World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

Check for updates

Empowering the visually impaired: A YOLO-v5 CNN-based android app for currency recognition and object detection

Harshit Kumar [1], Anna Paul [1], Anjana V P [1], Anandakrishnan S Babu [1] and Gopakumar V T [2, *]

[1] *Adi Shankara Institute of Engineering and Technology, APJ Abdul Kalam Kerala Technological University, Kaladi, Kerala, India, 683574.*
[2] *Mar Baselios Chritian College of Engineering & Technology, APJ Abdul Kalam Kerala Technological University, Kuttikkanam, Peermade, Idukki, Kerala, India 68553.*

World Journal of Advanced Research and Reviews, 2023, 19(02), 1459–1466

Publication history: Received on 01 July 2023; revised on 17 August 2023; accepted on 19 August 2023

Article DOI: https://doi.org/10.30574/wjarr.2023.19.2.1625

Abstract

This paper presents an Android mobile Application (app) that assists people with visual impairments in currency recognition and general object detection. Our project's goal is to assist visually impaired individuals by offering them a practical alternative with a cost-effective solution. The YOLO-v5 CNN model-based currency note detection and object recognition app is proposed for this purpose, and we find it is fast and accurate. The app can identify rupee notes with denominations of 10 new, 10 old, 20 new, 20 old, 50 new, 50 old, 100 new, 100 old, 200, 500, 2000, and general objects such as mobile phones, laptops, chair, water bottle, vehicles, persons, book, door, watch etc.

Keywords: Currency recognition; General object detection; YOLO-v5; CNN component; Android studio; GitHub; Google Colab.

1. Introduction

Human vision is crucial for understanding the environment, yet visually impaired individuals lack access to this information without external help. They often rely on external knowledge, experience, and technology to navigate unfamiliar places for independent mobility. Researchers aim to assist visually impaired individuals in comprehending their surroundings and exploring autonomously. A system is being considered to enhance navigation for the visually impaired in both indoor and outdoor environments. As blindness rates increase globally, various assistive technologies are being developed to help visually impaired people live normal lives. India has a significant share of the world's blind population, with around 8 million blind individuals out of 39 million people. However, only a small percentage, 5%, have access to assistive technology [1]. In today's fast-paced world, environmental obstacles pose challenges, particularly for those with vision impairment who often rely on external assistance or technology for decision-making.

2. Related work

Rohit Agarwal et al. [2], D. Munteanu et al. [3], Yeong-Hwa Chang et al. [4], a Danish mobile software called Be My Eyes [5] the OrCam Read [6] were developed different types of assistive devices. But unfortunately, these assisting devices have several drawbacks, including limited availability, low weight capacity, short-distance detection, inaccurate results (leading to high false positives), high costs (OrCAM costs around Rs3 lakh), dependence on high-bandwidth AI systems requiring a personal assistant, and absence of devices in the Indian market. To address these issues, the proposal suggests creating a cost-effective Android app. This app, which only requires an Android phone and internet connection, aims to provide a solution that is widely accessible in India. By utilizing open-source software, the app can be provided

* Corresponding author: Gopakumar V T

Copyright © 2023 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution Liscense 4.0.

to users for free. Its user-friendly design eliminates the need for a dedicated helper, offering a straightforward solution for users' needs.

3. A brief overview of the proposed solution

As shown in figure, the project addresses the lack of accessible and affordable solutions for assisting visually impaired individuals. It introduces an Android app that utilizes voice inputs and outputs to aid users. The app can identify Indian currency notes and general objects, with users issuing voice commands for different functions. Images taken by the phone's camera are processed by an API server, and the app converts the results into spoken output. This solution targets Indian users and aims for widespread availability. The app's process involves capturing images, sending them to the server for analysis, and converting the results into voice output for the user.

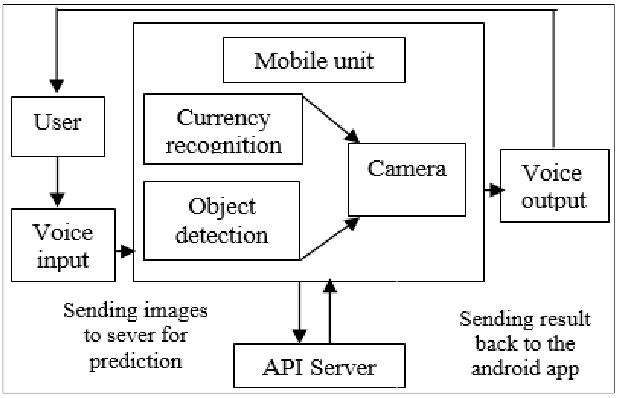

Figure 1 Proposed model

3.1. Framework Created For The Product

A framework has been developed as show in figure 2, that not only facilitates project construction but also enhances the performance of machine learning (ML) models over time. This framework enables automatic dataset updates and model retraining for improved accuracy. The framework involves an Android app uploading images to a server via a POST request, where an ML model processes the image and sends back predictions. Server-stored results, including bounding boxes and labels, contribute to creating a structured dataset. This combined dataset is then used to train new ML models, replacing the old ones and leading to automatic accuracy improvement. This automated process reduces the need for extensive human intervention. Moreover, the framework aids in transforming unstructured datasets to structured ones by labeling a small portion, and it allows easy image capture and annotation via mobile phones using APIs.

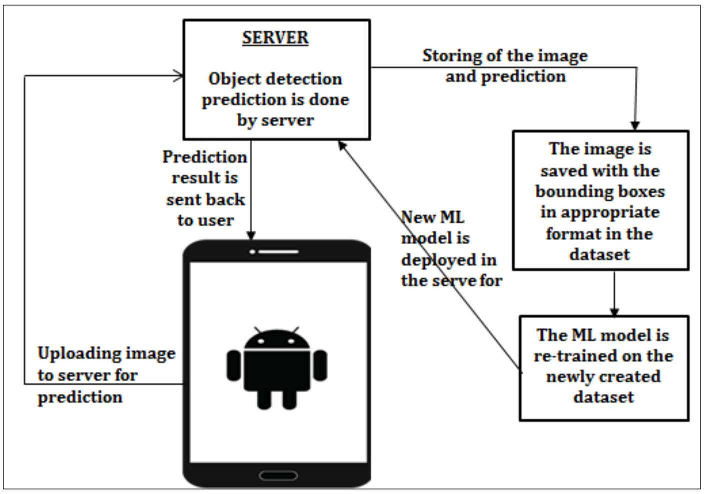

Figure 2 Framework

It can also be used for audio, text as well as video ML models also. The framework can also be easily deployed on a local machine to avoid cloud server costs and securely store any important dataset.

3.2. Block Diagram

The figure 3 shows the block diagram of our proposed system. The Indian currency dataset images were captured manually using a phone's camera, then organized into classes, labeled, and annotated using tools like LabelImg and a custom augmentation tool. The general object dataset was assembled similarly, with various images downloaded from diverse sources. After collecting images of items like laptops, people, vehicles, etc., a YOLOv5 machine learning model was trained on both datasets [7]. These models are hosted on a server, with API requests connecting them to a mobile app. Voice input selects between currency and general object detection, converted to text via Speech-to-Text. The app determines the chosen module based on the text and proceeds accordingly.

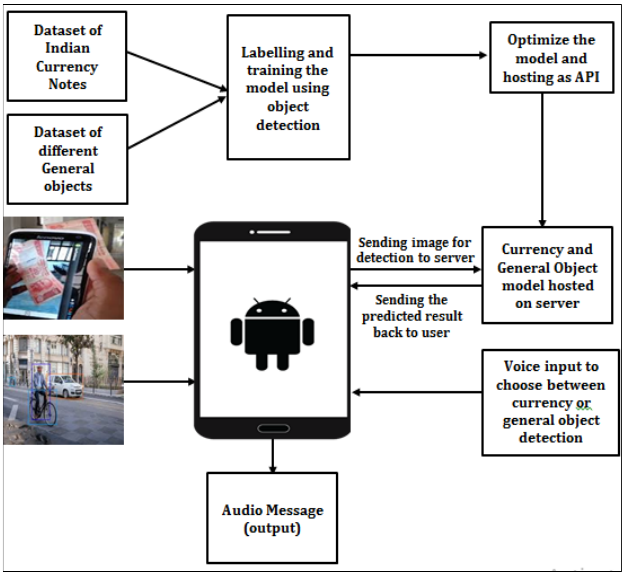

Figure 3 Block diagram

The process involves capturing an image with the camera, sending it to the server through a POST request for detection, and receiving results. The outcome is transformed into text, converted into speech via Text-to-Speech, and presented as audio output to the user. The project utilized several open-source tools, including Android Studio, Git/Github, Google Colab, YOLOv5, LabelImg, FastAPI, Teachable Machine, and Visual Studio Code.

4. Technical details

In this project, we have used datasets to train the model. Following is the information regarding the dataset, ML model, training information, and Android app information.

4.1. Dataset

Table 1 The dataset specification

Number of images of each currency note in the whole dataset	**~200 images per label (200 images of each note)**
Source of dataset	Manually taken
Notes Taken	Rs 10 new, 10 old, 20 new, 20 old, 50 new, 50 old, 100 new, 100 old, 200, 500, and 2000
Labelling format	YOLO format
Labelling tool used	LabelImg

Table 1 shows the information about the dataset. The Indian currency dataset images were captured manually, each label comprising approximately 200 images, all manually labeled and annotated using the LabelImg tool. For developing the Android app, a combination of Java and Kotlin was used in the Android Studio IDE. Java was chosen due to its historical prominence in Android development, while Kotlin's ease of use and availability of new tutorials made it an attractive option. Both languages can coexist within Android Studio without compatibility issues.

4.2. Machine Learning Model

Table 2 The Machine Learning model specification and training parameters

Methodology	**Transfer Learning Supervised Learning**
Machine learning Algorithm	YOLOv5
Model Size	83.4MB
Input Image Format	JPEG
Image Parameters	
Epochs	50
Batch Size	16
Image Size	640x640

Table-2 shows the information about the machine learning model and the training parameters. The methodology used is transfer learning and supervised learning. The algorithm used YOLOv5 which was released in June 2020. The model size of the trained model is 83.4 MB. For training, the model epoch is chosen as 50 and the batch size is 16. The input image size for the model is 640x640.

5. Results and discussion

After completing the project, we have received many results. These results include the model training output, APIs created for Indian currency detection and general object detection and the final Android app created. During the course of the project, we have also created a few tools to work on the dataset and speed up the process. Following are the results which we obtained.

5.1. Output from the Yolov5 ML Model

We trained the model on the YOLOv5 algorithm in Google Colab. We trained for 50 epochs. The batch size was 50 and the input image size was 640x640. The training of the model took around 50 mins and an accuracy of 95% was obtained. Fig 4 shows the output obtained after running the trained model on a test dataset. The app can detect rupees 10 new, 10 old, 20 new, 20 old, 50 new, 50 old, 100 new, 100 old, 200, 500, and 2000 currency notes.

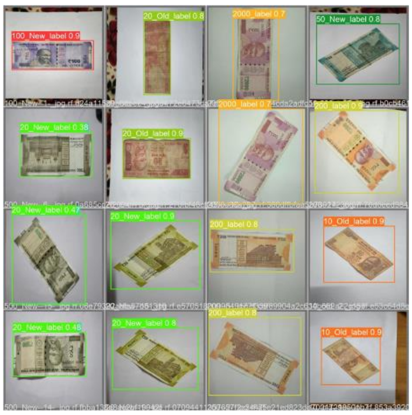

Figure 4 Test dataset output on YOLOv5 Model

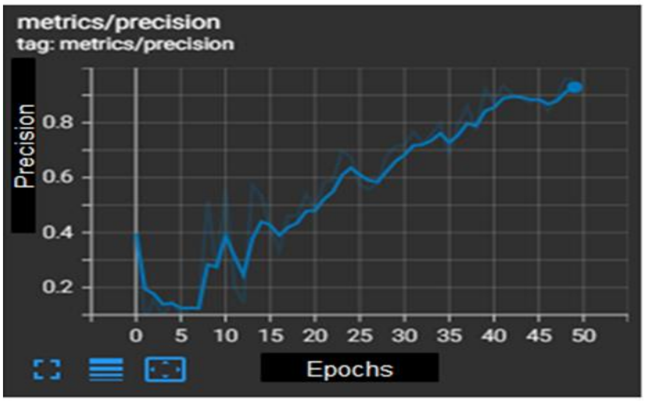

Figure 5 YOLOv5 precision graph

Figure 5 shows the precision of the model during the training, the x-axis is the epochs and the y-axis is the precision.During the start, the precision drops heavily from 0 to 5 epochs. This is because the model first gets high confidence about the data at the first epoch but as slowly the training continues the precision drops significantly. But after the fifth epoch, as the training continues the precision increases and by the end, precision reaches 95%. Figure 6 shows the mean average precision with threshold 0.5 (mAP 0.5). This metric defines the ability of the model to predict the bounding boxes correctly. The 0.5 is the threshold i.e., 50% confidence. mAP 0.5 means the ability of the model to produce the bounding boxes with the confidence of 50% or higher. These bounding boxes predicted are compared against the ground truth box which is the bounding boxes mentioned in the dataset. The figure 7 shows the confusion matrix of the model. A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It basically tells the false positive and false negative for each label.

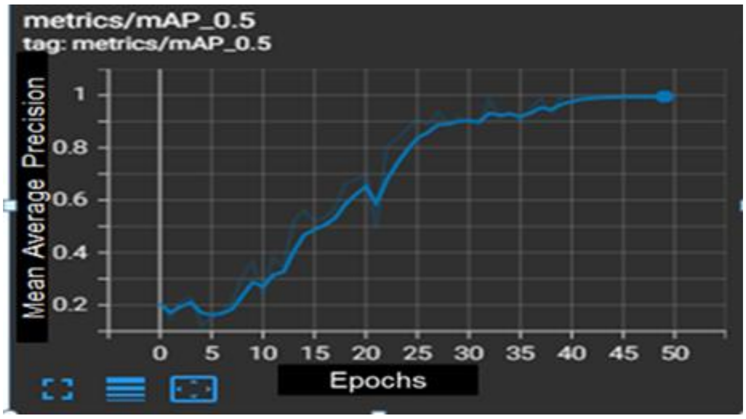

Figure 6 YOLO-v5 mAP 0.5

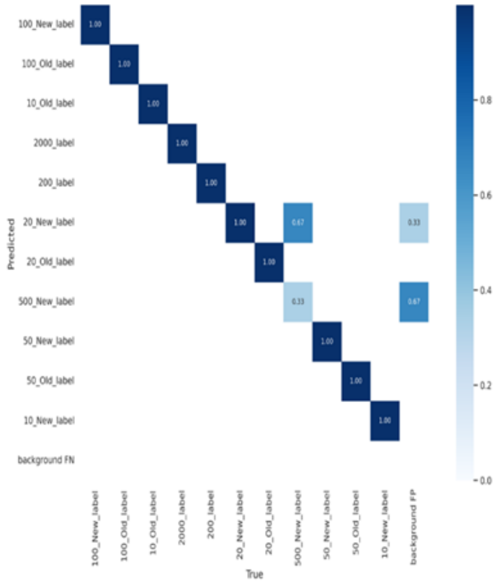

Figure 7 Confusion matrix

The x-axis is the True value fed to the model and the y-axis is the predicted value. In this confusion matrix it shows when 100 new labels are fed to the model, it predicted all the values correctly.

5.2. API Currency Output

The output of our Currency detection from the Fast API can be seen in figure 9 and it was showing an accuracy of 74%. The left image shows the API interface and the API output (Response body) sent back to the user. And the right image shows the model output for the same. The box defines the location of the object and the label and confidence score are also mentioned in the box.

Figure 8 Fast API Currency output

Figure 9 Fast API object detection display

5.3. API Object Detection Output

Fig. 9 displays the results of our Fast API object detection. The API output (Response body) that is returned to the user is shown in the left image beside the API interface. The model output for the same is also shown in the right image. The three water bottles in the image which is accurately identified by the model. The model also predicted the boundary boxes for each bottle and the confidence level.

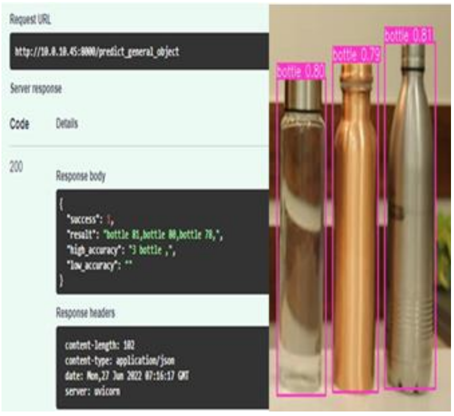

Figure 10 Fast API general object output

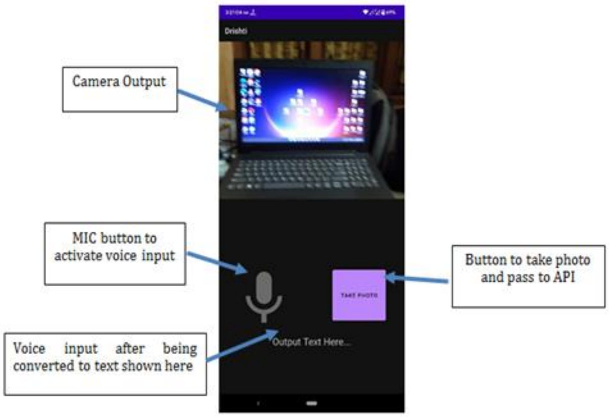

Figure 11 App interface

5.4. APP Interface

The UI of the Android app is displayed in figure 11. The camera output takes over half of the screen. By pressing the MIC, speech is captured and then converted into text utilizing speech-to-text functions. The text box displays the function's output. By pressing the take photo button, a picture is taken and sent right away to the server. General object detection is the pre-selected API call for the Take photo button.

5.4.1. Tools Created for this project

For the project, we built some tools which we used to work on the dataset. These tools are automation scripts that we used to complete many human-intensive tasks. The different tools created are Rename tool, Augmentation tool [8], dataset summary tool, image format conversion tool and auto cropping tool as shown in figure 12,13 14,15 and 16 respectively.

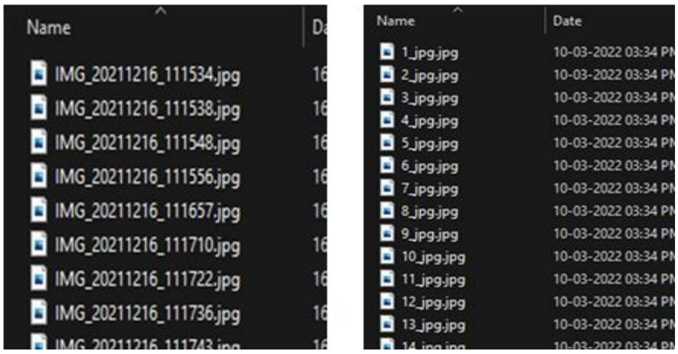

Figure 12 Rename tool and results

Figure 13 Original image (for augmentation)

Figure 14 Augmented image

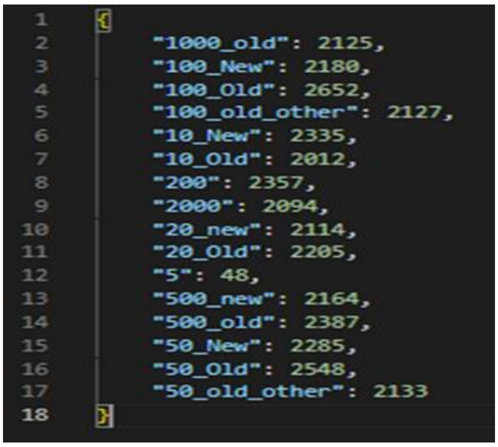

Figure 15 Dataset summary tool

Figure 16 Auto cropping Tool

6. Conclusion

Here an android application designed to assist visually impaired individuals. the app utilizes voice commands for operation and provides voice output to help users identify various objects in their surroundings, including indian currency notes. the app incorporates a general object detector and an indian currency detector, utilizing specialized datasets for accurate detection. the models are connected to the app via an api for prediction. the app also aims to add features like battery alerts, obstacle detection, ocr for multilingual text reading, and improved voice recognition. the goal is to enhance usability and functionality for visually impaired users.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to disclosed.

References

[1] Syed Tehzeeb Alam, et al., “Smart Assistive Device for Visually Impaired People”, International Journal Of Engineering Research & Technology (IJERT) Volume 04, Issue 03 (March 2015), http://dx.doi.org/10.17577/IJERTV4IS030921

[2] R. Agarwal et al., "Low cost ultrasonic smart glasses for blind," 2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2017, pp. 210-213, doi: 10.1109/IEMCON.2017.8117194

[3] D. Munteanu and R. Ionel, "Voice-controlled smart assistive device for visually impaired individuals," 2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC), 2016, pp. 186-190, doi: 10.1109/ISETC.2016.7781087I.

[4] Y. -H. Chang, N. Sahoo and H. -W. Lin, "An intelligent walking stick for the visually challenged people," 2018 IEEE International Conference on Applied System Invention (ICASI), 2018, pp. 113-116, doi: 10.1109/ICASI.2018.8394480.

[5] https://www.neilsquire.ca/eyes-app-review/

[6] https://www.orcam.com/en/myeye2/

[7] A.Kahlfaoui, A. Badri and I. E. Mourabit, "Comparative study of YOLOv3 and YOLOv5's performances for real-time person detection," 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2022, pp. 1-5, doi: 10.1109/IRASET52964.2022.9737924.

[8] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," 2018 International Interdisciplinary PhD Workshop (IIPhDW), 2018, pp. 117-122, doi: 10.1109/IIPHDW.2018.8388338.