International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 04 | Apr 2022 www.irjet.net p-ISSN: 2395-0072

YOLOv5 BASED WEB APPLICATION FOR INDIAN CURRENCY NOTE DETECTION

Sree Ashrit Dande[1], Gowtham Reddy Uppunuri[2], Dr. Ajay Singh Raghuvanshi[3]

[1,2]UG Student, National Institute of Technology, NIT Raipur, Chhattisgarh, India
[3] Assistant Professor, Dept. of Electronics and Telecommunication Engineering, National Institute of Technology, NIT Raipur, Chhattisgarh, India

---***---

Abstract - *Indian currency notes, issued by Indian government, are involved in monetary transactions across India. The paper aims to provide an efficient method for detection of Indian currency notes, which is an application of object detection using deep learning. The detection of currency notes aids visually impaired people who can't recognise the monetary value in their possession. Hence, an effective model is designed in which currency notes are detected and human speech of detected currency note is provided. The notes are first trained using YOLO-v5 model and then, the results are validated against validation data. The evaluation metrics are observed and model losses are minimized. Then, a new test data is considered as inference for recognition of classes. The web app is designed using YOLOv5, Flask and web technologies. The web app detected Indian currency notes with bounding box probability greater than 0.80 for all the classes. The labels of each note are detected with bounding box probability greater than 0.90 and labels are spoken out clearly using this designed model.*

Key Words*:* **Indian Currency notes, YOLO-v5 Algorithm, Flask, JavaScript, Evaluation Metrics, Model losses**

1. INTRODUCTION

Object Detection is technique used in detection of real-time objects, in images and videos using either machine learning or deep learning and OpenCV. Object detection is applied in surveillance, image retrieval, security, Advanced Driver Assistance Systems (ADAS). Object detection includes image classification and object localization [1].

Based on WHO statistics, at least 2.2 billion people have a near or distance vision impairment. The leading causes for visual impairment include uncorrected refractive errors, cataract, diabetic retinopathy, glaucoma and age-related vision impairment [2]

In 2017, National Center for Biotechnology Information (NCBI) reported that 253 million people globally were visually impaired among which 217 million had visual impairment and remaining were identified as blind [3]. Visually impaired lack the ability to interpret bank-notes is very limited and usually depend on others. The aim is to overcome this difficulty and to provide easier currency detection for visually impaired. The currency notes are detected based on feature extraction, considering geometric size and characteristic texture [4].

In India, Indian Currency Recognition System currently uses CNN and OpenCV based recognition system to detect monetary notes [5]. This model has lower detection accuracy than the proposed system which uses YOLOv5 for object detection.

The proposed system involves YOLOv5 algorithm to detect test images. The model is trained using transfer learning and is later inferenced against test data.

A web-based application is designed using YOLOv5 and Flask. This app provides labels on the screen as well as audio file of detected currency note, when a currency note image is uploaded. The audio file can be listened to using play button provided in the web app. This will not only aid visually impaired, but also, provide easier way to recognize Indian currency notes to tourists and foreigners visiting India.

The challenge is to design a web application based on YOLOv5 model to detect Indian currency notes accurately and provide optimized and efficient results, both on laptop and mobile.

The rest of the paper is organized as follows. First, review of related work is briefly discussed in Section II, followed by methodology in Section III. The experimental results are presented, followed by conclusion and future scope are presented in Section 1V.

2. LITERATURE REVIEW

Shenmei Luo and Wenbin Zheng (2021) proposed a table detection method for academic paper based on YOLOv5 (You Only Look Once v5) where detection accuracy and speed of proposed approach are observed to be superior compared to other competitive approaches [6].

Sagar et. al (2021) designed a system for detection of Indian currencies that aids visually impaired people to recognize and read out the possible Indian paper currencies with 79.83% accuracy [4]. The proposed model is based on YOLOv5, where it compares the efficiency of YOLOv5 with CNN algorithm for currency detection.

Sai Nadh et. al (2020) has developed a module for blind people and design mobile application for visually impaired Indian citizens to recognize currency value when held near android phone to predict the currency value so that currency can be recognized correctly [7].

Shreya et. al (2021) designed a mobile application for Visually Impaired Persons – Blind Assist [8] based on web technologies to scan currency note using device's camera and get its denomination using a voice assistant.

Raghad Raied Mahmood et. al proposed real-time Iraqi currency note detection system that aids visually impaired people using YOLO-v3 and gTTS to obtain audio output. custom Iraqi banknote dataset for detection and recognition of banknotes. The Iraqi notes were detected in a short time with high mAP of 97.405% [9].

Mansi Mahendru and Sanjay Kumar Dubey (2021) proposed system for visually impaired to detect multiple objects and prompt voice to alert person stating near and farther objects around them using gTTS and TensorFlow [10]. A comparison between YOLO and YOLO_v3 for object detection was tested under same criteria based on measures like accuracy and model performance.

3. METHODOLOGY

The experimental procedure is divided into – YOLOv5 Model, Evaluation Parameters and Web app Design.

3.1 YOLO-v5 Model

The model is selected from ultralytics repository, which is the official YOLOv5 repository on GitHub. The repo is cloned along with installation of dependencies. Later, the configuration (yaml) files are customized by providing the class names from ₹10 to ₹2000. The YOLOv5 model chosen comprises of 283 convolutional layers.

Dataset and Experimental Environment:

Currency notes used in the experiment are collected from sources like banks and shops. The dataset comprises of old notes of ₹10, ₹20 and new notes of ₹50, ₹100, ₹200, ₹500 and ₹2000 images (in .jpg format), in equal proportions. The images data comprise 200 images per class, which are captured on different backgrounds and different visual settings.

Dataset Augmentation: These 1400 images are then augmented using rotation, blur and shift transformation techniques to generate dataset with 5600 images.

Dataset Annotation: These 5600 images are then manually annotated using MakeSense annotator tool. The annotations are then exported in YOLO format and stored in labels directory

Split the entire dataset along with their labels in 70-30 split. The training dataset comprises 3920 images and their labels. Remaining 1680 images along with their annotations comprise the validation dataset.

Mount the dataset onto Google Drive. Connect to GPU runtime in Google Colaboratory.

Setting up the config files: The yaml files (custom_train and custom_data) that contain model parameters are modified accordingly for seven classes. The class names along with total classes are set accordingly. The paths for train and validation dataset are provided. These files must be present to train the dataset.

Clone the YOLOv5 repository in the IPYNB file and install required dependencies. The running environment for this model is - CUDA:0 (Tesla T4, 15110MiB) (2 CPUs, 12.7 GB RAM, 39.9/78.2 GB disk). Development packages include torch 1.10.0 and Python 3.8.

Train the model: The image size is set to 416. The model epochs are set to 30 and batch size is set to 32. Set the directory path for train dataset and configuration files. Then, run the python command to train the entire dataset using YOLOv5 model. The evaluation metrics are – Precision, Recall and mean Average Precision (mAP). The losses calculated for box, object and class for each epoch.

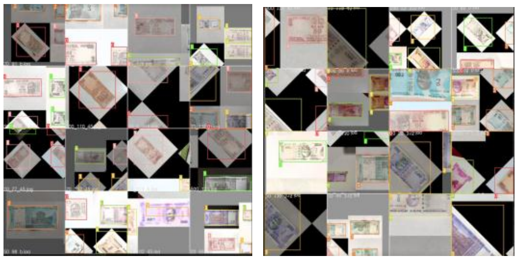

Fig -1: Model storing train data labels

Store the best model weights in .pt format.

Inference the model against test data: The model is inferenced using best.pt as model weight and setting the directory path of test data. The test data contains 100 images of all classes and video file containing currency note image frames.

3.2 Evaluation Parameters

The evaluation metrics used to analyse the results include – Precision, recall and mAP. Higher values of recall, precision and mAP (mean Average Precision) are preferred to detect the currency notes accurately. The other metrics include box loss, obj loss and cls loss.

In a confusion matrix with True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN), the evaluation metrics Precision, recall and F1- score can be calculated.

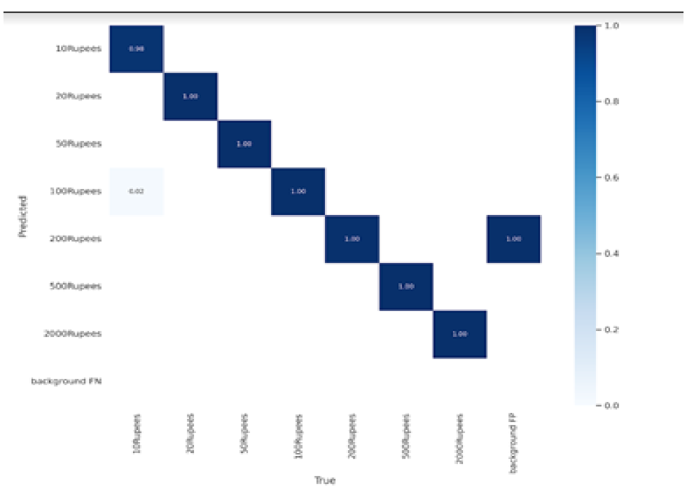

Fig. -2: Confusion Matrix of train data

Precision: Precision determines correctly identified positives among total true positives [11].

$$\text{Presicison} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

Recall: Recall determines correctly identified positives among total identified positives [11].

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

mAP (mean Average Precision): mAP represents mean of average accuracy in recognition of currency labels. mAP evaluates comprehensive detection accuracy [11].

mAP @0.5 indicates the precision -recall area that lies above the threshold 0.5.

Model Losses: There are three losses for every epoch, namely box loss, classification loss and object loss.

Box loss: It represents how well the algorithm locates object center and predict bounding boxes [12].

Objectness or obj Loss: The probability that object exists in proposed region of interest [12].

Classification Loss or cls Loss: Classification loss represents how well the algorithm predicts class of given object correctly [12].

The evaluation metrics along with model losses for each epoch are shown in Fig 3.

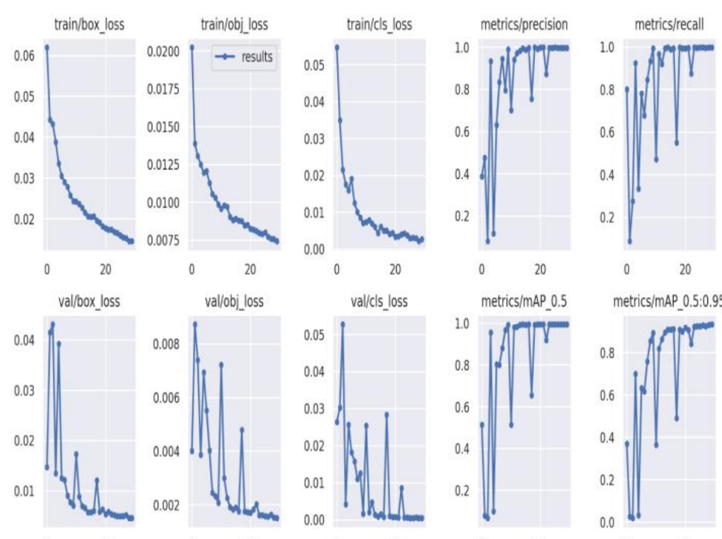

Fig. -3: Losses and evaluation metrics for 30 epochs on train data

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 04 | Apr 2022 www.irjet.net p-ISSN: 2395-0072

Table -1: YOLOv5 calculations on train data

Denomination	Precision	Recall	mAP
10	0.998	0.997	0.968
20	0.997	1.00	0.967
50	0.998	0.996	0.973
100	0.997	0.998	0.971
200	1.00	0.997	0.969
500	0.986	0.976	0.963
2000	0.973	0.941	0.961

3.3 Web Application

The web app design is subdivided into front-end and back-end design. The web application detects Indian notes with accuracy >90% and provides speech out as audio file in English and Hindi languages.

Front end design: In front end, web page is formatted and designed using web technologies. The webpage layout is designed using HTML. Then, buttons are provided to upload and send images to the server. These buttons are styled using CSS and they forward image to API from local system. CSS is used to style webpage. JavaScript makes the page responsive to both mobile and laptop.

Back-end design: It involves object detection of currency notes. Back-end server is built using Flask in python [13]. The server generates routes for different webpages and API call. When an image is sent by user, using Send Button, the Ajax call is made and pre-trained model runs on the sent image. The obtained results comprise bounding box and its probability along with class label. These results from API call are forwarded back to render the webpage. Then, the labels are generated for the sent image. The speech out files are produced using gTTS in back-end. Finally, using play button, the user can hear the audio generated file of detected label of the sent image, both in English and Hindi languages [14].

4. RESULTS

4.1 YOLOv5 Model

The YOLOv5 detected results for test data images (single and multiple class) are presented in tabular format.

Table -2: Test data results for proposed YOLOv5 model

Denomination	Notes correctly detected (out of 10)	Highest Bounding box probability	Least Bounding box probability
10	8	0.96	0.82
20	9	0.95	0.93
50	9	0.95	0.83
100	10	0.92	0.84
200	9	0.96	0.83
500	8	0.96	0.87
2000	7	0.93	0.80

The model was tested against images with 3 classes and 4 classes, each containing 10 images. The model detected 8 correctly in 3 classes and 7 correctly in 4 classes.

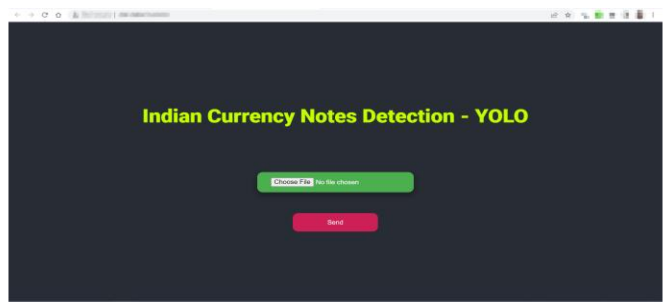

Fig -4: Web Application Design on Laptop

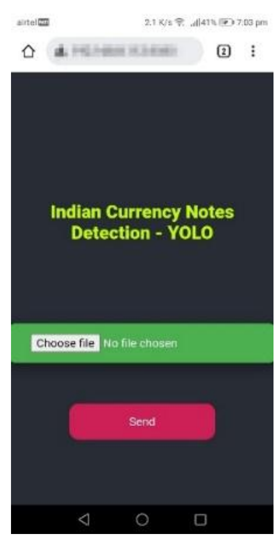

Fig -5: Web Application Design on Mobile

Flowcharts for YOLOv5 and web app design are shown in Fig. 6.

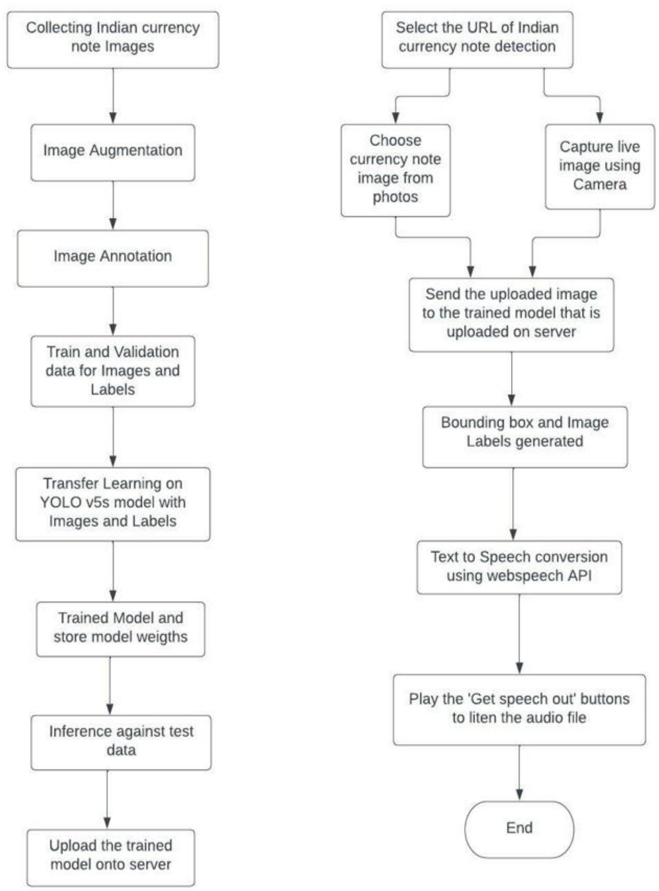

Fig -6: Flowchart for YOLO and web app design

4.2 Web Application

The designed web app detects currency note images with bounding box probability greater than 0.90. The results inferenced against test data containing single class are shown in Fig 7, 8, 11, 12 and 13. The results for multiple class images are shown in Fig 9, 10 and 14.

The results obtained on laptop for single class and multi-class currency note images are shown below.

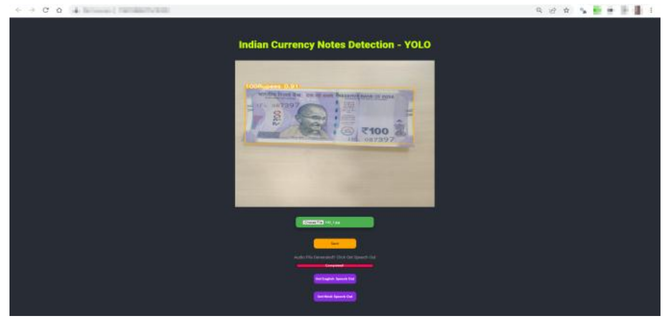

Fig -7: 100 rupees note detected

Fig -8: 2000 rupees note detected

Multiple class image:

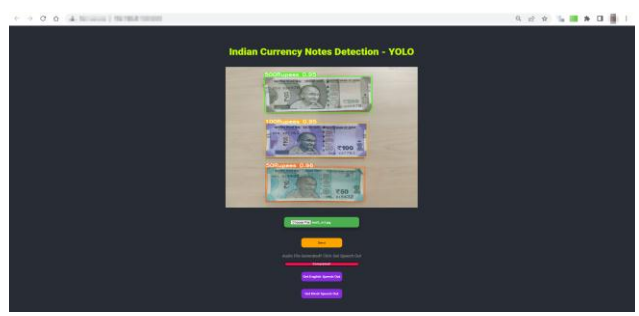

Fig -9: Detected □500, □100 and □50 notes

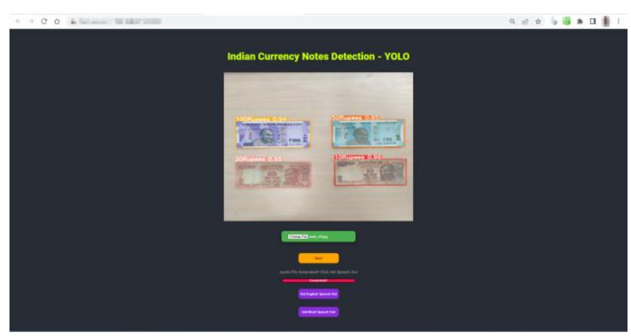

Fig -10: Detected □100, □20, □50 and □10 notes

The results obtained on mobile phone for single-class and multi-class currency note images are shown below.

Fig -11: 20 rupees note detected

Fig -12: 500 rupees note detected

Fig -13: 50 rupees note detected

Multiple class image:

Fig -14: Detected ₹2000, ₹500, ₹100 and ₹200 notes

5. CONCLUSION AND FUTURE SCOPE

The proposed YOLOv5 model detects currency notes with high precision, recall and mAP. The model detected 85 images accurately, out of 100 test data images. A native YOLOv5 based web app was designed that detects currency notes with high bounding box probability. The web app provides speech out files of detected currency note labels in English as well as Hindi.

The model can be extended to detect foreign currency notes. Optical Character Recognition (OCR) can be introduced in the system to enhance the model performance.

REFERENCES

[1] J. G. Shanahan and L. Dai, "Realtime object detection via deep learning-based pipelines," *Int. Conf. Inf. Knowl. Manag. Proc.*, pp. 2977–2978, 2019, doi: 10.1145/3357384.3360320.

[2] World Health Organization;, "Blindness and vision impairment: WHO Factsheet ," *World Heal. Organ.*, vol. 11, no. 2018, pp. 1–5, 2021, [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

[3] P. Ackland, S. Resnikoff, and R. Bourne, "World blindness and visual impairment: Despite many successes, the problem is growing," *Community Eye Heal. J.*, vol. 30, no. 100, pp. 71–73, 2018.

[4] S. D. Achar, C. Shankar Singh, C. S. Sumanth Rao, K. Pavana Narayana, and A. Dasare, "Indian Currency Recognition System Using CNN and Comparison with YOLOv5," *2021 IEEE Int. Conf. Mob. Networks Wirel. Commun. ICMNWC 2021*, pp. 1–6, 2021, doi: 10.1109/ICMNWC52512.2021.9688513.

[5] H. Hassanpour, A. Yaseri, and G. Ardeshiri, "Feature extraction for paper currency recognition," *2007 9th Int. Symp. Signal Process. its Appl. ISSPA 2007, Proc.*, pp. 7–10, 2007, doi: 10.1109/ISSPA.2007.4555366.

[6] S. Luo and W. Zheng, "You-Only-Look-Once-v5 Based Table Detection for Academic Papers," *2021 Int. Conf. Digit. Soc. Intell. Syst. DSInS 2021*, pp. 53–56, 2021, doi: 10.1109/DSInS54396.2021.9670600.

[7] K. K. S. N. Reddy, C. Yashwanth, S. H. Kvs, P. A. T. V. Sai, and S. Khetarpaul, "Object and Currency Detection with Audio Feedback for Visually Impaired," *2020 IEEE Reg. 10 Symp. TENSYMP 2020*, no. June, pp. 1152–1155, 2020, doi: 10.1109/TENSYMP50017.2020.9230687.

[8] P. Shreya, N. Shreyas, D. Pushya, and N. Uma Maheswar Reddy, "BLIND ASSIST : A One Stop Mobile Application for the Visually Impaired," *2021 IEEE Pune Sect. Int. Conf. PuneCon 2021*, pp. 1–4, 2021, doi: 10.1109/PuneCon52575.2021.9686476.

[9] R. R. M. Et al., "Currency Detection for Visually Impaired Iraqi Banknote as a Study Case," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 6, pp. 2940–2948, 2021, doi: 10.17762/turcomat.v12i6.6078.

[10] M. Mahendru and S. K. Dubey, "Real time object detection with audio feedback using Yolo vs. Yolo_V3," *Proc. Conflu. 2021 11th Int. Conf. Cloud Comput. Data Sci. Eng.*, pp. 734–740, 2021, doi: 10.1109/Confluence51648.2021.9377064.

[11] R. Padilla, S. L. Netto, and E. A. B. Da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," *Int. Conf. Syst. Signals, Image Process.*, vol. 2020-July, pp. 237–242, 2020, doi: 10.1109/IWSSIP48289.2020.9145130.

[12] M. Kasper-Eulaers, N. Hahn, P. E. Kummervold, S. Berger, T. Sebulonsen, and Ø. Myrland, "Short communication: Detecting heavy goods vehicles in rest areas in winter conditions using YOLOv5," *Algorithms*, vol. 14, no. 4, 2021, doi: 10.3390/a14040114.

[13] R. Swami, S. Khurana, S. Singh, S. Thakur, and P. K. R. Sajjala, "Indian Currency Classification Using Deep Learning Techniques," *2022 12th Int. Conf. Cloud Comput. Data Sci. Eng.*, pp. 567–572, 2022, doi: 10.1109/confluence52989.2022.9734149.

[14] R. C. Joshi, S. Yadav, and M. K. Dutta, "YOLO-v3 Based Currency Detection and Recognition System for Visually Impaired Persons," *2020 Int. Conf. Contemp. Comput. Appl. IC3A 2020*, pp. 280–285, 2020, doi: 10.1109/IC3A48958.2020.233314.