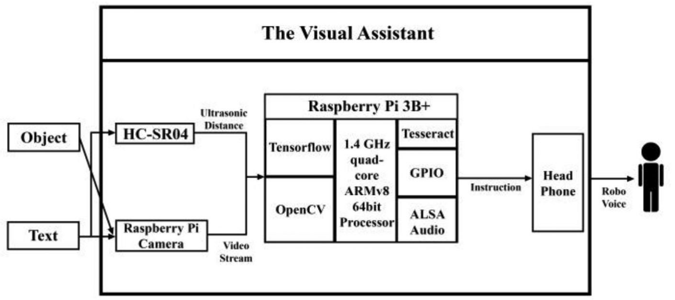


Fig. 1. Hardware configuration of the proposed system. The visual assistant takes the image as inputs, processes it through the Raspberry Pi Processor, and gives the audio feedback through a headphone.

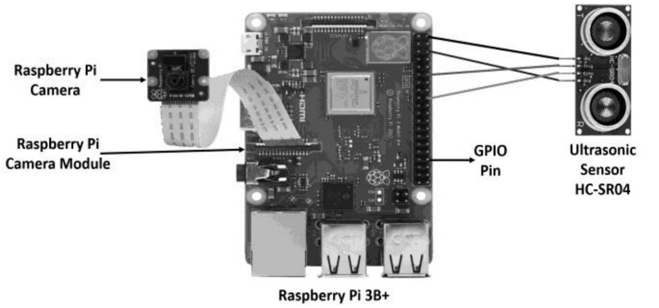


Fig. 3. Basic hardware setup: Raspberry Pi 3 Model B+ and associated module with the camera and ultrasonic sensors.

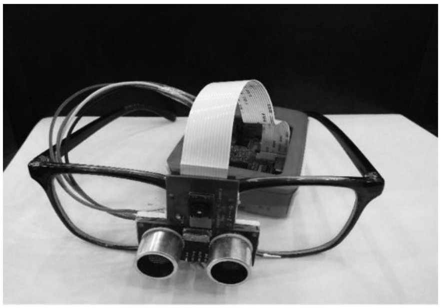


Fig. 2. Proposed prototype. Raspberry Pi with the camera module and ultrasonic sensors mounted on a regular pair of eyeglasses.

used to extract text from an image. In addition, eSpeak, which is a compact open-source speech synthesizer (text-to-speech), is used for auditory feedback for object type and distance between the object and the user. For obstacles within 40–45 inches of the user, the ultrasonic transducer (HC-SR04) sets off a voice alarm, while the eSpeak speech synthesizer uses audio feedback to inform the user about his or her distance from the obstacle, thereby, alerting the blind person and avoiding any potential accident.

Raspberry Pi 3 Model B+ was chosen as the functional device owing to its low cost and high portability. Also, unlike many existing systems, it offers a multiprocessing capability. To detect obstacles and generate an alarm, a TensorFlow object detection API has been used. The API was constructed using robust deep learning algorithms that require massive computing power. Raspberry Pi 3 Model B+ offers a 1.2 GHz quad-core ARM Cortex A53 processor that can output a video at a full 1080p resolution with desired details and accuracy. In addition, it has 40 general purpose input/output (GPIO) pins, which were used, in the proposed design, to configure the distance measurement by the ultrasonic sensors.

A. Data Acquisition

Fig. 3 shows how the Raspberry Pi 3 Model B+ is connected to other components in the system. Data are acquired in two ways. Information that have red, green, and blue (RGB) data were acquired using the Raspberry Pi camera module V2, which has a high quality, 8-megapixel Sony IMX219 image sensor. The camera sensor, featuring a fixed focus lens, has been custom designed to fit onboard into Raspberry Pi. It can capture 3280 pixels × 2464 pixels static images and supports 1080p, 720p, and 640 pixels × 480 pixels video. It is attached to the Pi module through small sockets, using the dedicated camera serial interface. The RGB data are retrieved by our program, in real time, and can recognize objects from every video frame that is already known to the system.

To acquire data from the ultrasonic rangefinder, HC-SR04 was mounted below the camera, as shown in Fig. 3. There are four pins on the ultrasound module that were connected to the Raspberry Pi's GPIO ports. VCC was connected to pin 2 (VCC), GND to pin 6 (GND), TRIG to pin 12 (GPIO18), and the ECHO to pin 18 (GPIO24). The ultrasonic sensor output (ECHO) will always give output LOW (0 V), unless it has been triggered, in which case, it will give output HIGH (5 V). Therefore, one GPIO pin was set as an output to trigger the sensor and one as an input to detect the ECHO voltage change. However, this HC-SR04 sensor requires a short 10 s pulse to trigger the module. This causes the sensor to start generating eight ultrasound bursts, at 4 kHz, to obtain an echo response. So, to create the trigger pulse, the trigger pin is set HIGH for 10 s and then set to LOW again. The sensor sets ECHO to HIGH for the time it takes for the pulse to travel the distance and the reflected signal to travel back. Once a signal is received, the value changes from LOW (0) to HIGH (1) and remains HIGH for the duration of the echo pulse. From the difference between the two recorded time stamps, the distance between the ultrasound source and the reflecting object can be calculated. The speed of sound depends on the medium it is traveling through and the temperature of that medium. In our proposed system, 343 m/s, which is the speed of sound at sea level, has been used.

B. Feature Extraction

The TensorFlow object detection API is used to extract features (objects) from images captured from the live video stream. The TensorFlow object detection API is an open-source framework, built on the top of TensorFlow, which is easy to integrate, train, and create models that perform well in different scenarios. TensorFlow represents deep learning networks as the core of the object detection computations. The foundation of TensorFlow is the graph object, which contains a network of nodes. GraphDef objects can be created by the ProtoBuf library to save the network. For the proposed design, a pretrained

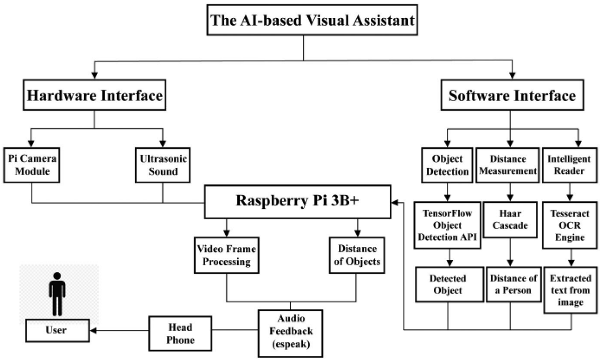


Fig. 4. Complete workflow of the proposed system. The hardware interface collects data from the environment. The software interfaces process the collected data and generate an output response through the audio interface. Raspberry Pi 3B+ is the central processing unit of the system.

model, called single-shot detection (SSD)Lite-MobileNet, from the TensorFlow detection model zoo, has been used. The model zoo is Google's collection of pretrained object detection models trained on different datasets, such as the common objects in context (COCO) dataset [28]. This model was particularly chosen for the proposed prototype because it does not require high-end processing capabilities, making it compatible with the low processing power of the Raspberry Pi. To recognize objects from the live video stream, no further training is required since the models have already been trained on different types of objects. An image has an infinite set of possible object locations and detecting these objects can be challenging because most of these potential locations contain different background colors, not actual objects. The SSD models usually use the one-stage object detection, which directly predicts object bounding boxes for an image. This has a simple and faster architecture, although the accuracy is comparatively lower than the other state-of-the-art object detection models having two or more stages.

C. *Workflow of the System*

Fig. 4 shows the complete workflow of the proposed system with the hardware and software interfaces. Every frame of the video is being processed through a standard convolutional network to build a feature representation of the original image or the frame. This backbone network is then pretrained on Image-Net in the SSD model, as an image classifier, to learn how to extract features from an image using SSD. Then, the model manually defines a collection of aspect ratios for bounding boxes, at each grid cell location. For each bounding box, it predicts the offsets for the bounding box coordinates and dimensions. Along with this, the distance measurement is processed using both the depth information and the ultrasonic sensor. In addition, the reading assistant works without interrupting any of the prior processes. All the three features run in the software interface with the help of the modules from the hardware interface.

D. *Object Detection*

The human brain focuses on the region of interests and salient objects, recognizing the most important and informative parts of the image [29]. By extracting these visual attributes [30], the deep learning techniques can mimic human brains and can detect salient objects from images, video frames [31], and even from optical remote sensing [32]. A pixelwise and nonparametric moving object detection method [33] can extract from the spatial and temporal features and detect moving objects with intricate background from the video frame. Many other techniques for object detection and tracking, from the video frame, such as the object-level RGB-D video segmentation, are also commonly used [34].

For object detection, every object must be localized within a bounding box, in each frame of a video input. A "region proposal system" or Regions + CNN (R-CNN) can be used [35], where, after the final convolutional layers, a regression layer is added to get a number that consists of four variables x_0, y_0, width, and height of the image. This process must train the support vector machine for each class, to classify between the object and background, while proposing the region in each image. In addition, a linear regression classifier needs to be trained, which will output some correction factor. To eliminate the unnecessary bounding boxes from each class, the intersection over union method must be applied to filter out the actual location of an object in each image. Methods used in faster R-CNN dedicatedly provide region proposals, followed by a high-quality classifier to classify these proposals [35]. These methods are very accurate but come at a big computational cost. Furthermore, because of the low frame rate, these methods are not fit to be used on embedded devices.

Object detection can also be done by combining the two tasks into one network by having a network that produces proposals instead of having a set of predefined boxes to look for objects. The computation that is already made during the classification, to localize the objects, could be reused. This is achieved by using the convolutional feature maps from the later layers of a network, upon which convolutional filters can be run, to predict class scores and bounding box offsets at once. The SSD detector [36] uses multiple layers that provide a finer accuracy on the objects with different scales. As the layers go deeper, the bigger objects become more visible. SSD is fast enough to infer objects in the real-time video. In SSDLite, MobileNetv2 [37] was used as the backbone and has depthwise separable convolutions for the SSD layers. The SSDLite models make predictions on a fixed-sized grid. Each cell in this grid is responsible for detecting objects, in a location, from the original input image and produces two tensors as the outputs that contain the bounding box predictions for different classes. SSDLite has several different grids ranging in size from 19 × 19 to 1 × 1 cells. The number of bounding boxes per grid cell is 3, for the largest grid, and 6, for the others, making a total of 19 × 17 boxes.

For the designed prototype, Google's object detection API, COCO, has been used, which has 3 000 000 images of 90 most found objects. The API provides five different models, making a tradeoff between the speed of execution and the accuracy in placing bounding boxes. SSDLite-MobileNet, whose architecture is shown in Fig. 5, is chosen as the object detection algorithm since it requires less processing power. Basically, SSD is designed to be independent of the base network and so it can run on

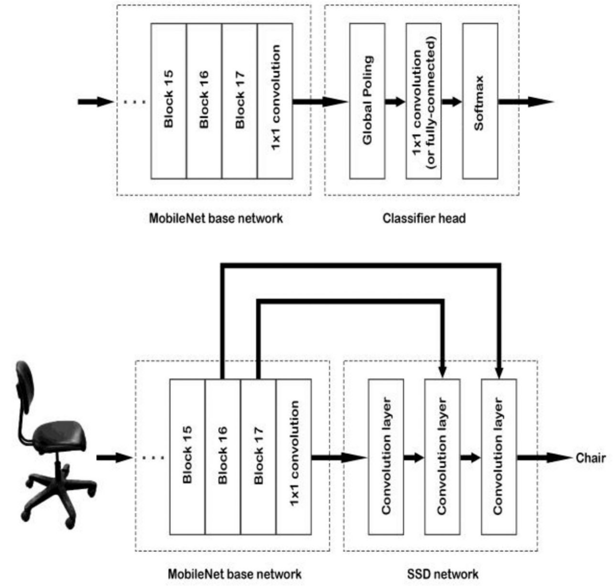


Fig. 5. SSD Lite-MobileNet architecture.

MobileNet [35]. With the SSDLite on top of the MobileNet, we were able to get around 30 frames per second (fps), which is enough to evaluate the system in real-time test cases. In places where online access is either limited or absent, the proposed device can operate offline as well. In SSDLite-MobileNet, the "classifier head" of MobileNet, which made the predictions for the whole network, gets replaced with the SSD network. As shown in Fig. 5, the output of the base network is typically a 7 × 7 pixel image, which is fed into the replaced SSD network to do further feature extraction. Not only the replaced SSD network takes the output of the base network but it also takes the outputs of several previous layers. The MobileNet layers convert the pixels from the input image into features that describe the contents of the image and pass these along to the other layers. A new family of object detectors, such as POLY-YOLO [38], DETR [39], Yolact [40], and Yolact++ [41], introduced instance segmentation along with object detection. Despite the efforts, many object detection methods still struggle with medium and large-sized objects. Researchers have, therefore, focused on proposing better anchor boxes to scale up the performance of an object detector with regards to the perception, size, and shape of the object. Recent detectors offer a smaller parameter size while significantly improving mean average precision. However, large input frame sizes limit their use in the systems with low processing power.

For object detection, MobileNetv2 is used as the base network, along with SSD since it is desirable to know both high-level as well as low-level features by reading the previous layers. Since object detection is more complicated than the classification, SSD adds many additional convolution layers on the top of the base network. To detect objects in live feeds, we used a Pi camera. Basically, our script sets paths to the model and label maps, loads the model into memory, initializes the Pi camera, and then begins performing object detection on each video frame from the Pi camera. Once the script initializes, which can take up to a maximum of 30 s, a live video stream will begin and common objects inside the view of the user will be identified. Next, a rectangle is drawn around the objects. With the SSDLite model and the Raspberry Pi 3 Model B+, a frame rate higher than 1 fps can be achieved, which is fast enough for most real-time object detection applications.

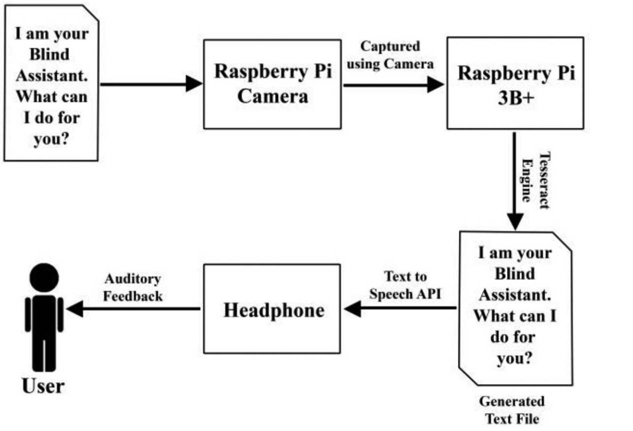


Fig. 6. Workflow for the reading assistant. Raspberry Pi gets a single frame from the camera module and runs through the Tesseract OCR engine. The test output is then converted to the audio.

E. Reading Assistant

The proposed system integrates an intelligent reader that will allow the user to read text from any document. An open-source library, Tesseract version-4, which includes a highly accurate deep learning-based model for text recognition, is used for the reader. Tesseract has unicode (UTF-8) support and can recognize many languages along with various output formats: plain-text, hocr (HTML), pdf, tsv, and invisible-text-only pdf. The underlying engine uses a long short-term memory (LSTM) network. LSTM is part of a recurrent neural network, which is a combination of some unfolded layers that use cell states in each time steps to predict letters from an image. The captured image is divided into horizontal boxes, and in each time step, the horizontal boxes are being analyzed with the ground truth value to predict the output letter. LSTM uses gate layers to update the cell state, at each time step, by using several activation functions. Therefore, the time required to recognize texts can be optimized.

Fig. 6 shows the working principle of the reading assistant. An image is captured from the live video feed without interrupting the object detection process. In the background, Tesseract API will extract the texts from the image and save them in a temporary text file. Then it reads out the text from the text file using the text-to-speech engine eSpeak. The accuracy of the Tesseract OCR engine depends on ambient lighting and background and usually works well in the white background and brightly illuminated places.

IV. System Evaluation and Experiments

A. Evaluation of Object Detection

Our model (SSDLite) is pretrained on the Image-Net dataset for the image classification. It draws a bounding box on an

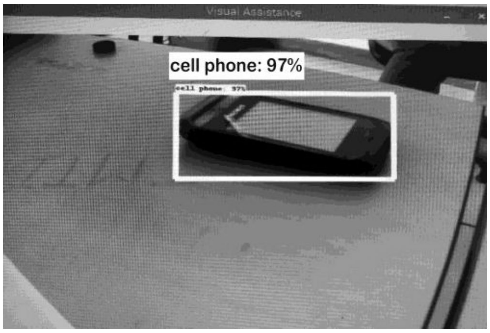


Fig. 7. Single Object Detection. The object detection algorithm can detect the cell phone with 97% confidence.

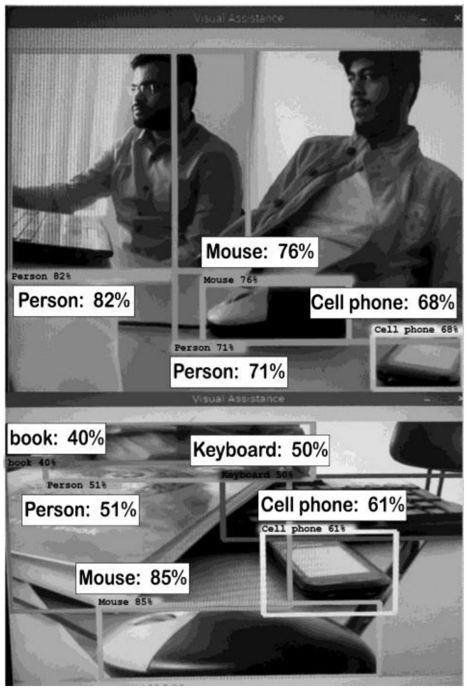


Fig. 8. Detecting multiple objects, with various confidence levels, from a single frame (white boxes are added for better visibility for readers).

object and tries to predict the object type based on the trained data from the network. It directly predicts the probability that each class is present in each bounding box using the softmax activation function and cross entropy loss function. The model also has a background object class when it is classifying different objects. However, there can be a large number of bounding boxes detected in one frame with only background classes. To avoid this problem, the model uses hard negative mining to sample negative predictions or downsampling the convolutional feature maps to filter out the extra bounding boxes.

Fig. 7 shows the detection of a single object from a video stream. Although most part of the image contains the background, the model is still able to filter out other bounding boxes and detect the desired object in the frame, with 97% confidence. The device can also detect multiple objects, with different confidence levels, from one video frame, as shown in Fig. 8. Our model can easily identify up to four or five objects, simultaneously, from a single video frame. The confidence level indicates the percentage of times the system can detect an object without any failure.

TABLE I
PERFORMANCE OF SINGLE AND MULTIPLE OBJECT DETECTION

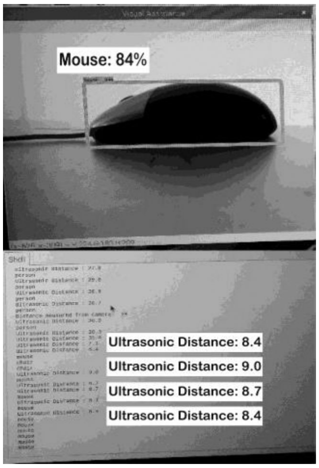
Test Cases	Actual Object (s)	Predicted Object (s)	Failure Case (s)
1	Person	Person	None
2	Mouse	Mouse	None
3	Person	Person	None
4	Notebook	Notebook	None
5	Cell Phone	Cell Phone	None
6	Person, Chair, Mouse	Person, Chair, Mouse	None
7	Cell Phone, Notebook	Cell Phone, Laptop	Laptop
8	Notebook, Person	Notebook, Person	None
9	Pen, Mouse, Keyboard	Pen, Mouse, Keyboard	None
10	Bottle, Cell Phone	Bottle, Cell Phone	None
11	Clock, Backpack	Clock, Backpack	None
12	Bottle, Cup, Chair	Bottle, Cup, Chair	None
13	Chair, Person	Chair, Person	None
14	Laptop, Bed, Cup	Laptop, Bed, Cup	Bed
15	Person, Chair, Cup	Person, Chair, Cup	None
16	Person, Bench, Chair	Person, Bench, Chair	None
17	Cell phone, Cup	Cell phone, Cup	None
18	Knife, Cell phone	Knife, Cell phone	None
19	Knife, Spoon, Banana	Knife, Spoon, Banana	None
20	Apple, Orange, Banana, Bowl, Knife, Spoon	Banana, Bowl, Knife, Spoon	Orange, Apple
21	Bicycle, Person, Chair	Bicycle, Person, Chair	None
22	Person, Bicycle, Car, Motorbike	Person, Bicycle, Car, Motorbike	None

Table I summarizes the results from single and multiple object detection, for 22 unique cases, consisting of either a single item or a combination of items, commonly found in indoor and outdoor setups. The system can identify single items with near 100% accuracy with zero failure cases. Where multiple objects are in the frame, the proposed system can recognize each known object within the view. For any object situated in the range of 15–20 m from the user, the object can be recognized with at least 80% accuracy. The camera identifies objects based on their ground truth values (in %), as shown in Figs. 7 and 8. However, to make the device more reliable, the ultrasonic sensor is also used to measure the distance between the object and the user. Whenever there are multiple objects, in front of the user, the system will generate feedback for the object, which is closest to the user. An object with a higher ground truth value has a higher priority. The pretrained model, however, is subject to failure due to variation in the shape and color of the object as well as changes in ambient lighting conditions.

B. *Evaluation of Distance Measurement*

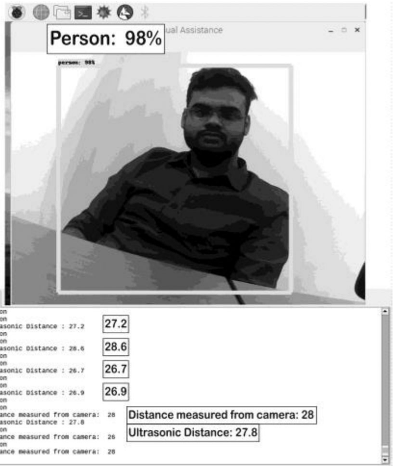
Fig. 9 shows the device measuring the distance between a computer mouse and the blind person using the ultrasonic sensor. If the distance measured from the sensor is less than 40 cm, the user will get a voice alert saying that the object is within 40 cm. The sensor can measure distances within a range of 2–120 cm by sonar waves.

Fig. 10 demonstrates the case where the combination of camera and ultrasonic sensor is used to identify a person's face



```
Ultrasonic Distance : 27.8
person
Ultrasonic Distance : 29.6
person
Ultrasonic Distance : 26.9
person
Ultrasonic Distance : 26.7
person
Distance measured from camera:  28
Ultrasonic Distance : 30.3
person
Ultrasonic Distance : 30.3
Ultrasonic Distance : 35.6
Ultrasonic Distance : 7.1
Ultrasonic Distance : 8.4
mouse
chair
chair
Ultrasonic Distance : 9.0
mouse
Ultrasonic Distance : 8.3
Ultrasonic Distance : 8.7
mouse
Ultrasonic Distance : 8.7
mouse
Ultrasonic Distance : 8.4
mouse
mouse
mouse
mouse
mouse
mouse
```

Fig. 9. Measuring the distance of a mouse from the prototype device using ultrasonic sensor.



```
person
person
Ultrasonic Distance : 27.2
person
person
Ultrasonic Distance : 28.6
person
person
Ultrasonic Distance : 26.7
person
person
Ultrasonic Distance : 26.9
person
person
Distance measured from camera:  28
Ultrasonic Distance : 27.8
person
Distance measured from camera:  26
person
Distance measured from camera:  28
```

Fig. 10. Face detection and distance measurement from a single video frame.

and determine how far the person is from the blind user. The integration of the camera with the ultrasonic sensor, therefore, allows simultaneous object detection and distance measurement, which adds novelty to our proposed design. We have used the Haar cascade algorithm [42] to detect face from a single video frame. It can also be modified and used for other objects. The bounding boxes, which appear while recognizing an object, consist of a rectangle. The width w, height h, and the coordinates of the rectangular box (x_0, y_0) can be adjusted as required.

Fig. 11 demonstrates how the distance between the object and the blind user can be simultaneously measured by both the camera and the ultrasonic sensor. The dotted line (6 m) represents the distance measured by the camera and the solid line (5.6 m) represents the distance calculated from the ultrasonic sensor. Width w and height h of the bounding box are defined in the .xml file with feature vectors, and they vary depending on the distance between the camera and the object. In addition to the camera, the use of the ultrasonic sensor makes object detection more reliable. The following equation, which can be derived by considering the formation of image, as light passes through the camera lens [43], is used to calculate the distance between the object and user:

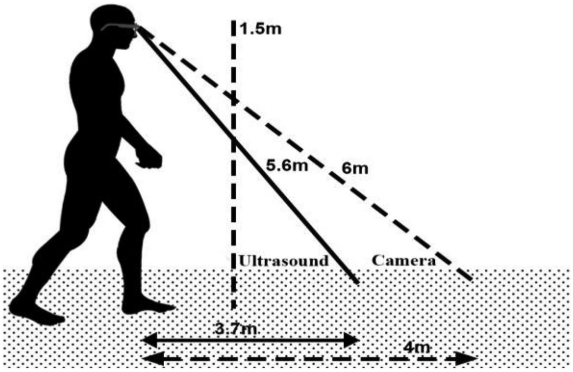


Fig. 11. Demonstration of the distance measurement using camera and ultrasonic sensor.

TABLE II
DISTANCE MEASUREMENT BETWEEN OBJECT AND USER

	Distance (in cm)		
Test Cases	Actual	Ultrasonic Sensor	Camera
1	30.5	30.2	31
2	15.6	15.6	17
3	23.1	23.2	23
4	42.4	42.1	41
5	0.9	1.1	1

$$\text{distance (inches)} = \frac{(2 \times 3.14 \times 180)}{(w + h \times 360) \times 1000 + 3}. \quad (1)$$

The actual distance between the object and the user is measured by a measuring tape and compared with that measured by the camera and the ultrasonic sensor. Since the camera can detect a person's face, the object used in this case is a human face, as shown in Fig. 10. Table II summarizes the results. The distance measured by ultrasonic sensors is more accurate than that measured by the camera. Also, the ultrasonic sensor can respond in real time so that it can be used to measure the distance between the blind user and a moving object. The camera, with a higher processing power and more fps, has a shorter response time. Although the camera takes slightly more time to process, both camera and ultrasonic sensors can generate feedback at the same time.

C. Evaluation of Reading Assistant

The integrated reading assistant in our prototype is tested under different ambient lighting conditions for various combinations of text size, font, color, and background. The OCR engine performs better in an environment with more light as it can easily extract the text from the captured image. While comparing text with different colored background, it has been shown that a well-illuminated background yields better performance for the reading assistant. As given in Table III, the performance of the reading assistant is tested under three different illuminations: bright, slightly dark, and dark, using the green and black-colored

TABLE III
PERFORMANCE OF THE READING ASSISTANT

Test Cases	Text	Text Color	Paper Color (Background)	Ambient Lighting	Performance
1	What can I do for you?	Black	White	Bright	Reads Accurately
2	What can I do for you?	Black	White	Slight Dark	Does not Read Accurately
3	What can I do for you?	Black	White	Dark	Does not Read Accurately
4	I am doing well.	Green	White	Bright	Reads accurately
5	I am doing well.	Green	White	Slight Dark	Does not Read Accurately
6	I am doing well.	Green	White	Dark	Does not Read Accurately

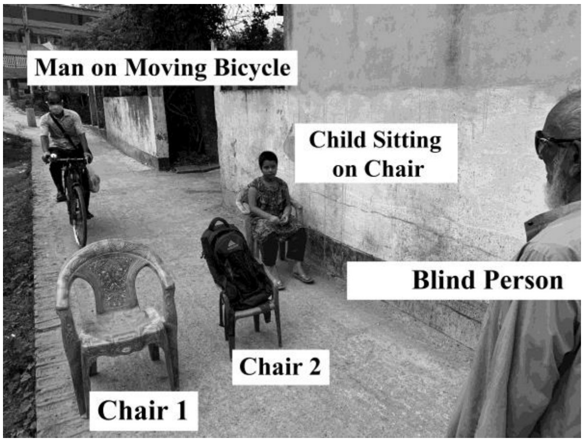


Fig. 12. Typical outdoor test environment.

texts, written on white pages. When the text color is black, the device performed accurately in bright and even in a slightly dark environment but under the dark condition, it failed to read the full sentence. For the green-colored text, the reading assistant had no issues in the brightly lit environment but failed to perform accurately in slightly dark and dark conditions.

D. Experimental Setup

The usability and performance of the prototype device is primarily tested in controlled indoor settings that mimic real-life scenarios. Although the proposed device functioned well in a typical outdoor setting, as shown in Fig. 12, the systematic study and conclusions, discussed in the following sections, are based on the indoor setup only.

A total of 60 completely blind individuals (male: 30 and female: 30) volunteered to participate in the controlled experiments. The influence of gender or age on the proposed system is beyond the scope of our current work and has, therefore, not been investigated here. However, since the gender-based blindness studies [44], [45] have shown blindness to be more prevalent among women than in men, it is important to have female blind users represented in significant numbers, in the testing and evaluation of any visual aid. Dividing 60 human samples into 30 males and 30 females to study separately, could, therefore, prove useful to conduct the gender-based evaluation study of the proposed system in future endeavors. A short training session, over a period of 2 hours, is conducted to familiarize the blind participants with the prototype device. During the training, the evaluation and scoring criterion were discussed in detail.

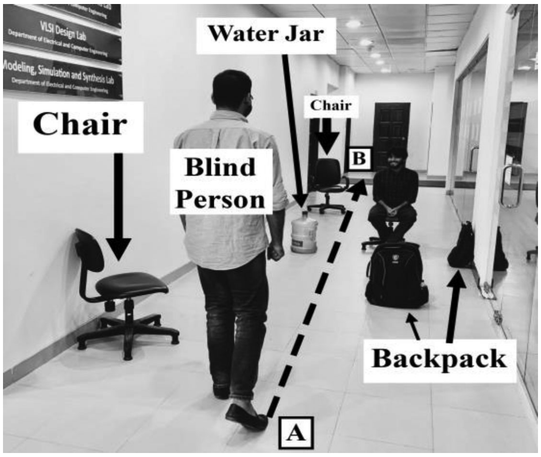


Fig. 13. Testing the prototype in an indoor setting.

The indoor environment, as shown in Fig. 13, consisted of six stationary obstacles of different heights and a moving person (not shown in Fig. 13). The position of the stationary objects was shuffled to create ten different indoor test setups, which were assigned, in random, to each user. A blind individual walks from point *A* to point *B*, along the path *AB* ($\sim$15 m in length), first with our proposed blind assistant, mounted on a pair of eyeglasses, and then with a traditional white cane. For both the device and the white cane, the time taken to complete the walk was recorded for each participant. Based on the time, the corresponding velocity for each participant is calculated. The results from the indoor setting, as shown in Fig. 13, are summarized and discussed in Section V.

E. Assessment Criterion

Blind participants were instructed to rate the device based on its comfort level or ease-of-use, mobility, and preference compared with the more commonly used traditional white cane. Ratings were done on a scale of 0–5 and the user experiences for comfortability, mobility, and preference over the white cane are divided into the following three categories based on the scores:

1) worst (score: 0–2);
2) moderate (score: 3);
3) good (score: 4 and 5).

The preferability scores also refer to the likelihood that the user would recommend the device to someone else. For example, a score of 3 for preferability means that the user is only slightly impressed with the overall performance of the device, while a score of 1 means that the blind person highly discourages the use of the device. The accuracy of the reading assistant was also scored on a scale of 0–5, with 0 being the least accurate and 5 being the most. The total score, from each user, is calculated by summing the individual scores for comfort, mobility, preferability, and accuracy of the reading assistant. In the best-case scenario, each category gets a score of 5 with a total score of

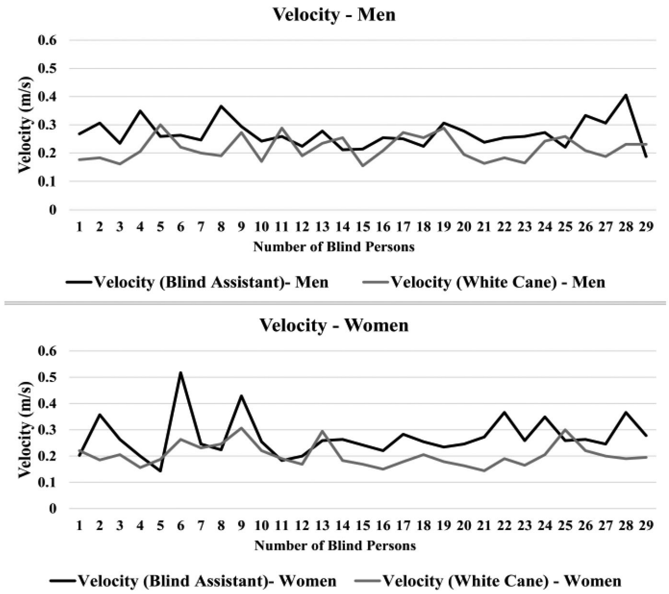


Fig. 14. Velocity of blind participants walking from point *A* to *B* in Fig. 13.

TABLE IV
AVERAGE VELOCITY OF BLIND PARTICIPANTS

	Blind Assistant	White Cane
Gender	Avg. Velocity (m/s)	Avg. Velocity (m/s)
Male	0.2627	0.2172
Female	0.2690	0.2144

TABLE V
PARAMETERS AND VALUES USED FOR *T*-TEST

	Blind Assistant	White Cane
Mean Velocity (m/s)	0.2659	0.2158
Standard Deviation (m/s)	0.0650	0.0439
Sample Size	60	60

20. Depending on the total score, the proposed blind assistant is labeled as "not helpful (total score: 0–8)," "helpful (total score: 9–15)," and "very helpful (total score: 16–20)." These labels were set after an extensive discussion with the blind participants prior to conducting the experiments. Almost all the blind users were participating in such a study for the first time with no prior experience of using any form of ETA. Therefore, it was necessary to set a scoring and evaluation criterion that could be easily adopted without the need for advanced training and extensive guidelines.

V. RESULTS AND DISCUSSION

Fig. 14 plots the velocity at which each blind user completes a walk from point *A* to point *B*, as shown in Fig. 13. For each user, the speed achieved using the blind assistant and the white cane is plotted. The plots for male and female users are shown separately. Table IV lists the average velocity for 30 male and 30 female participants. It is evident from the table that, on an average, the blind assistant provides slightly faster navigation than the white cane, for both the genders. To compare the performances between our proposed blind assistant and the white cane, a *t*-test is performed with a sample size of 60, using the following statistics:

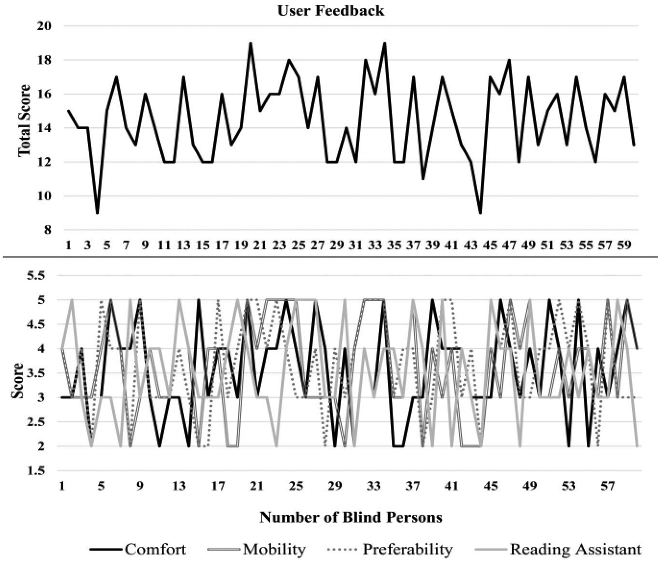


Fig. 15. User rating for the proposed device tested in the indoor setup of Fig. 13.

$$t = \frac{|\bar{x}_b - \bar{x}_w|}{\sqrt{\frac{s_b^2}{n_b} + \frac{s_w^2}{n_w}}} \tag{2}$$

where x_b, s_b, and n_b are the mean, standard deviation, and sample size, respectively, for the experiment with the blind assistant. The corresponding values for the white cane are denoted by x_w, s_w, and n_w. Table V lists the values used in the *t*-test.

With a *t*-value equal to 4.9411, the two-tailed *P* value is less than 0.0001. Therefore, by the conventional criteria and at 95% confidence interval, the difference in velocity between the blind assistant and white cane can be considered statistically significant.

The user ratings are plotted in Fig. 15, which shows the individual scores for comfort, mobility, preference, and accuracy of the reading assistant, on a scale of 0–5, for each of the 60 users. In addition, the total score, rated on a scale of 0–20, is also shown. The average of all scores is 14.5, which deems our proposed device as "helpful" based on the criterion defined in Section IV-E. Since we only used a prototype to conduct the experiments, the comfort level was slightly compromised. However, the mobility and preference of the proposed device over the white cane gained high scores. The pretrained model, which was used, could be retrained with more objects for better performance. The reading assistant performed well under brightly illuminated settings. One major limitation of the reading assistant, as pointed out by the users, is that it was unable to read texts containing tables and pictures.

A cost analysis was done with similar state-of-the-art assistive navigation devices. Table VI compares the cost of our blind

TABLE VI
COST OF PROPOSED DEVICE VERSUS EXISTING VISUAL AIDS

Device	Estimated Cost (USD)
Our Proposed Device	68
Lan *et al.* [26]	240
Jiang *et al.* [17]	97
Rajesh *et al.* [46]	70
White Cane	25

assistant with some of the existing platforms. The total cost of making the proposed device is roughly US $68, whereas some existing devices, with a similar performance, appear more expensive. The service dogs, another viable alternative, can cost up to US $4000 and require high maintenance. Although the white canes are cheaper, they are unable to detect moving objects and do not include a reading assistant.

VI. CONCLUSION

This research article introduces a novel visual aid system, in the form of a pair of eyeglasses, for the completely blind. The key features of the proposed device include the following.

1) The hands free, wearable, low power, low cost, and compact design for indoor and outdoor navigation.
2) The complex algorithm processing using the low-end processing power of Raspberry Pi 3 Model B+.
3) Dual capabilities for object detection and distance measurement using a combination of camera and ultrasound sensors.
4) Integrated reading assistant, offering image-to-text conversion capabilities, enabling the blind to read texts from any document.

A detailed discussion, on the software and hardware aspects of the proposed blind assistant, has been given. A total of 60 completely blind users have rated the performance of the device in well-controlled indoor settings that represent real-world situations. Although the current setup lacks advanced functions, such as wet-floor and staircases detection or the use of GPS and mobile communication module, the flexibility in the design leaves room for future improvements and enhancements. In addition, with the advanced machine learning algorithms and a more improved user interface, the system can further be developed and tested in a more complex outdoor environment.

REFERENCES

[1] Blindness and vision impairment, World Health Organization, Geneva, Switzerland, Oct. 2019. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment

[2] J. Bai, S. Lian, Z. Liu, K. Wang, and D. Liu, "Virtual-blind-road following-based wearable navigation device for blind people," *IEEE Trans. Consum. Electron.*, vol. 64, no. 1, pp. 136–143, Feb. 2018.

[3] B. Li *et al.*, "Vision-based mobile indoor assistive navigation aid for blind people," *IEEE Trans. Mobile Comput.*, vol. 18, no. 3, pp. 702–714, Mar. 2019.

[4] J. Xiao, S. L. Joseph, X. Zhang, B. Li, X. Li, and J. Zhang, "An assistive navigation framework for the visually impaired," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 5, pp. 635–640, Oct. 2015.

[5] A. Karmel, A. Sharma, M. Pandya, and D. Garg, "IoT based assistive device for deaf, dumb and blind people," *Procedia Comput. Sci.*, vol. 165, pp. 259–269, Nov. 2019.

[6] C. Ye and X. Qian, "3-D object recognition of a robotic navigation aid for the visually impaired," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 2, pp. 441–450, Feb. 2018.

[7] Y. Liu, N. R. B. Stiles, and M. Meister, "Augmented reality powers a cognitive assistant for the blind," *eLife*, vol. 7, Nov. 2018, Art. no. e37841.

[8] A. Adebiyi *et al.*, "Assessment of feedback modalities for wearable visual aids in blind mobility," *PLoS One*, vol. 12, no. 2, Feb. 2017, Art. no. e0170531.

[9] J. Bai, S. Lian, Z. Liu, K. Wang, and D. Liu, "Smart guiding glasses for visually impaired people in indoor environment," *IEEE Trans. Consum. Electron.*, vol. 63, no. 3, pp. 258–266, Aug. 2017.

[10] D. Dakopoulos and N. G. Bourbakis, "Wearable obstacle avoidance electronic travel aids for blind: A survey," *IEEE Trans. Syst. Man, Cybern. Part C Appl. Rev.*, vol. 40, no. 1, pp. 25–35, Jan. 2010.

[11] E. E. Pissaloux, R. Velazquez, and F. Maingreaud, "A new framework for cognitive mobility of visually impaired users in using tactile device," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 6, pp. 1040–1051, Dec. 2017.

[12] K. Patil, Q. Jawadwala, and F. C. Shu, "Design and construction of electronic aid for visually impaired people," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 2, pp. 172–182, Apr. 2018.

[13] R. K. Katzschmann, B. Araki, and D. Rus, "Safe local navigation for visually impaired users with a time-of-flight and haptic feedback device," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 3, pp. 583–593, Mar. 2018.

[14] J. Villanueva and R. Farcy, "Optical device indicating a safe free path to blind people," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 1, pp. 170–177, Jan. 2012.

[15] X. Yang, S. Yuan, and Y. Tian, "Assistive clothing pattern recognition for visually impaired people," *IEEE Trans. Human-Mach. Syst.*, vol. 44, no. 2, pp. 234–243, Apr. 2014.

[16] S. L. Joseph *et al.*, "Being aware of the world: Toward using social media to support the blind with navigation," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 3, pp. 399–405, Jun. 2015.

[17] B. Jiang, J. Yang, Z. Lv, and H. Song, "Wearable vision assistance system based on binocular sensors for visually impaired users," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1375–1383, Apr. 2019.

[18] L. Tepelea, I. Buciu, C. Grava, I. Gavrilut, and A. Gacsadi, "A vision module for visually impaired people by using raspberry PI platform," in *Proc.15th Int. Conf. Eng. Modern Electr. Syst. (EMES)*, Oradea, Romania, 2019, pp. 209–212.

[19] L. Dunai, G. Peris-Fajarnés, E. Lluna, and B. Defez, "Sensory navigation device for blind people," *J. Navig.*, vol. 66, no. 3, pp. 349–362, May 2013.

[20] V.-N. Hoang, T.-H. Nguyen, T.-L. Le, T.-H. Tran, T.-P. Vuong, and N. Vuillerme, "Obstacle detection and warning system for visually impaired people based on electrode matrix and mobile kinect," *Vietnam J. Comput. Sci.*, vol. 4, no. 2, pp. 71–83, Jul. 2016.

[21] C. I. Patel, A. Patel, and D. Patel, "Optical character recognition by open source OCR tool Tesseract: A case study," *Int. J. Comput. Appl.*, vol. 55, no. 10, pp. 50–56, Oct. 2012.

[22] A. Chalamandaris, S. Karabetsos, P. Tsiakoulis, and S. Raptis, "A unit selection text-to-speech synthesis system optimized for use with screen readers," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1890–1897, Aug. 2010.

[23] R. Keefer, Y. Liu, and N. Bourbakis, "The development and evaluation of an eyes-free interaction model for mobile reading devices," *IEEE Trans. Human-Mach. Syst.*, vol. 43, no. 1, pp. 76–91, Jan. 2013.

[24] B. Andò, S. Baglio, V. Marletta, and A. Valastro, "A haptic solution to assist visually impaired in mobility tasks," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 5, pp. 641–646, Oct. 2015.

[25] V. V. Meshram, K. Patil, V. A. Meshram, and F. C. Shu, "An astute assistive device for mobility and object recognition for visually impaired people," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 5, pp. 449–460, Oct. 2019.

[26] F. Lan, G. Zhai, and W. Lin, "Lightweight smart glass system with audio aid for visually impaired people," in *Proc. IEEE Region 10 Conf.*, Macao, China, 2015, pp. 1–4.

[27] M. M. Islam, M. S. Sadi, K. Z. Zamli, and M. M. Ahmed, "Developing walking assistants for visually impaired people: A review," *IEEE Sens. J.*, vol. 19, no. 8, pp. 2814–2828, Apr. 2019.

[28] T-Y Lin *et al.*, "Microsoft COCO: Common objects in context," Feb. 2015. [Online]. Available: https://arxiv.org/abs/1405.0312

[29] J. Han *et al.*, "Representing and retrieving video shots in human-centric brain imaging space," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2723–2736, Jul. 2013.

[30] J. Han, K. N. Ngan, M. Li, and H. J. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 141–145, Jan. 2006.

[31] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.

[32] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[33] Y. Yang, Q. Zhang, P. Wang, X. Hu, and N. Wu, "Moving object detection for dynamic background scenes based on spatiotemporal model," *Adv. Multimedia*, vol. 2017, Jun. 2017, Art. no. 5179013.

[34] Q. Xie, O. Remil, Y. Guo, M. Wang, M. Wei, and J. Wang, "Object detection and tracking under occlusion for object-level RGB-D video segmentation," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 580–592, Mar. 2018.

[35] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[36] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vision*, vol. 9905, Sep. 2016, pp. 21–37.

[37] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 4510–4520.

[38] P. Hurtik, V. Molek, J. Hula, M. Vajgl, P. Vlasanek, and T. Nejezchleba, "Poly-YOLO: Higher speed, more precise detection and instance segmentation for YOLOv3," May 2020. [Online]. Available: http://arxiv.org/abs/2005.13243

[39] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," May 2020. [Online]. Available: http://arxiv.org/abs/2005.12872

[40] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vision*, Seoul, South Korea, 2019, pp. 4510–4520.

[41] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT++: Better real-time instance segmentation," Dec. 2019. [Online]. Available: https://arxiv.org/abs/1912.06218

[42] R. Padilla, C. C. Filho, and M. Costa, "Evaluation of Haar cascade classifiers designed for face detection," *Int. J. Comput., Elect., Autom., Control Inf. Eng.*, vol. 6, no. 4, pp. 466–469, Apr. 2012.

[43] L. Xiaoming, Q. Tian, C. Wanchun, and Y. Xingliang, "Real-time distance measurement using a modified camera," in *Proc. IEEE Sensors Appl. Symp.*, Limerick, Ireland, 2010, pp. 54–58.

[44] L. Doyal and R. G. Das-Bhaumik, "Sex, gender and blindness: A new framework for equity," *BMJ Open Ophthalmol.*, vol. 3, no. 1, Sep. 2018, Art. no. e000135.

[45] M. Prasad, S. Malhotra, M. Kalaivani, P. Vashist, and S. K. Gupta, "Gender differences in blindness, cataract blindness and cataract surgical coverage in India: A systematic review and meta-analysis," *Brit. J. Ophthalmol.*, vol. 104, no. 2, pp. 220–224, Jan. 2020.

[46] M. Rajesh *et al.*, "Text recognition and face detection aid for visually impaired person using raspberry PI," in *Proc. Int. Conf. Circuit, Power Comput. Technol.*, Kollam, India, 2017, pp. 1–5.