International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Conference Proceedings: Volume: 05 ACME – 2026 DOI: 10.55041/ISJEM.ACME109
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

YOLO-Based Real-Time Object Detection with Audio Feedback for Visual Accessibility

D SIRISHA [1]
Assistant Professor,
Department of AI&DS
Annamacharya Institute of Technology and Sciences, Tirupati – 517520, A.P.
sirishaaids@gmail.com

K VIJAY KUMAR [2]
UG Scholar, Department of AI&DS
Annamacharya Institute of Technology and Sciences, Tirupati – 517520, A.P.
Kalavijaykumar2004@gmail.com

D VISHNU VARDHAN REDDY [3]
UG Scholar, Department of AI&DS
Annamacharya Institute of Technology and Sciences, Tirupati – 517520, A.P.
dvishnuvardhanreddy87@gmail.com

A SNEHA [4]
UG Scholar, Department of AI&DS
Annamacharya Institute of Technology and Sciences, Tirupati – 517520, A.P.
alisettysneha@gmail.com

K CHARANYA REDDY [5]
UG Scholar, Department of AI&DS
Annamacharya Institute of Technology and Sciences, Tirupati – 517520, A.P.
charanyareddy24@gmail.com

Abstract— Blind persons need some sort of help to feel secure while moving. Vision Assist is a smartphone app for visually impaired persons. It is a deep learning-based intelligent assistant for visually impaired persons. It makes the user more accessible by understanding the camera input and providing feedback. It is a smartphone app that is compatible with all smartphones.With the help of the app's real-time object recognition and the ability to set the app's settings according to the user's requirements, visually impaired persons can move independently and confidently in the environment. Users can get feedback about the environment using text-to-speech communication

Keywords— Vision Assist, Smartphone app, Visually Impaired, Deep Learning, Real-time Assistance, Accessibility, Object Recognition, Independence, Confidence.

I.INTRODUCTION

The disabled persons, especially the blind and the partially sighted, have to face a number of challenges in their day to day life. Even though the White cane and Guide dogs, though very useful, are to some extent limited in the sense that they cannot sense an object in the environment at a distance and/or provide contextual information. Therefore, the incorporation of such AI object detection techniques into the most frequently used devices such as mobile phones can be highly useful in addressing this challenge as they can provide in detail the information about the environment of the user. This paper aims at implementing an object detection application in real-time to assist the blind and visually impaired persons. The application, with the incorporation of deep learning techniques, can detect different objects within the range of the camera of the smartphone. The detected objects can then be described to the user through the use of text-to-speech programs in order to give the user an immediate insight into the environment. The main objectives of this study are:For the purpose of increasing the number of possibilities for the movement of the person with impaired vision, and for the purpose of ensuring that

the system used is rather convenient.The design of the application interface at the moment is based on the assumption of its compatibility with all possible types of non smart phones, beginning with classical ones with a minimum of possibilities up to low-end ones with the possibility of a limited download capability in order to ensure the maximum coverage of the potential number of users of the application. Besides, additional features include more custom options, which are mostly related to the Idea of how the given system might be developed in

order to provide different settings for the users.A. Novelty and contributions of the Vision Assist Deep Learning: Vision Assist makes use of the latest algorithms in object detection, which can accurately identify objects in real-time.Immediate Feedback: The application gives timely feedback to the users through text-to-speech, which allows them to understand their surroundings.Customization Options: The application allows users to customize settings such as voice speed and recognition, which gives them a personalized experience based on their unique needs.Wide Accessibility: The application can run on all smartphones, which allows visually impaired persons to access the application with ease.

II. LITERATURE SURVEY

"Imagine living in a world without sight. Daily activities have become daunting tasks. For this, various innovative assistive technologies have been introduced to provide visual aid and make life easier for people with impaired vision. These technologies try to fill the gap by communicating the hurdles and dangers, enabling people to move confidently through their surroundings.This literature review explores the triumphs and drawbacks of the technologies already available for assisting blind and visually impaired people. By studying these technologies, we can learn more about how to make them better and more efficient for a more inclusive and accessible society."The proposed method, 'Vision Assist,' has tremendous potential to improve the quality of life of people with visual impairment. Real-time object identification of the surroundings can provide immense support to access the surroundings. This paper covers the existing methodologies of object detection systems from a methodology, effectiveness, and challenges point of view, which are specific to blind people.

1. Traditional Approaches

- Ultrasonic and Infrared Sensors

The early assistive devices for the blind used ultrasonic and infrared sensors for the identification of hurdle sin the environment of the blind person. These devices,similar to the Ultra cane, also use the principle of echolocation for operation on the basis of feedback using vibrations. Even though these devices came into contact with the neighboring hurdles easily, there are some disadvantages of these devices such as range problems, identification of certain objects, providing minute details of the environment, etc., and hence did not prove to be very successful.

2. Computer Vision-Based Approaches

- Image Processing Techniques

Although the computer vision approach is still evolving,object detection has been improved using various imageprocessing techniques. Different techniques for thedetection of objects from the images are used,including the use of edge detection, contour detection,and Optical Character Recognition techniques. Eachtechnique has a high computational cost andperforms poorly in the case of real-time processing and varying illumination conditions.

- Deep Learning Models

The use of deep learning techniques has improved object detection by a great margin, both in terms of accuracy and speed of the object detection process.Conventional Neural Networks and Region-based CNN are the most dominant techniques used for the object detection approach. The ability of these techniques to perform the task of object detection in real-time makes them the best suitable techniques for the development of assistive devices for the blind.

3. Assistive Technologies and Applications

There are various smartphone applications that are developed in this context so far that can help the blind detect objects using obstacle detection techniques from the camera of the smartphone itself. Some of these applications include "Seeing AI"developed by Microsoft that can use the camera of the smartphone for object detection, reading text, and identifying faces using deep learning technologies for accurate results along with speed as well.

- Wearable Devices

Wearable devices that are equipped with camera technology can also be used for object detection without the use of hands. Some of these devices include Or Cam My Eye and Aira that use audio feedback for the blind person about the environment using the camera technology integrated into these devices. These devices also make use of computer vision technology for object detection.

- Smart Glass

Envision Glasses are smart glasses that make use of augmented reality technology for object detection functions. This allows the blind person an immersive experience wherein the person can recognize objects, read text, navigate, etc.

The technology makes use of your smartphone along with cloud services for an efficient solution.

A. Existing System

There are various existing systems that are similar to the goals and objectives of the "Vision Assist" project, a few of these are mentioned
below:
1) White Cane: The white cane is an important device used by people with visual impairments to move around. The user can feel objects in front of them. However, it has limitations in recognizing objects at a distance or above ground level.

2) Guide Dogs: Guide dogs are trained to assist people with visual impairments. These dogs are effective in guiding people with visual impairments away from obstacles. However, only a limited number of people with visual impairments are able to afford them.

3) Electronic Travel Aids (ETAs): These are devices used by people with visual impairments. These devices include ultrasonic canes and belts. These devices are effective in guiding people with visual impairments. However, they are limited to recognizing objects only.

4) Smartphone Apps: These are applications used by people with visual impairments. These applications are installed on smartphones. The smartphone has a camera that can recognize objects. These applications are effective in guiding people with visual impairments. However, they are limited to recognizing objects only.
B. Object Detection and Tracking algorithms
1. YOLO: You Only Look Once, or YOLO, is a real-time object detection algorithm that ranks among the top today. It's an amalgamation of speed and accuracy. Therefore, it's one of the most commonly used algorithms in computer vision. The YOLO architecture is one of the most prominent single-stage object detection networks. It makes use of a convolutional neural network to directly process the entire image at once and predict bounding boxes and probabilities of each class for the detected objects. The image is divided into an S x S grid. Each grid cell processes a fixed number of bounding boxes along with their probabilities and the probabilities of the class of each object. The CNN of the YOLO algorithm consists of several convolutional layers. The output of these convolutional layers is then fed into several fully connected layers. The bounding box parameters and probabilities of each class are then directly computed from the learned features of the image. The architecture has evolved from YOLO1 to YOLO8.

Fig. 1. YOLO Timeline in years from 2015 to 2021

2. Coco.Name:
Name file is an essential component used in conjunction with YOLOv3 and other object detection models to translate the numerical predictions of the model into human-readable labels. This file contains the names of the 80 object classes that the YOLOv3 model can detect from the COCO data set.

elephant	skis	wine glass	broccoli	dining table
bear	snowboard	cup	carrot	toilet
zebra	sports ball	fork	hot dog	tv
giraffe	kite	knife	pizza	laptop
backpack	baseball bat	spoon	donot	mouse
umbrella	baseball glove	bowl	cake	remote
handbag	skateboard	banana	chair	keyboard
tie	surfboard	apple	couch	cell phone
suit case	tennis racket	sandwich	plotted plant	microwave
frisbee	bottle	orange	bed	oven

Fig. 2. Items present in the coco names file.

3. Yolov3.weights: These are the hefty pre-trained parameters for YOLOv3, which helps the model detect objects in images or videos. They are downloaded separately due to their hefty size, but they're important for the model to work correctly.

Fig. 3. Yolov3.weights working

4. YOLOv8.tflite: YOLOv8.tflite is the TensorFlow Lite variant of the YOLOv8 object detection model. The conversion aims to deploy the YOLOv8 model on mobile and embedded platforms. The goal is to optimize the YOLOv8 model for object detection on mobile and embedded platforms.

5. CNN: Convolutional neural networks are effective for image-based tasks such as object detection. Therefore, CNNs have been integrated into Vision Assist. CNNs are effective for real-time object detection and recognition. CNNs learn spatial hierarchies of features from the images. YOLO (You Only Look Once) is a CNN-based object detection algorithm that is effective for real-time object detection and recognition. YOLO is effective for providing users with immediate insight into their environment. This is particularly important for assisting visually impaired users.

The following are the sources for the datasets and weights:

1. YOLO: The original YOLO and its updated versions are available on the official GitHub repository of YOLO.

2. coco.names: This file can usually be found within the YOLO repository mentioned above, where the implementation of the YOLO object detection algorithm is done, or within the documentation of the COCO dataset.

3. YOLOv3.weights: The pre-trained weights of the YOLOv3 model are available for download on the official GitHub repository of YOLO.

4. YOLOv8.tflite: The YOLOv8 and its TensorFlow Lite version are available on the GitHub repository of Ultralytics.

5. CNN (Convolutional Neural Network): The training datasets used for CNN models are quite varied. However, some of the most commonly used ones are ImageNet, CIFAR-10, and COCO dataset.

III. METHODOLOGY

1. Imports and setup:
Importing the necessary libraries: CV2 for real-time computer vision, NumPy for numerical computations, and pyttsx3 for text-to-speech conversion.

- CV2 is used for real-time vision.
- NumPy is used for all the heavy numerical computations.
- pyttsx3 is used for text-to-speech.

2. Loading the YOLO model and its classes (net and classes):

- The net object is used to load the weights and configuration of the YOLO model.
- The classes are loaded from the coco.names file containing the object classes.

3. Camera Setup and Initialization:

- Cap: Start capturing images using the camera.
- detected objects: A list to store information about detected objects.
- engine: Initialize text-to-speech engine.
- focal length, known_object_height: Parameters used to calculate distance.

4. Distance Calculation Function:

- Calculate_distance: A function to determine the distance of an object from the camera using its pixel height.
- distance = (Known_object_height * focal_Length) / object_height_pixels

5. Frame Smoothing Function:

- Smooth_frame: A function to smooth the frames to produce smooth video output.

6. Main Processing Loop:

- Previous_frame: Store previous frames to smooth them.
- While True: The main loop to run through each frame.
- ret, Frame: Read the previous frame from the camera.
- Frame Smoothing: Use smooth_frame to smooth frames.
- blob: Preprocess frames using YOLO.
- -net.set_input: Use YOLO to process frames.

7. Object Detection and Distance Calculation:

- We maintain a list for boxes, confidences, and class_ids to store the detected boxes, their respective confidences, and class_ids.
- Output processing: We process the items yielded by YOLO one by one.
- Object detection: We process the object detection results and filter out the boxes with confidences higher than 0.5.
- From the valid object detection results, we compute center_x, center_y, width, and height to obtain the bounding box.
- We append the bounding box coordinates to boxes.
- We append the class ID to the confidences list. (In the original, the description said the confidence list is used to store the class ID, but the corresponding line in the original code is written in such a way that it can be interpreted as storing the class ID along with the confidence.)

8. Non-Maximum Suppression (NMS):

- We use NMS to filter out the bounding boxes, leaving the important ones.

9. Drawing Bounding Boxes and Alerts:

- We use the boolean flag alert_triggered to decide whether to trigger an alert.
- We process the indices yielded by NMS.
- We extract the bounding box coordinates x, y, w, h.
- We add the detected object details to the list detected_objects.
- We compute the distance = Calculate_distance(h) to estimate the distance of the object.
- If the object is too close, i.e., the distance is less than 0.5, we display the alert message using Cv2.putText.
- We trigger the voice alert using the statement engine.say.
- object_info: prepare the summary of object details.
- cv2.rectangle: draw the bounding box around the detected item.
- cv2.putText: show the object information on the frame.
- if alert_triggered: place a red warning frame around the entire image.

10. Display the frame and handle exiting:

- cv2.imshow: show the current frame with detected objects and alerts.
- cv2.waitkey(1): wait for a key press for 1 millisecond.
- if key == 27: exit when the Esc key is pressed.
- ap.release: release the camera.
- cv2.destroyAllWindows: close all OpenCV windows.

11. Print detected objects:

- for obj in detected_objects: iterate over all detected objects.
- print: output the details of each detected object..
-

VII. CONCLUSION

This project is based on the development of a real-time object detection system that can aid visually impaired people. The project incorporates computer vision, deep learning, and software engineering with the focus on precise object detection, precise estimation of the distance of objects from the user, smooth processing of frames for better visual clarity, and timely alerts through visual and auditory notifications for the enhanced safety of the user.

In the future, the intention is to develop an end-to-end mobile application with various features.

Moving forward, we intend to develop a fully featured mobile app, tailored to meet the needs of the visually impaired population. This app will be designed to provide an amalgamation of various useful features in one single, seamless product. Some of the prominent features of the app include:

- Object Recognition: This feature will allow the visually impaired to recognize objects in real time.
- Currency Detection: This feature will help the visually impaired distinguish between different denominations of money.
- Scene Description: This feature will provide an in-depth description of the environment.
- Text-to-Speech: This feature will help the visually impaired read written or displayed text.
- Emergency Contact Alerts: This feature will send alerts to trusted contacts in emergency situations, providing enhanced safety.

Furthermore, the project also plans to provide the visually impaired with specialized hardware, which can provide real-time guidance, along with directions and alerts based on the environment around the visually impaired individual. This project, with its cutting-edge mobile app and specialized hardware, can greatly benefit the visually impaired population in the country.

REFERENCES

[1] EAI Endorsed Transactions on Internet of Things, vol. 10, 2023. DOI: 10.4108/eetiot.4541.

[2] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: challenges, architectural successors, datasets and applications," Multimed. Tools Appl., vol. 82, pp. 9243–9275, 2023. DOI: 10.1007/s11042-022-13644-y.

[3] M. Pulipalupula, S. Patlola, M. Nayaki, M. Yadlapati, J. Das, and B. Reddy, "Object Detection using You Only Look Once (YOLO) Algorithm in Convolution Neural Network (CNN)," in 2023 International Conference for Convergence in Technology (I2CT), 2023, pp. 1-4. DOI: 10.1109/I2CT57861.2023.10126213.

[4] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: challenges, architectural successors, datasets and applications," Multimed. Tools Appl., vol. 82, pp. 9243–9275, 2023. DOI: 10.1007/s11042-022-13644-y.

[5] K. Vaishnavi, G. Reddy, T. Reddy, N. Iyengar, and S. Subhani, "Real-time Object Detection Using Deep Learning," J. Adv. Math. Comput. Sci., vol. 38, pp. 24-32, 2023. DOI: 10.9734/jamcs/2023/v38i81787.

[6] D. Li, "Research advances in object detection based on deep learning," Appl. Comput. Eng., vol. 5, pp. 603-608, 2023. DOI: 10.54254/2755-2721/5/20230656.

[7] P. Druzhkov and V. Kustikova, "A survey of deep learning methods and software tools for image classification and object detection," Pattern Recognition and Image Analysis, vol. 26, no. 1, pp. 9–15, 2016.

[8] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. Mathieu, and Y. LeCun, "Learning convolutional feature hierarchies for visual recognition," in Advances in Neural Information Processing Systems (NIPS), 2010.

[9] C. Szegedy, A. Toshev, and D. Erhan, “Deep neural networks for object detection,” in NIPS, 2013.

[10] P. Dollar, R. Appel, S. Belongie, and P. Perona, “Fast feature pyramids for object detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 8, pp. 1532–1545, 2014