Advancing Accessibility: An Artificial Intelligence Framework for Obstacle Detection and Navigation Assistance for the Visually Impaired

Liliek Triyono[1,2] *, Rahmat Gernowo*[1]*, Prayitno*[2]*, Saifur Rohman Cholil*[1]*, Idhawati Hestiningsih*[2]*, Wiktasari*[2] *,Sirli Fahriah*[2]

[1]Doctoral Program of Information System School of Postgraduate Studies, Diponegoro University, Semarang, Indonesia
[2]Department of Electrical Engineering, Politeknik Negeri Semarang, Semarang 50275, Indonesia

Abstract. The white cane has long been a fundamental tool for individuals with visual impairments, aiding in surface detection and obstacle identification. However, its limitations in detecting moving objects and distant obstacles pose significant safety risks, particularly in congested areas and busy streets. While service animals offer an alternative, they come with training challenges and high costs. To address these limitations and enhance safety, this paper proposes a comprehensive collision detection and prevention system. The proposed system integrates cutting-edge technologies, including image processing, deep learning, Internet of Things (IoT), cloud computing, and audio production devices. By combining these technologies with the white cane, the system offers a sophisticated navigation option for the visually impaired, effectively detecting and preventing potential collisions. In busy environtment scenarios, the system proves its effectiveness by complementing the white cane's use, overcoming its inherent limitations, and significantly improving navigation capabilities. Through this innovative approach, blind individuals gain enhanced situational awareness, empowering them to navigate diverse environments with increased confidence and safety. By mitigating the drawbacks of the white cane, the proposed system provides a comprehensive and cost-effective solution to enhance the mobility and safety of the visually impaired. This research contributes to the advancement of assistive technologies, offering a valuable resource for researchers, policymakers, and practitioners in the field of accessibility and inclusive design.

1 Introduction

People with visual impairment (PVI) struggle with daily tasks such as recognizing objects, navigating indoors or outdoors, and averting obstacles [1], [2] and shopping [3]. People with visual impairment (PVI) struggle with daily tasks such as navigating indoors or outdoors, recognizing objects, and averting obstacles [4], [5]. The use of global positioning systems (GPS) to address outdoor navigation issues. GPS employs geostationary satellite signals with a precision of several meters, which is adequate for outdoor navigation. However, indoor navigation remains a significant problem requiring precise and reliable solutions. For instance, GPS cannot be used to automate underground and nearby roofs, walls, and other objects in lofty structures [6], [7].

Utilizing Computer Vision (CV), research has enabled PVIs to navigate indoors and recognize objects [8]–[10]. Typical CV navigation systems employ uniquely affixed tags, such as Augmented Reality (AR) identifiers, to assist with indoor navigation and object recognition[11]–[13]. It consists of affixed tags, a database to store tag information, a camera to capture real-time images, a processing unit to implement the techniques used, and two-way communication between the system and PVI to receive input [14], [15]. In many real-world scenarios, however, the markers cannot be identified due to blurred motion or deformation, poor illumination conditions, or a distance that is excessive from the camera [16].

In recent years, Deep Learning algorithms have been used to enhance object detection software in the CV field. Convolutional neural networks augment the network level, thereby enhancing its detection capabilities. There are two types of deep learning for detection: two-stage and single-stage.

Region-based Convolutional Neural Networks (R-CNNs) [17], Fast R-CNNs [18], and Faster R-CNNs [19] are two-stage algorithms that use the Regional Proposal network to generate regions of interest in the first stage and deploy the Regional Proposal network to the pipeline for object classification and regression in the second stage. Using the Regional Proposal algorithm, R-CNN predicts the location of objects.

Each candidate region's features are extracted, inputted into CNNs, and then evaluated using Support Vector Machines (SVM). R-CNN enhances the accuracy of target detection, but it has a very low efficiency. The R-CNN detects areas of interest in images using the Regional Proposal Network method. Then, a classifier is used to classify these bounding rectangles, also known as areas of interest. The speedier R-CNN enhances detection accuracy, but its detection speed is too sluggish for high-resolution real-time applications. In the field of object detection, single-stage detectors such as You Only Look Once (YOLO) [20] and Single Shot Detector (SSD) [21] have been developed to increase the detection efficacy, making them suitable for real-time applications. These detectors approach object detection as a straightforward regression problem in an effort to simplify the process. They use input images to simultaneously predict class probabilities and bounding box coordinates. Despite the fact that these models obtain a slightly lower level of accuracy than two-stage object detectors [22], they make up for it by delivering substantially speedier inference times.

Specifically, YOLO is a convolutional neural network (CNN) designed to facilitate rapid, precise, real-time object detection [20]. It employs a unified architecture that accurately anticipates both the object categories and their spatial locations within an image. In an effort to establish a balance between precision and performance, the YOLO model has undergone numerous iterations and enhancements over time.

YOLO v1, the initial iteration of YOLO, for instance, divides the input image into a 7 x 7 grid of cells. Each cell contains two prediction boxes, and the final results are derived directly using the Intersection over Union (IOU) metric of the prediction boxes [23]. This method enables YOLO v1 to detect objects within an image efficiently. YOLO v2 goes one step further than its predecessor by partitioning the image into a grid of 13 x 13 cells. The final prediction boxes are constructed based on the regression results obtained from these anchor frames [24]; each cell is associated with five predefined anchor frames.

By integrating anchor box regression, YOLO v2 improves the precision of object detection while preserving real-time processing capabilities. In conclusion, single-stage detectors such as YOLO and SSD have revolutionized the field of object detection by refining and optimizing the process for real-time applications. Particularly, YOLO has emerged as a CNN-based method that combines quickness and accuracy. With subsequent iterations, such as YOLO v1 and YOLO v2, the precision of object detection has been improved, enabling more accurate and efficient real-time detection and localization of objects within images.

YOLOv2, an evolution of the original YOLO model, introduces a number of enhancements to increase object detection performance. Utilizing a feature map with a higher resolution aids the network in detecting objects of differing sizes within an image. In addition, each convolutional layer in YOLOv2 is subjected to additional batch normalization, which contributes to enhanced training stability and accelerated convergence [25]. YOLOv2 uses anchor boxes, which serve as predefined reference boxes for predicting object locations and sizes, to estimate bounding boxes. Incorporating multi-scale detection capabilities, YOLOv3 builds upon the strengths of its predecessors by introducing additional advancements. This requires the incorporation of a more potent feature extractor network as well as several modifications to the training loss function. These enhancements allow YOLOv3 to detect both large and small targets in an image effectively [26]. In addition, YOLOv3 introduces the concept of three distinct scale prediction networks, which play a crucial role in detecting objects of differing dimensions, such as medium-scale objects and small objects nested within larger objects. When an object is detected, prediction networks are subjected to bounding box regression, which yields the final prediction box. Each prediction network is associated with three predetermined anchor boxes, allowing for the precise localization of objects at various scales.

In YOLOv4, the primary components of the detection network are CSPDarknet53, SPP (Spatial Pyramid Pooling), and PANet (Path Aggregation Network). The CSPDarknet53 network comprises the BasicConv convolutional network module and the CSP residual network module, and functions as the backbone for feature extraction. The SPP module consists of the BasicConv and maxpooling convolutional network modules, whereas PANet incorporates features from multiple layers to capture richer semantic data via upsampling and downsampling operations. These architectural components contribute synergistically to YOLOv4's exceptional detection capabilities [27].

1.1 Related Works

In recent years, significant progress has been made in the application of deep learning and CVs to biomedical images [28], cancer prediction [29], object detection [30], and object recognition [31]. Smartphones are now necessary due to their comprehensive abilities, such as processing power, built-in cameras, and multiple sensors. This developing and growing smartphone technology and capability allows researchers to create new PVI object identification applications [8], [32] and safely traverse interior environments [33], [34].

Shrinivas et al. estimate a head-on obstacle using a body-mounted camera and gyro sensors as well as a generic laptop embedded with an OMAP-3 compact device. The proposed system represents an important step toward the creation of a vision assistance for visually impaired and blind individuals [35].

To aid visually impaired individuals, Van-Nam et al. propose a method involving a sequential process of obstacle detection, localization, and auditory communication. The system consists of two essential components: environment information acquisition and analysis, and data representation. The first component employs a mobile Kinect device to collect environmental data and analyze the presence of potential obstacles for visually impaired individuals. The second component concentrates on representing obstacle information using an electrode matrix, which enables the transmission of detected obstacle details in a user-friendly format [36].

Kiran et al. utilized a mechanical device to detect obstacles on the ground, irregular surfaces, gaps, and other dangers. The primary issue with the white cane is that users must be trained, and this device examines only a limited area in front of the user. It is incapable of detecting obstacles beyond its range and can only be detected through physical contact. This technology employs a camera for image capture and a microphone for audio capture, followed by a computer for processing the input and providing an immediate response based on the current frames [37].

Lee et al. [38] proposed a system for indoor navigation using markers and augmented reality. It performs hybrid localization using both marker images and smartphone Inertial Measurement Unit (IMU) data. First, an indoor map is constructed to record the locations of indoor locations. Then, markers are generated and printed for the locations that have been registered. The navigation system assists users in reaching their destinations successfully. As the markers are installed on the floor, however, they cannot be detected in a congested environment. In addition, it was incapable of identifying markers at great distances.

Mekhalfi et al. suggested a computervision-based navigation system [39]. It included a speech recognition module for receiving instructions and providing PVI with voice feedback. To calculate the distance between obstacles, a laser sensor was employed. PVI location was determined using a set of markers and an IMU sensor, and a path planning module was utilized to generate a safe path for the user to walk through. They captured the scene with a handheld camera and sent the images to the navigation or identification units.
However, the size and weight of the processing unit is a major issue because PVI cannot be worn for an extended period of time - a disadvantage when compared to utilizing a smartphone. The average processing time for recognition had to be reduced as well. Finally, obstacle detection sensors are expensive and not widely available to the general public.

Using a multi-label convolutional SVM, Bazi et al. suggested a navigation system to assist PVI in recognizing various objects in images [40]. It takes pictures and transmits them to a laptop processing unit using a portable camera installed on a thin shield worn by the user. In order to provide a fresh set of feature maps for object recognition, a group of linear SVMs were applied as a filter in each convolutional layer. The results are then once more input to a linear SVM classifier, which performs the classification operation. The processing unit's size and weight, however, are once again a significant issue for PVI. Additionally, it was unable to pick up markers at greater distances. This study's primary contribution is to propose a system and assess whether or not it is capable of conducting the intended function.

1.2 Motivation

In addition to the previously mentioned applications, object detection can have a transformative effect in resolving navigational difficulties for individuals with visual impairments. Visually impaired individuals may find it difficult to navigate unfamiliar environments because they rely heavily on auditory cues and the assistance of others. We can empower individuals with visual impairments to navigate independently and with greater confidence by leveraging the power of object detection technology illustrated in Figure 1.

2 System Description

The following list summarizes the key actions involved in utilizing a navigation system for indoor navigation: The building should be well-prepared for PVI by placing markers at the key interest spots, a map should be created to link these places, and navigation commands should be used to aid PVI travel from their current location and arrive at their goal successfully.

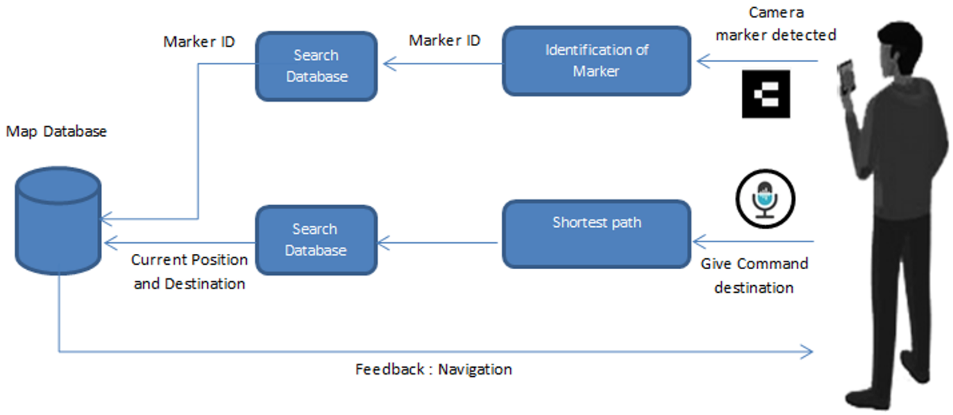

Figure 1. Proposed System for Detecting Markers for People Using Tiny-YOLOv4

2.1 Marker Selection

The following list summarizes the key actions involved in utilizing a navigation system for indoor navigation: The building should be well-prepared for PVI by placing markers at the key interest spots, a map should be created to link these places, and navigation commands should be used to aid PVI travel from their current location and arrive at their goal successfully [3]. Based on the available technology evaluation, the participants expressed support for the tag CV technique. There are several tags available, but the persegi marker is the most popular one since it offers eight matching digits, which are ideal for estimating camera posture [13].

To compare and identify QR codes and square identifiers, the researchers utilised two programmes. The application operates by opening the camera to take a live image feed, turning the picture to grayscale using OpenCV, and then using the right library to find and recognise the markers that are present. To detect and identify markers in the first application, researchers employed QR codes and an open source library called Zxing. The Aruco library, an open source library, and Aruco markers are used by researchers to find and identify markers in the second application [41].

2.2 Positioning

A sighted individual must make a map of the whole building for each floor before employing a navigation device. To find interesting locations like labs and lecture halls, they must walk about the structure. The marker is then produced and applied to the wall in the desired location. This marking later aids PVI's navigation inside the structure. The admin app is then utilized to scan each marker and enter the information into the Firebase Database. Identification markers, floor numbers, and names of interest points like "Ruang Kelas 202" are all included in the data. The building's floors are each subjected to this process again. Figure 2 shows a blueprint of the second floor with interest points marked with a red circle.

2.3 Navigation

Using an auditory interface, the navigation system was created to be simple to use for PVI users. The prototype program opens the camera to get a stream of frames and transforms them into grayscale images with a simple tap on any area of the screen.

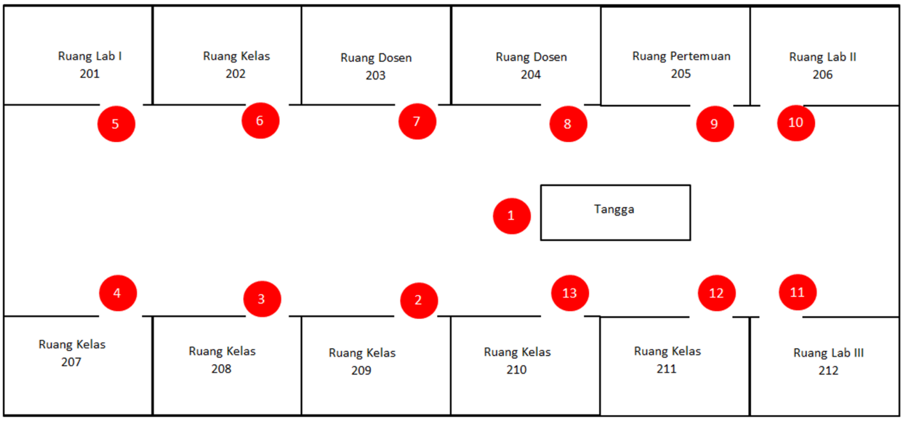

Figure 2. 2nd floor blueprint

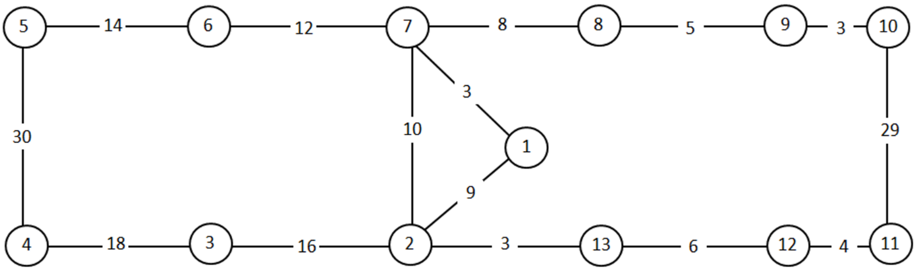

Figure 3. Graphics built for blueprints. The circle represents the destination point while the edge represents the available path and the number of steps in between

The PVI is instructed to move the smartphone left and right after opening the camera by an audio message, which uses a Text to Speech (TTS) module to look for any markers. When necessary, the PVI is provided with audible feedback via this module.

Text analysis and voice synthesis are the two core elements of a typical TTS paradigm. This part transforms numerical and abbreviated symbols into written language. The navigational directions are then translated into a voice that PVI can easily comprehend using speech synthesis. The Google TTS library is utilized in this module to transform text to speech [42]. If a marker is found, the system will utilize it as the beginning point and then issue a voice command to PVI to choose a destination location. The speech recognition API is used to translate voice commands sent by PVI to the navigation system into text using Natural Language Processing (NLP). The NLP algorithm offers a method for accurately translating voice commands to text. Researchers used the NLP algorithm from the Google API to create this article.

After then, PVI receives auditory feedback to determine whether or not their instruction was acknowledged. If the destination is not found, the system asks PVI to input it again. The prototype calculates the shortest path from the starting point to the destination using Dijkstra's algorithm and instructs PVI to begin walking in the shortest route. The returned path is a list of waypoints that PVI must traverse to reach the specified location [43]. PVI must adhere to navigation instructions to proceed from one location to another until it reaches its destination.

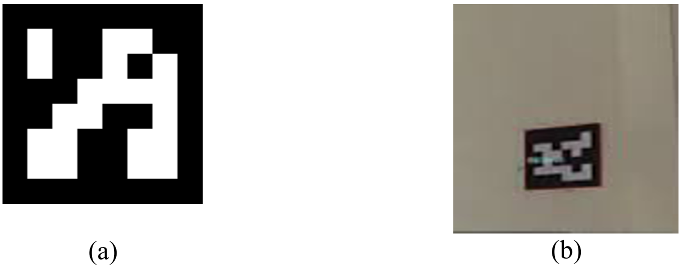

Figure 4. (a) Marker Aruco (b) Marker Aruco installed in wall

When PVI reaches any point on the graph by detecting a marker placed on the wall, the prototype issues a navigation command that guides it to the next point on the graph.

Researchers used the Aruco library to detect and calculate the distance between navigational markers and the camera. It depends on the size of the marker as seen in the captured image, so calibration of the camera is necessary beforehand. You can estimate the distance between the marker position and the camera using the Aruco library. If any marker is detected, a function in the library returns two vectors representing its position. The translation vector indicates the distance between the marker and the camera's coordinate system's origin. The orientation is described by the rotation vector. To minimize drifting errors, researchers also affixed markers to the walls so they could be seen from a variety of perspectives [44]. This library, however, cannot detect markers under extreme conditions. Therefore, researchers propose a deep learning model to address the issue.

Figure 5 depicts an illustration of this procedure. Consider that PVI is in front of point 7 and desires to travel to point 11. The PVI touches the screen, which opens the camera and maneuvers the phone in accordance with the voice command of the app. The system detects and designates point 7 as the beginning point, and provides verbal confirmation that it has been chosen. If multiple markers are detected simultaneously, the one closest to the phone will be designated as the beginning point based on the distance between the phone and the marker. In this example, PVI selects point 11 by verbally stating "Raung Lab III 212" as the destination. Then, the shortest path from point 7 to point 11 is calculated and returned as a list. PVI is tasked with following this list of concise points and proceeding from point 7 to points 2, 13, 12, and 11 in order to achieve the objective. From point 7, the researcher system provides PVI navigation feedback in order to reach point 2. To accomplish this successfully, PVI must detect wall markers as instructed. When a marker for point 2 is detected, the researcher's application sends PVI there and notifies it of the required distance. To improve the accuracy of the system, it compares the number of steps recorded in the database with the number of steps counted by the smartphone sensor as PVI walks from one marker to another. From point 2, the same procedure is repeated to lead PVI to the subsequent point, point 13. The route then continues to point 12 before reaching point 11, the final destination. When they reach their final destination, PVI receives notification that they have successfully arrived.

To ensure the accuracy of the navigation system, researchers evaluate all possible situations and conditions that PVI may encounter during navigation. An Android smartphone (Xiaomi Redmi Note 11) capable of recording video at 30 frames per second can be proposed for use in research. This means the camera takes 24 images per second and transmits them to the application for processing. If it fails to detect a marker in one frame, it is likely to do so in the next if the majority of images sent in one second contain essentially the same scene. If PVI discovers the marker once more, there are two potential outcomes.

If this marker is included in the list of destination points, the system continues to provide navigation instructions from this marker to the destination.

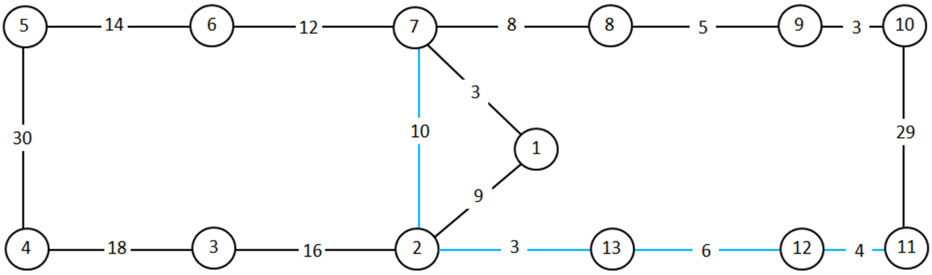

Figure 5. An example of the shortest path to the destination using a navigation system.

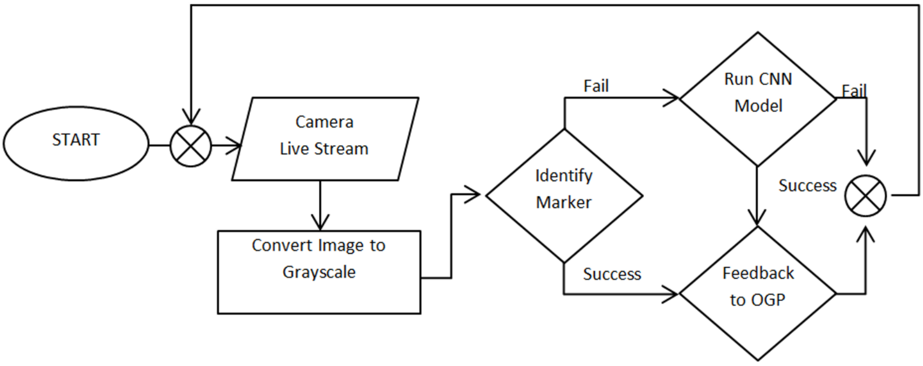

Figure 6. Marker detection process flow diagram

If this marker is not in the list, the system will search for a new shortest path between the new marker and the destination point. If PVI moves in a legal direction, it is likely that the camera will locate another marker because the current marker encompasses the majority of the building. If the camera fails to detect the marker for a specified period of time, say 30 seconds, the researcher's app notifies PVI that they are walking in the correct direction.

3 Object Detection

The preceding system relied on the utilization of advanced artificial intelligence algorithms and high-performance hardware to process data efficiently [35]–[37], [45]–[51]. However, a noteworthy drawback of this system was the requirement for users to bear the responsibility of carrying such equipment, consequently imposing an additional burden and contributing to the overall challenges faced in their daily lives [52]–[56].

3.1 Dataset

The YOLO (You Only Look Once) model, renowned for its real-time object detection capabilities, is evaluated using the COCO (Common Objects in Context) dataset [57], a widely recognized benchmark for evaluating object/obstacle detection performance. This dataset contains a variety of real-world scenarios and poses a formidable challenge to object recognition tasks.
The EfficientDet model, however, emerges as a compelling alternative in the pursuit of perpetually enhancing object detection efficiency. Notably, the EfficientDet model has

demonstrated its ability to outperform models of comparable scale across multiple benchmark datasets by achieving higher mean average precision (mAP) scores. Remarkably, this accomplishment is accomplished with fewer parameters and fewer computational resources. The practical implication of adopting the EfficientDet model is its accelerated performance on both GPU and CPU platforms, which provides a significant advantage over alternative object detection techniques. This efficiency improvement is especially beneficial when dealing with real-time applications or scenarios with limited computational resources. The EfficientDet model represents a significant advance in the field of object detection because it effectively combines superior performance with optimized computational efficiency. Its ability to achieve higher mAP scores while employing fewer parameters and reducing computational demands makes it an attractive option for practitioners and researchers, facilitating faster and more accurate object detection on a broad variety of hardware platforms.

3.2 Loss function

The three principal components of the YOLO v4 loss function are the confidence loss, the classification loss, and the regression loss. Cross-entropy loss is utilized for loss of confidence and loss of classification. For regression loss, CIOU loss function is utilized. Loss of classification is defined by the equation: (1).

$$Loss_{cls} = -\sum_{i=0}^{S^2} I_{i,j}^{obj} \sum_{c\in cls}\left[\hat{P}_i^j \log\left(P_i^j\right) + \left(1-\hat{P}_i^j\right)\log\left(1-P_i^j\right)\right] \tag{1}$$

The loss of confidence is defined in Eq. (2):

$$Loss_{conf} = -\sum_{i=0}^{S^2} \sum_{c\in cls}^{B} I_{i,j}^{obj}\left[\hat{F}_i^j \log\left(F_i^j\right) + \left(1-\hat{F}_i^j\right)\log\left(1-F_i^j\right)\right] - \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{noobj}\left[\hat{F}_i^j \log\left(F_i^j\right) + \left(1-\hat{F}_i^j\right)\log\left(1-F_i^j\right)\right] \tag{2}$$

In Eqs. (1) and (2), S2 represents 13 × 13, 26 × 26 and 52 × 52. B represents the number of prediction boxes. $\mathrm{I}_{\mathrm{i,j}}^{\mathrm{obj}}$ represents the 0 or 1 is the probability that the compartment contains an object at (i, j). λ_{noobj} represents the confidence loss weights for the bounding box. $\hat{F}_i^j$ represents the actual worth. $\hat{F}_i^j$equals 1 if the bounding box is responsible for predicting a particular object; otherwise, it equals 0. The P_i^j and $\hat{P}_i^j$ subtables represent, respectively, the predicted and actual values of the individual probabilities for a given object of the forecast, respectively.

3.3 Object Detection Methodology

In our methodology, we are employing cloud services and deep learning for the major parts because deep learning is the only technique that has the tools that most suit our purpose and has the potential to make the system perfect and further increase its functionality as the technology advances.

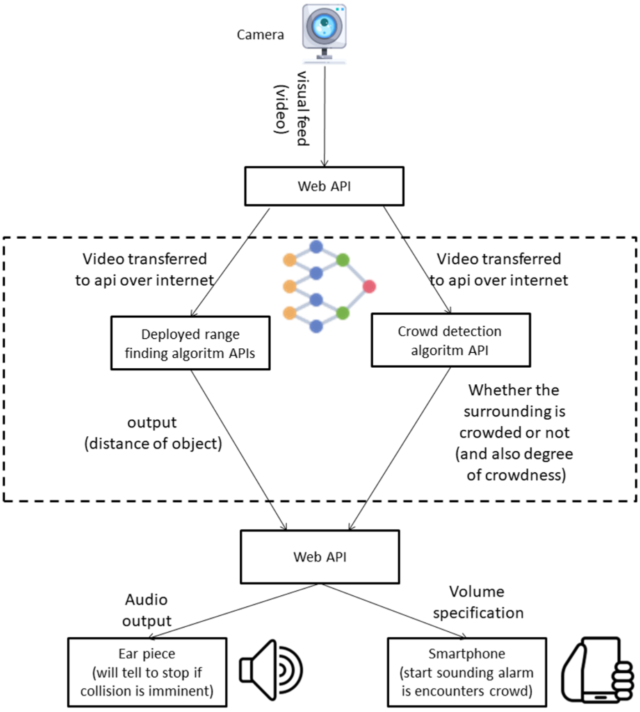

Figure7. Block diagram of the proposed object detection model

The majority of our methodology is based on cloud-based services and deep learning because deep learning is the sole method that has the tools we need and has a chance to make the system flawless and increase its capacity for use as technology advances. Cloud services offer a cost-effective solution for processing vast amounts of data within a short timeframe without the need to invest heavily in expensive hardware, such as high-end GPUs, which can be both financially burdensome and technically challenging to acquire and maintain. Additionally, accessing premium software for data processing can also be prohibitively expensive or inaccessible due to its developmental or limited availability to the public. By leveraging cloud services, organizations can overcome these hurdles and benefit from scalable computing resources and readily available software tools for efficient and cost-effective data processing. Consequently, it is preferable to utilize cloud services because they provide all the advantages and facilities that we would obtain by investing a substantial amount of capital at a low cost and are remotely accessible from any device, including smartphones. Hence, cloud services emerge as the optimal choice when dealing with intensive computational tasks, especially considering our project's requirement for real-time processing. Given that our users cannot feasibly carry the extensive hardware required to access our project's services on the go, relying on cloud services becomes the sole practical option to ensure seamless and efficient delivery of our project's functionalities in real-time.

Figure 7 is a block diagram that provides a visual representation of the exhaustive system architecture, emphasizing the key modules that are integral to the system's functionality. Detailed explanations of each module's function within the system are provided in the sections that follow.

a. Smartphone Camera: This module consists of a high-resolution, compact digital camera or webcam designed to be attached to headgear or a helmet and worn on the user's head. It records the environment and transmits a live video transmission. A compact laptop, which functions as the primary processing device, is connected to the camera.

b. Manager Module: This module facilitates the transmission of the live video signal from the head-mounted camera to the cloud service by acting as an intermediary.
It establishes an internet connection and assures the transmission of the video signal for later processing and decision-making.

c. Cloud Computing: Using various Application Programming Interfaces (APIs), the transmitted video data is received and processed in the cloud. Utilizing the computing power and resources of the cloud service, these APIs sequentially or concurrently execute multiple processes. Among the notable APIs utilized by the system are:
   - APIs for Range Finding Algorithms: These APIs employ specific algorithms to estimate the distance between the camera and detected objects, providing precise numeric measurements of the object's proximity.
   - Crowd Detection Algorithm APIs: These APIs analyze the video feed in order to identify and classify populations. They provide valuable information regarding the density or intensity of the gathering.

d. This module comprises of conventional earphones or headphones connected via an audio port to the user's device. It is the output medium for alerts and notifications delivered to the user. The system provides audible prompts through the earpiece to guide the visually impaired user and alert them to potential obstacles or hazards.

e. IoT Circuit Board: This component plays a crucial role in regulating the alerts' functionality. It controls the intensity of the audible signals transmitted through the earpiece and offers on/off functionality, allowing the user to tailor the device to his or her preferences.

f. Collision Warning System: This module's primary purpose is to emanate audible signals, alerting those in close proximity to the user to avoid collisions. This feature increases safety by providing an early warning system for the visually impaired user as well as those in the surrounding area.

g. Algorithmic Components: The system consists of multiple algorithms for sophisticated processing and decision-making. These algorithms, as illustrated in Figures 8 and 9, consist of:
   - This algorithm analyzes the live video input, detecting and estimating the distance of objects from the camera's position. It provides real-time data regarding the location and proximity of objects in the user's environment.
   - Crowd Detection Algorithm: This algorithm identifies and analyzes masses or gatherings within the video feed. By evaluating population intensity or density, it facilitates the user's navigation in congested areas.
   - And algorithm that will converts text messages or alerts into voice messages, enabling seamless communication with visually impaired users. It improves the system's utility by audibly conveying crucial information.

The system seeks to provide visually impaired individuals with enhanced situational awareness, precise distance estimation, crowd information, and aural alerts by integrating these modules and algorithms. This comprehensive strategy promotes safer navigation and collision avoidance, empowering users to navigate their surroundings with confidence.

The system is built to identify items in situations where there are few things present, or perhaps just one object. It effectively completes object classification and offers thorough details on the detected objects, including the precise areas of the image that have been correctly classified and their matching class labels.Even in scenarios where there are few objects or only one object of interest, the system accurately identifies and distinguishes objects using cutting-edge computer vision techniques and deep learning algorithms.

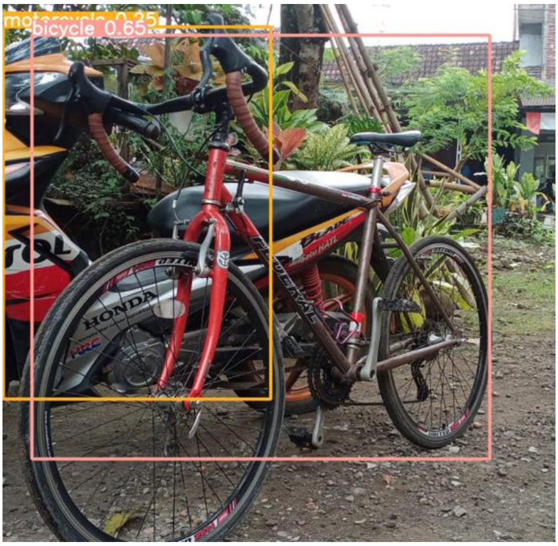

Figure 8. Sample Image: Detecting few objects

The system demonstrates its remarkable efficacy through its ability to accurately and precisely localize and segment objects within an image, harnessing the power of its advanced object identification capabilities. This profound functionality enables a comprehensive understanding of the spatial distribution and contextual relevance of the identified objects, contributing to a deeper level of insight.

Furthermore, going beyond the realm of mere object detection, the system offers a wealth of informative classification findings that significantly enhance its analytical capabilities. Each successfully identified object is accompanied by a specific zone of interest that precisely pinpoints its exact location within the image, further complemented by a meticulously assigned class label, providing invaluable details regarding the object's specific category or type. This meticulous analysis facilitated by the system encompasses not only the localized regions of the detected objects but also their corresponding class labels, serving as a rich source of information that goes a long way in aiding subsequent decision-making processes and downstream applications that heavily rely on accurate object classification and precise localization. Moreover, this comprehensive information serves to deepen our understanding of the overall composition and intricate layout of objects within the given image, opening doors to a more nuanced interpretation and interpretation of the visual content.

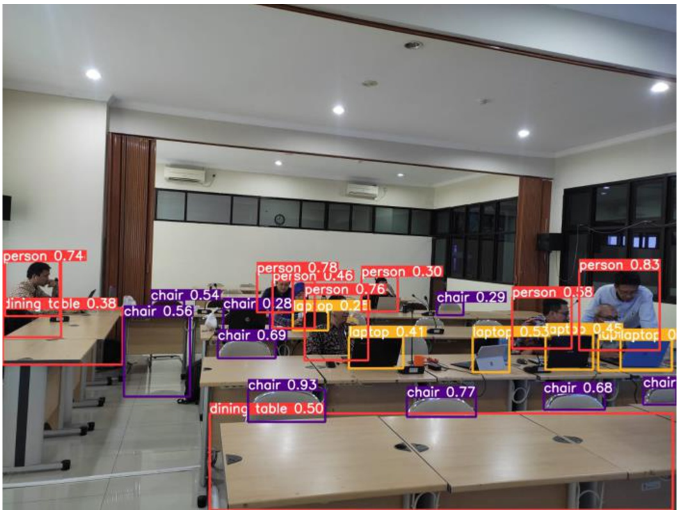

Figure 9. Sample Image: Detecting objects in crowded zone

4 Conclusion

In addition to aiding visually impaired individuals in averting collisions with objects, our proposed system includes sophisticated navigation and route guidance capabilities. Using artificial intelligence, cloud computing, mobile applications, and Internet of Things (IoT) technologies, our system provides visually impaired individuals with real-time navigation results during their commutes or leisurely excursions. The system identifies and analyses the encompassing environment, including objects, landmarks, and potential obstacles, by employing computer vision algorithms and object detection capabilities. The system calculates optimal navigation routes and provides auditory or haptic feedback to guide the user along the designated path by integrating GPS and map data. In addition, real-time environmental factors such as traffic conditions, pedestrian density, and road hazards are accounted for by our system. By perpetually analysing this information, the system dynamically adjusts the navigation instructions for visually impaired individuals to ensure safe and efficient navigation.

With our hands-free system, users only need to configure it once before departing their homes, eliminating the need for continuous manual input during their voyage. This seamless integration enables visually impaired individuals to navigate with ease, confidence, and independence, relieving them of the constant fear of collisions and enhancing their commute overall. We anticipate that as technology, particularly cloud computing, continues to advance, our proposed system will endure additional development and refinement. This ongoing evolution has the potential to facilitate the lives of the visually impaired by providing more sophisticated navigation features, increased precision, and seamless integration with other smart devices and platforms. Our ultimate goal is to enable visually impaired people to navigate the world with greater independence and accessibility.

References

[1] N. A. Giudice, *Navigating without vision: principles of blind spatial cognition*. Edward

Elgar Publishing, 2018.

[2] Y. Zhuang, J. Yang, Y. Li, L. Qi, and N. El-Sheimy, "Smartphone-based indoor localization with bluetooth low energy beacons," *Sensors (Switzerland)*, vol. 16, no. 5, pp. 1–20, 2016, doi: 10.3390/s16050596.

[3] M. Elgendy, C. Sik-Lanyi, and A. Kelemen, "Making shopping easy for people with visual impairment using mobile assistive technologies," *Appl. Sci.*, vol. 9, no. 6, 2019, doi: 10.3390/app9061061.

[4] A. Bhowmick and S. M. Hazarika, "An insight into assistive technology for the visually impaired and blind people: state-of-the-art and future trends," *J. Multimodal User Interfaces*, vol. 11, no. 2, pp. 149–172, 2017, doi: 10.1007/s12193-016-0235-6.

[5] E. Kostyra, S. Żakowska-Biemans, K. Śniegocka, and A. Piotrowska, "Food shopping, sensory determinants of food choice and meal preparation by visually impaired people. Obstacles and expectations in daily food experiences," *Appetite*, vol. 113, pp. 14–22, 2017, doi: 10.1016/j.appet.2017.02.008.

[6] R. Tapu, B. Mocanu, and T. Zaharia, "DEEP-SEE: Joint object detection, tracking and recognition with application to visually impaired navigational assistance," *Sensors (Switzerland)*, vol. 17, no. 11, 2017, doi: 10.3390/s17112473.

[7] R. Velázquez, E. Pissaloux, P. Rodrigo, M. Carrasco, N. I. Giannoccaro, and A. Lay-Ekuakille, "An outdoor navigation system for blind pedestrians using GPS and tactile-foot feedback," *Appl. Sci.*, vol. 8, no. 4, 2018, doi: 10.3390/app8040578.

[8] K. Manjari, M. Verma, and G. Singal, "A survey on Assistive Technology for visually impaired," *Internet of Things (Netherlands)*, vol. 11, 2020, doi: 10.1016/j.iot.2020.100188.

[9] R. Jafri, S. A. Ali, H. R. Arabnia, and S. Fatima, "Computer vision-based object recognition for the visually impaired in an indoors environment: a survey," *Vis. Comput.*, vol. 30, no. 11, pp. 1197–1222, 2014, doi: 10.1007/s00371-013-0886-1.

[10] S. A. S. Mohamed, M. H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila, "A Survey on Odometry for Autonomous Navigation Systems," *IEEE Access*, vol. 7, pp. 97466–97486, 2019, doi: 10.1109/ACCESS.2019.2929133.

[11] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognit.*, vol. 47, no. 6, pp. 2280–2292, 2014, doi: 10.1016/j.patcog.2014.01.005.

[12] E. Marchand, H. Uchiyama, and F. Spindler, "Pose Estimation for Augmented Reality: A Hands-On Survey," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 12, pp. 2633–2651, 2016, doi: 10.1109/TVCG.2015.2513408.

[13] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and R. Medina-Carnicer, "Generation of fiducial marker dictionaries using Mixed Integer Linear Programming," *Pattern Recognit.*, vol. 51, pp. 481–491, 2016, doi: 10.1016/j.patcog.2015.09.023.

[14] S. Al-Khalifa and M. Al-Razgan, "Ebsar: Indoor guidance for the visually impaired," *Comput. Electr. Eng.*, vol. 54, pp. 26–39, 2016, doi: 10.1016/j.compeleceng.2016.07.015.

[15] A. Morar *et al.*, "A comprehensive survey of indoor localization methods based on computer vision," *Sensors (Switzerland)*, vol. 20, no. 9, pp. 1–36, 2020, doi: 10.3390/s20092641.

[16] M. Elgendy, T. Guzsvinecz, and C. Sik-Lanyi, "Identification of markers in challenging conditions for people with visual impairment using convolutional neural network," *Appl. Sci.*, vol. 9, no. 23, 2019, doi: 10.3390/app9235110.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8691 LNCS, no. PART 3, pp. 346–

361, 2014, doi: 10.1007/978-3-319-10578-9_23.

[18] R. Girshick, "Fast R-CNN," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2015 Inter, pp. 1440–1448, 2015, doi: 10.1109/ICCV.2015.169.

[19] H. Rampersad, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *Total Perform. Scorec.*, pp. 159–183, 2020, doi: 10.4324/9780080519340-12.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[21] Wei Liu *et al.*, "SSD: Single Shot MultiBox Detector," *Eccv*, vol. 1, pp. 398–413, 2016, doi: 10.1007/978-3-319-46448-0.

[22] P. Soviany and R. T. Ionescu, "Optimizing the Trade-off between Single-Stage and Two-Stage Deep Object Detectors using Image Difficulty Prediction," *Proc. - 2018 20th Int. Symp. Symb. Numer. Algorithms Sci. Comput. SYNASC 2018*, pp. 209–214, 2018, doi: 10.1109/SYNASC.2018.00041.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

[24] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525, doi: 10.1109/CVPR.2017.690.

[25] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 6517–6525, 2017, doi: 10.1109/CVPR.2017.690.

[26] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018, [Online]. Available: http://arxiv.org/abs/1804.02767.

[27] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *ArXiv*, vol. abs/2004.1, 2020.

[28] S. Pang, Z. Yu, and M. A. Orgun, "A novel end-to-end classifier using domain transferred deep convolutional neural networks for biomedical images," *Comput. Methods Programs Biomed.*, vol. 140, pp. 283–293, 2017, doi: 10.1016/j.cmpb.2016.12.019.

[29] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, "A semi-supervised deep learning method based on stacked sparse auto-encoder for cancer prediction using RNA-seq data," *Comput. Methods Programs Biomed.*, vol. 166, pp. 99–105, 2018, doi: 10.1016/j.cmpb.2018.10.004.

[30] S. W. Yang and S. K. Lin, "Fall detection for multiple pedestrians using depth image processing technique," *Comput. Methods Programs Biomed.*, vol. 114, no. 2, pp. 172–182, 2014, doi: 10.1016/j.cmpb.2014.02.001.

[31] J. Tang, Q. Su, B. Su, S. Fong, W. Cao, and X. Gong, "Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition," *Comput. Methods Programs Biomed.*, vol. 197, p. 105622, 2020, doi: 10.1016/j.cmpb.2020.105622.

[32] C. González García, D. Meana-Llorián, B. C. Pelayo G-Bustelo, J. M. Cueva Lovelle, and N. Garcia-Fernandez, "Midgar: Detection of people through computer vision in the Internet of Things scenarios to improve the security in Smart Cities, Smart Towns, and Smart Homes," *Futur. Gener. Comput. Syst.*, vol. 76, no. October, pp. 301–313, 2017, doi: 10.1016/j.future.2016.12.033.

[33] B. Al-Madani, F. Orujov, R. Maskeliūnas, R. Damaševičius, and A. Venčkauskas, "Fuzzy logic type-2 based wireless indoor localization system for navigation of visually impaired people in buildings," *Sensors (Switzerland)*, vol. 19, no. 9, 2019, doi:

10.3390/s19092114.

[34] W. C. S. S. Simões, G. S. Machado, A. M. A. Sales, M. M. de Lucena, N. Jazdi, and V. F. de Lucena, “A review of technologies and techniques for indoor navigation systems for the visually impaired,” *Sensors (Switzerland)*, vol. 20, no. 14, pp. 1–35, 2020, doi: 10.3390/s20143935.

[35] S. T. Pundlik Matteo; Luo, Gang, “CVPR Workshops - Collision Detection for Visually Impaired from a Body-Mounted Camera,” *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, vol. NA, no. NA. pp. 41–47, 2013, doi: 10.1109/cvprw.2013.11.

[36] V.-N. N. Hoang Thanh-Huong; Le, Thi-Lan; Tran, Thanh-Hai; Vuong, Tan-Phu; Vuillerme, Nicolas, “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, 2016, doi: 10.1007/s40595-016-0075-z.

[37] K. Vetteth, P. Ganesh, and D. Srikar, “Collision avoidance device for visually impaired,” *Int. J. Sci. Technol. Res.*, vol. 2, no. 10, pp. 185–188, 2013.

[38] G. Lee and H. Kim, “A hybrid marker-based indoor positioning system for pedestrian tracking in subway stations,” *Appl. Sci.*, vol. 10, no. 21, pp. 1–20, 2020, doi: 10.3390/app10217421.

[39] Y. Li, S. Zhu, Y. Yu, and Z. Wang, “An improved graph-based visual localization system for indoor mobile robot using newly designed markers,” *Int. J. Adv. Robot. Syst.*, vol. 15, no. 2, pp. 1–15, 2018, doi: 10.1177/1729881418769191.

[40] Y. Bazi, H. Alhichri, N. Alajlan, and F. Melgani, “Scene description for visually impaired people with multi-label convolutional svm networks,” *Appl. Sci.*, vol. 9, no. 23, 2019, doi: 10.3390/app9235062.

[41] M. Elgendy, M. Herperger, T. Guzsvinecz, and C. S. Lanyi, “Indoor Navigation for People with Visual Impairment using Augmented Reality Markers,” *10th IEEE Int. Conf. Cogn. Infocommunications, CogInfoCom 2019 - Proc.*, pp. 425–430, 2019, doi: 10.1109/CogInfoCom47531.2019.9089960.

[42] L. López, G; Quesada, L; Guerrero, “Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces,” 2017.

[43] D. B. Johnson, “A Note on Dijkstra’s Shortest Path Algorithm,” *J. ACM*, vol. 20, no. 3, pp. 385–388, 1973, doi: 10.1145/321765.321768.

[44] A. de la V. Artificial, “ArUco: a minimal library for Augmented Reality applications based on OpenCV,” 2020. http://www.uco.es/investiga/grupos/%0Aava/node/26 (accessed Dec. 23, 2020).

[45] S. Kayukawa *et al.*, “BBEEP: A sonic collision avoidance system for blind travellers and nearby pedestrians,” *Conf. Hum. Factors Comput. Syst. - Proc.*, no. May, 2019, doi: 10.1145/3290605.3300282.

[46] A. Y. Rodríguez J. Javier; Alcantarilla, Pablo F.; Bergasa, Luis M.; Almazán, Javier; Cela, Andrés, “Assisting the Visually Impaired: Obstacle Detection and Warning System by Acoustic Feedback,” *Sensors (Basel).*, vol. 12, no. 12, pp. 17476–17496, 2012, doi: 10.3390/s121217476.

[47] S. K. Singh, S. Rathore, and J. H. Park, “Blockiotintelligence: A blockchain-enabled intelligent IoT architecture with artificial intelligence,” *Futur. Gener. Comput. Syst.*, 2020, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X19316474.

[48] K. K. K. Singh Vibha, “Smart Wireless Network Algorithm in the Era of Big Data,” in *Lecture Notes in Networks and Systems*, vol. NA, no. NA, 2021, pp. 1–8.

[49] F. S. . L. Bashiri Eric; Badger, Jonathan C.; D’Souza, Roshan M.; Yu, Zeyun; Peissig, Peggy L., “ISVC - Object Detection to Assist Visually Impaired People: A Deep Neural Network Adventure,” in *Advances in Visual Computing*, vol. 11241, Springer

International Publishing, 2018, pp. 500–510.

[50] X. Chen and A. L. Yuille, "A Time-Efficient Cascade for Real-Time Object Detection: With applications for the visually impaired," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, 2005, p. 28, doi: 10.1109/CVPR.2005.399.

[51] M. Afif, R. Ayachi, Y. Said, E. Pissaloux, and M. Atri, "An evaluation of retinanet on indoor object detection for blind and visually impaired persons assistance navigation," *Neural Process. Lett.*, 2020, doi: 10.1007/s11063-020-10197-9.

[52] M. Afif, R. Ayachi, E. Pissaloux, Y. Said, and M. Atri, "Indoor objects detection and recognition for an ICT mobility assistance of visually impaired people," *Multimed. Tools …*, 2020, doi: 10.1007/s11042-020-09662-3.

[53] T. Winlock, E. Christiansen, and S. Belongie, "Toward real-time grocery detection for the visually impaired," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, 2010, pp. 49–56, doi: 10.1109/CVPRW.2010.5543576.

[54] R. C. Y. Joshi Saumya; Dutta, Malay Kishore; Travieso-González, Carlos M., "Efficient Multi-Object Detection and Smart Navigation Using Artificial Intelligence for Visually Impaired People," *Entropy (Basel).*, vol. 22, no. 9, pp. 941-NA, 2020, doi: 10.3390/e22090941.

[55] U. S. Masud Tareq; Malaikah, Hunida M.; Islam, Fezan Ul; Abbas, Ghulam, "Smart Assistive System for Visually Impaired People Obstruction Avoidance Through Object Detection and Classification," *IEEE Access*, vol. 10, no. NA, pp. 13428–13441, 2022, doi: 10.1109/access.2022.3146320.

[56] Bogusław Cyganek, *Object Detection and Recognition in Digital Images*. John Wiley & Sons Ltd, 2013.

[57] A. Bhalla, S. Goutham, K. Prakash, and T. Sanjana, "VIEW: Optimization of Image Captioning and Facial Recognition on Embedded Systems to Aid the Visually Impaired," 2021, doi: 10.1109/C2I454156.2021.9689405.