iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** III **Month of publication:** March 2025

DOI: https://doi.org/10.22214/ijraset.2025.68026

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

"Smart Vision: Real-Time Object Detection and Audio Assistance for the Visually Impaired Using TensorFlow and SSD MobileNet"

Diksha R. Pawar[1], Pravin L. Yannawar[2]
[1]*Research Student,* [2]*Professor, Dr. Babasaheb Ambedkar Marathwada University, Chh. Sambhajinagar, India*

Abstract: ***Visual impairment affects approximately 8.5 billion people worldwide, limiting their ability to interpret visual information daily. To address this challenge, we propose an Android-based object detection application, "Digital Eyes," which assists visually impaired individuals by detecting objects and converting them into audio feedback. The application utilizes TensorFlow's SSD MobileNet model for real-time object recognition. It identifies 10 categories of objects, including chairs, bottles, laptops, and persons, and provides spoken feedback via Text-to-Speech (TTS) technology. The system performs all tasks locally, eliminating the need for an internet connection and making it accessible on most smartphones. SSD MobileNet was chosen due to its balance between speed and accuracy, achieving a mean Average Precision (mAP) of ~74 and a frame rate of 59 FPS, making it suitable for real-time applications. The proposed system enhances the independence of visually impaired users by identifying objects and reading text from the environment. The results demonstrate high accuracy in object detection and reliable audio feedback, enabling a smoother user experience.***

Keywords: Object Detection, TensorFlow, Text-to-Speech, Android, SSD MobileNet.

I. INTRODUCTION

According to a World Health Organization (WHO) research that was recently published, there are an estimated 8.5 billion individuals worldwide who suffer from visual impairment, which of 3.9 billion and 24.6 are blind. Billion have limited vision [1]. Dealing with vision loss or low vision is just one of the difficulties that the visually impaired face in everyday life. Compared to all senses of the human eyesight is the most important sense. One quick look around us tells how visual most of the information in our surroundings is, eye problems can disturb a person's regular activities. Identifying items in everyday life, reading text, and crossing a street are a few cases of such issues. Timetables in train stations, signs showing the right way or potential risk, and billboards advertising a new product on the market are all examples of visual information that we all see in our daily lives [2]. The majority of this information is inaccessible to the blind and visually impaired. The proposed system is a basic object detection application based on an Android app called "Digital Eyes" that can help those who are visually impaired. Using object detection, this app attempts to simulate the human eye using a smartphone camera. Using advanced computer vision techniques, people's daily lives can be improved. Object detection is one of the computer vision algorithms that has seen many expanded applications in recent years. To recognize an object from an input image, object recognition technology uses contrasting properties such as intensity, edge, and shape. Because of advancements in object detection algorithms, we can now incorporate complicated algorithms into Android applications. For object detection in our Android application, the trained TensorFlow models and SSD techniques are used. 10 categories can be identified using the objects; however, the scope can be expanded using a newly trained model. The 10 categories include, [Chair, Bottle, Mouse, Laptop, Books, Potted Plants, Person, Watch, and Mobile]. The application can be used easily on a mobile phone or any other smartphone with minimum computational resources, i.e., no Internet connectivity. There are issues with the application's illumination and speed [3]. Real-time detection will be made easier with effective outcomes. This is related to the similar intelligence found in cameras, which may be utilized for a variety of purposes, including surveillance, traffic, and robotics. The goal of this research is to detect objects with impaired vision by utilizing audio feedback and extracting information from the actual video feeds. It's easy to use the application, and includes speech synthesizing, allowing the identified object to be communicated to blind persons via voice output.

II. LITERATURE SURVEY

According to the below information, it is recognized that the YOLO and SSD models support real-time processing with Fps values around 46 and 59, respectively, making them suitable for applications that require fast object detection in real-time scenarios.

©IJRASET: All Rights are Reserved

Table 1. Related work using several algorithms

Model type	Accuracy	Latency of Model	Fps	Real-Time
SSD	~74	Low	59	YES
YOLO	~60	Low	46	YES
R-CNN	~60	High	<1	NO
Faster-RCNN	~70	High	7	NO
Fast R-CNN	~70	High	<1	NO

The above table displays the comparison of several average accuracies (MAP) models, per second image (fps) latency of the model, and after this, whether they are suitable for Applications that run in real-time or those that don't. Fast R-CNN, Faster R-CNN, and Real-Time R-CNN on the other hand have higher latencies and lower FPS values, indicating that they might not be suitable for real-time applications [4]. It's important to note that the model choice depends on your application's specific requirements. Higher accuracy might be more critical for some situations, while real-time processing might be a priority in others. Additionally, model performance can vary based on hardware and optimization techniques. Always consider a trade-off between accuracy and speed based on your application's needs.

Table 2. Recent Related Work and Goal

Sr. No	Paper Name and year	Author	User specified	Category of Identification and Detection	Goal
1	Detecting the objects on the road using modular lightweight network.[2018]	Cao, S., Liu, Y.	the user of the road	Car& Person	Identify and locate on-road objects, such as bicycles, cars, and people [5]
2	Object Detection in Refrigerators using Tensorflow [2018]	Agrawal	Everybody	food from the supermarket	Recognize items kept in refrigerators and inform the user whether the item is likely to be found there [6].
3	Classification of Vegetables using TensorFlow [2018]	Patil & Gaikwad	cashier at a supermarket	Vegetables	Helping the checkout staff identify the types of vegetables that clients have purchased [7].
4	TensorFlow: A Vegetable Classification System and Its Performance Evaluation [2017]	Ruedeenir amana, Ikeda and Barolli	Farmers	Vegetables	For helping farmers in categorizing varieties of vegetables [8]
5	FoodTracker: A Real-time Food Detection Mobile Application by Deep Convolutional Neural Networks [2019]	Sun, Radecka, and Zilic	Any individual who is concerned about their health	Food	Identify food and provide the user with nutrition information [9]
6	A Mobile Application for Cat Detection and Breed Recognition Based on Deep Learning [2019]	Zhang, Yang and Sinnott	Everyone	Cat	Sort the various cat species.[10]

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The R-CNN [4] approach to object detection, according to the results, was significantly quicker than the earlier methods based on classification techniques. The selected search was used by RCNN to obtain only 2000 regions in each frame as opposed to a large number of regions. Thus, only 2000 regions would be covered by the feature extraction process. R-CNN fast, a new variant, Since R-CNN didn't send CNN 2000 regional offers every time, it was much better than R-CNN. The CNN process was instead performed once every frame. The new approach was put into practice in a way that was comparable to the old ways, but for identifying the suggested regions, an independent network was employed rather than a selective search algorithm. In terms of accuracy, the SSD was extremely comparable to R-CNN. Due to its ability to balance speed and accuracy, the SSD is now the best algorithm [11]. As a result, object-based detection systems frequently use the SDD algorithm. The recognition of the objects was improved with the new technique You Only Look Once (YOLO). The methods mentioned above never take into account the complete image because they only look at the specified regions to identify the objects in the image. Instead, the technology was used to detect objects in areas where there was a high possibility of their presence. However, with YOLO, a single convolutional network was used to assess the entire image [12].

III. SYSTEM DESCRIPTION

A. Object detection

The purpose of detection of the object is to find complete occurrences of things from a well-known class in an image or video, such as people, automobiles, or faces [2]. Visually challenged individuals who use object detection can interpret their surroundings without difficulty and stay independent of others.

Input: an image with one or several items, such as a photo.

Output: 1 or additional limiting boxes, each with its class name (for instance, those determined by spot, size, and height).

B. Tensor Flow

TensorFlow is a framework for a software library that has been used to develop the Identification and recognition of objects. An already-trained object makes up this identification of a model that uses an SSD technique to recognize things that work more effectively and precisely. The mobile net SSD v1 model is used in this object detection strategy, which also includes datasets of 80 regularly seen object types.

C. Android Studio

The use of the Android SDK is to create the application of Android, which visually challenged users may easily use to identify things and understand their surroundings. This platform is used to implement the application's front end and back end. To operate the system, this platform comes with all the necessary libraries and packages

D. Mobile device-based object RECOGNITION system

Numerous people have attempted to implement object recognition on mobile phones [13]. Because of the rapidly advancing technology of these devices. Smartphones make it possible to create user-friendly, portable, and widely accessible blind-specific applications, doing away with the requirement for specialized processing machinery. However, some of these applications use a client-server architecture due to a mobile phone's limited computing capability. One such popular app is Google Goggles, which requires an internet connection and is limited in its ability to add new images to the applications for blind users that simply make use of the local processing on a smartphone. Computing power. For example, in this paper, an Android app has been created that completes all tasks locally and provides feedback in the form of audio.

IV. TRAINING OBJECTIVE

The MultiBox objective is the source of the SSD training objective [14]. Which has been expanded to support several item categories. Let $_{x}^{p}ij = \{1, 0\}$ indicator for comparing the category p's i-th default box to its ^{j}th ground truth box. As a result of the aforementioned similar method, we can $\sum_{i} x^{p}ij \geq 1$. The localization loss (loc) and the confidence loss (conf), when added together, form the overall objective loss function:

$$\boldsymbol{L(x, c, l, g)} = \frac{1}{N}(\mathbf{L}\boldsymbol{con}_{f(x,c)+} \boldsymbol{\alpha L}_{\boldsymbol{loc}(x,l,g))} \quad (1)$$

N stands for the number of default boxes that were found. The same if the loss was set to 0 by wet and N=0. Smooth L1 loss characterizes localization loss between the box's (g) ground truth parameters and (l) projected parameters. We regress to offsets for the default bounding box's width (w) and height (h), as well as its center (cx, cy).

The Localization loss between the Predicted box l and the ground truth box g is defined as the smooth L1 loss with cx,xy as the offset to the default bounding box d of width w and height h

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx, cy, w, h\}} x_{ijsmoothL1(l_{i-\hat{g}_j^m})}^{k}$$

$$\hat{g}_{j=(g_j^{cx}-d_i^{cx})/d_i^w}^{cx} \qquad \hat{g}_{j=(g_j^{cy}-d_i^{cy})/d_i^h}^{cy}$$

$$\hat{g}_j^w = \log\left(\frac{g}{d}\frac{j^w}{i_w}\right) \qquad \hat{g}_j^h = \log\left(\frac{g}{d}\frac{j^h}{i_h}\right)$$

$x_{ij=\{1}^{p}$ If IOU>0.5 between default box I and the ground true box j on class p (2)

$_{0}$

Otherwise.

The loss of confidence is calculated as the SoftMax loss over all classes of confidences (c).

$$\mathbf{Lcon_f(x, c)} = -\sum_{i \in Pos}^{N} x_{ijlog(\hat{c}_i^p)} - \sum_{i \in Neg} \log(\hat{c}_i^o) \quad where \quad \hat{c}^p_{i=\frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}} \tag{3}$$

Where N is the number of match default boxes.

Where N represents all of the successful matches and α is the localization loss weight.

V. PROPOSED SYSTEM

The system was built on Android software that recognizes various items in a live video feed and includes a text reader in real time. In the proposed system, a real-time text reader function that takes advantage of the Google Play Services mobile vision and TTS engine using a text recognizer class to extract text from a live video feed is described in API [15][16]. The object identification API model from Google's TensorFlow was used to create the real-time text reader feature, which was implemented using the SSD algorithm. For both real-time and offline object detection, an SSD algorithm-based model was used.

A. System Design

The system captures incoming data in real-time video feed using a cell phone. The user has two options in the application for finding things. The application's camera automatically opens and starts to take pictures of the surroundings and objects. For processing, data is given to the TensorFlow object detection model, and after it has finished, it classifies the items it has found and returns the outcomes in the form of spoken feedback. When reading text, it makes use of the Text-Recognizer class in the Google Play services mobile vision API to identify text that feeds the real-time video stream to Google's TTS (text-to-speech) engine, which converts it into speech and reads out the text.

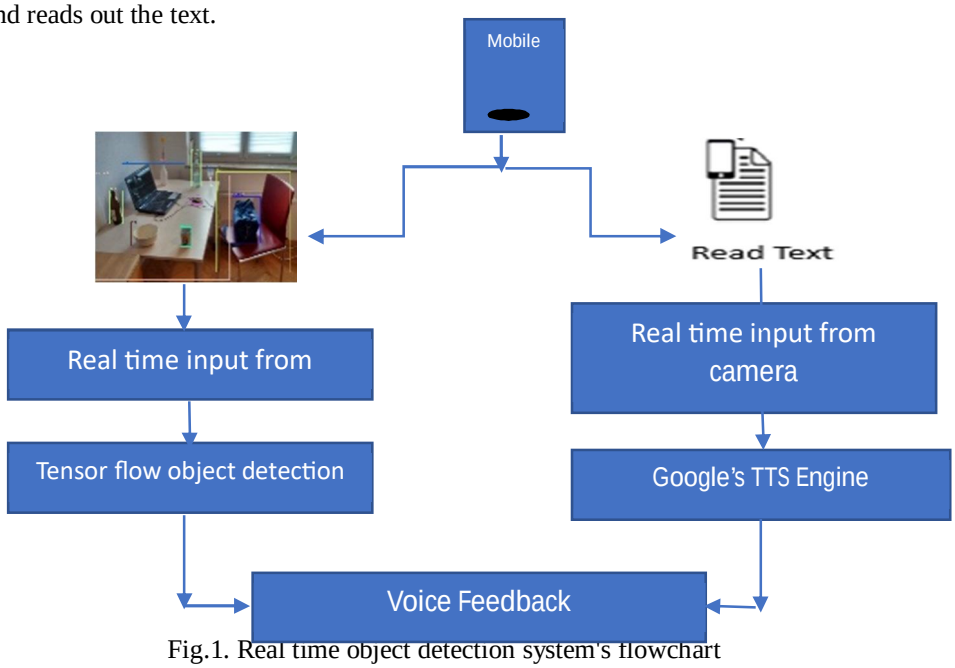


Fig.1. Real time object detection system's flowchart

B. Implementation

The following technologies were integrated into the system during development. The application was produced using the Android SDK, which is the officially designated Integrated development environment (IDE) made especially for creating Android applications [3]. Through Android hardware, the Android framework offers photo and video capture. Camera Intent or the camera2 API. It is a tool for reading text and recording real-time video of object recognition.

The object detection models in the Android application are implemented using the TensorFlow library. High-performance numerical computing is offered. Due to its flexible architecture, it is simple to implement the calculation across any platform [4]. SSD-Mobile-Net for real-time processing, the Custom object detection model was used. A single neural network that teaches to predict the locations of bounding boxes makes up the SSD architecture. The system employs a real-time text reader and two object detector modules.

1) Object Detection: In this, the SSD- MobileNet model is used in the app for detecting objects. For the entire input image, only one neural network is used. After breaking the real-time input image into several pieces, the network model predicts the items by applying a quadrilateral that contains the object and its probability score [3].

2) Text Recognizer: By analyzing the images, this object determines the text that is displayed there. Once configured, you can use it to search for text in any form of image. The implementation of the reading text feature made use of Google Text-to-Speech, which speaks to the recognized items and detected text [16][17].

C. Dataset

Custom-object Detection is a project-specific dataset that was used to train the SSD MobileNet model, which was capable of classifying objects into 80 various categories [4]. These classes cover a broad range of frequently occurring objects that can be seen in daily life, including people, animals, automobiles, household items, and more. We employ trained TensorFlow models and the SSD method in our Android application to identify objects. 10 categories can be identified using the objects; however, the scope can be expanded using a newly trained model The 10 categories include, Chair, Bottle, Mouse, Laptop, Books, Potted Plants, Person, Watch, Mobile, and Keyboard. Because there are so many different classes, it is possible to create models that can identify and categorize a wide range of things. Images from a variety of sources, including, photos from the internet, and images taken in the real world, are collected in the Custom dataset.

VI. ANALYSIS OF THE SYSTEM

TensorFlow Object identification model, which in this case was successful in finding a balance between speed and precision using an SSD method, about 80 items are correctly detected by this model. After this idea was put into action, a speech output for the object being spotted was expected. After testing on a variety of things, we discovered that the outcomes can occasionally vary and that the accuracy of recognizing objects depends on several things [18].

Here are a few findings that demonstrate the predictive power of successfully recognizing an objects:

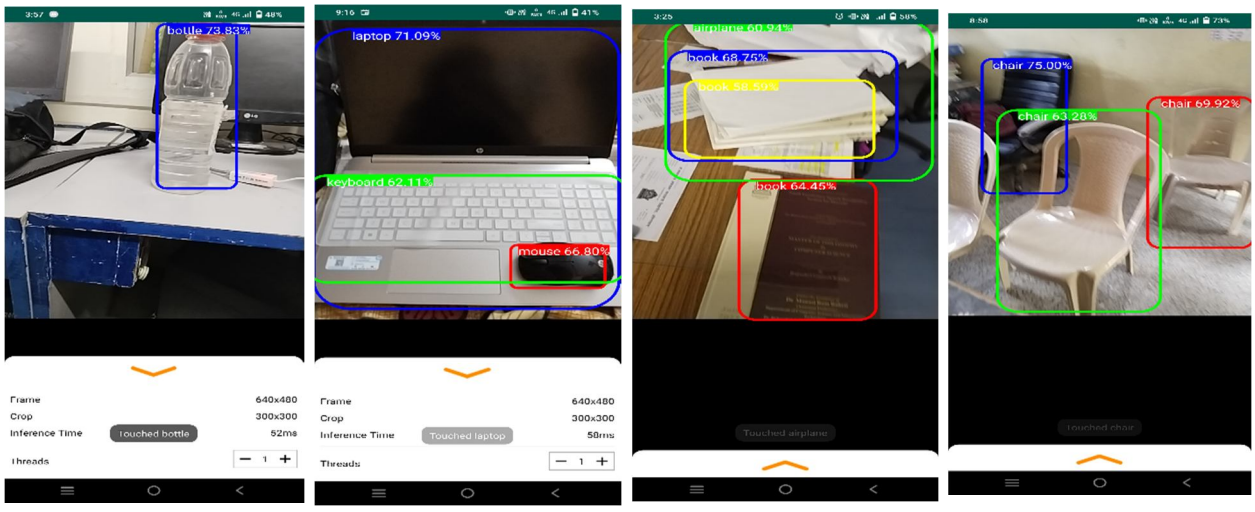


Fig.2. Real Time object detection

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue III Mar 2025- Available at www.ijraset.com

VII. RESULT & DISCUSSION

In the research ,80 different types of objects study in my custom dataset that was used to create this model for improvement of this model, we had categorized Objects into 10 distinct Classes few of the examples are “Chair”, “ Bottle” , “Mouse” “ Laptop”, “Books”, “Potted Plants” , “Person”, “Watch” ,“Mobile”, and “Keyboard” We used 8 GB RAM So the accuracy of the model is good and like this if there is increment in it, the accuracy of the model is quite better and additionally many more images can be added to the dataset . The outcome of this project displays the objects that were detected, encircled by a rectangle box, with a label displaying the name of the object as well as the percentage of detection accuracy on top. It is capable of identifying any number of elements that are visible in a single image. This model's accuracy is good because the IoU score is more than 0.5, The suggested approach will be capable of identifying what is seen to the camera and turning it into an audio files. The proposed system's accuracy is given in the below:

Table 3. Testing for Object Detection

Class Input	No. of Input Class given	No of times Correctly detected	Accuracy of Model
Bottle	10	07	70
Laptop	10	09	90
Mouse	10	06	60
Keyboard	10	07	70
Person	10	09	90
Books	10	07	70
Mobile	10	08	80
Potted Plant	10	05	50
Chair	10	08	80
Clock	10	06	60

Figure.3. shows a sample output of detected items with bounding boxes, corresponding labels, and accuracy scores. These pictures were captured using an Android phone and the object-detecting app.

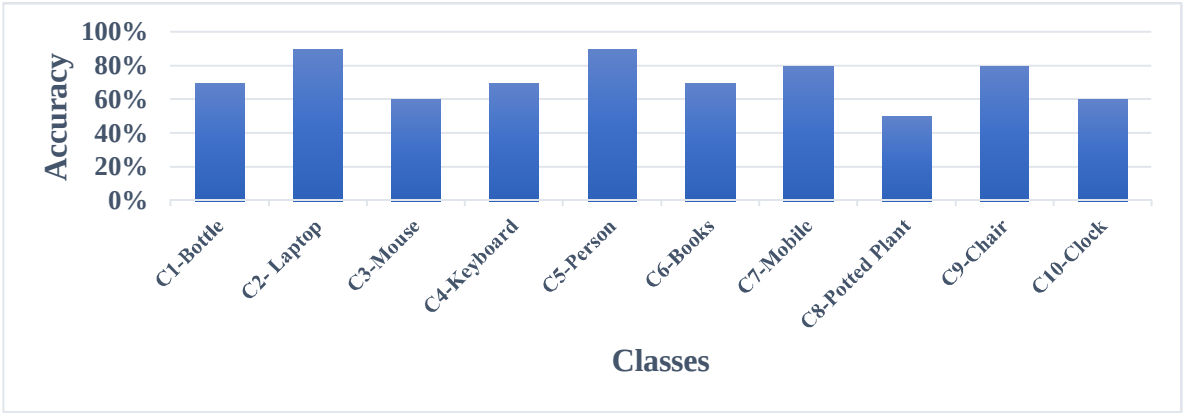


Fig. 3. Sample results for object detection collected from an Android app on a

VIII. CONCLUSION

In this study, an object detection application was developed using a model that applied the SSD method. It worked offline and provided the highest level of accuracy possible. To identify items, an object detection API was used. Future work will focus on improving the model's effectiveness by training it on a huge No of images, developing live stream picture recognition and capture, and increasing the number of steps it goes through during training to get better outcomes.

©IJRASET: All Rights are Reserved

For those who are blind, the voice synthesis in this system offers practical functions. A mobile-compatible object identification model for visually challenged people was developed using the TensorFlow lite module. Further stability and functionality improvements can be made to the Android app.

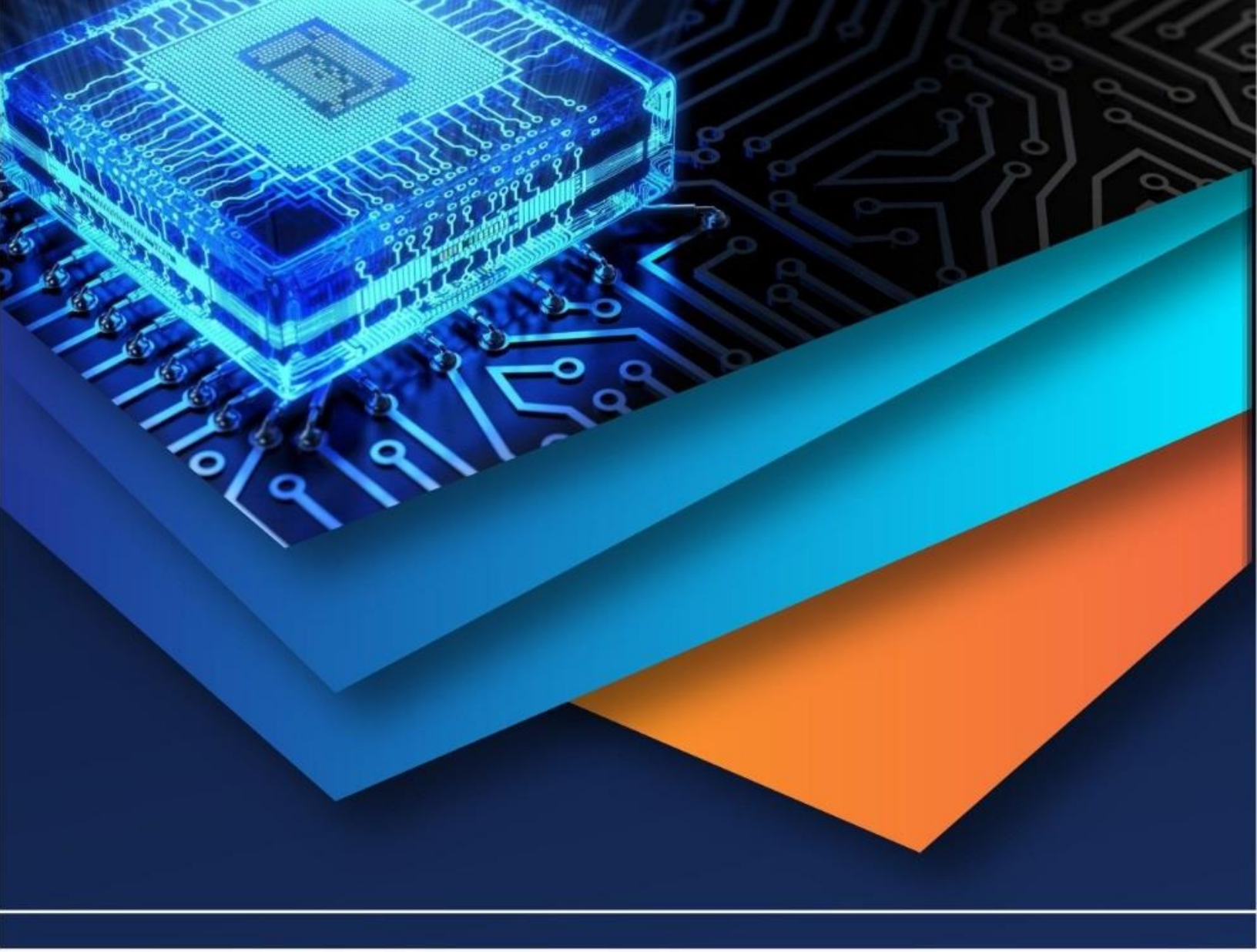
IX. FUTURE SCOPE

In the future, we want to try several things to improve the application's value to users who are blind or visually impaired. First, in the future, we plan to expand support for more languages. At the moment, only English is supported. To read texts about medications, newspapers, etc., we also wish to provide support for OCR. They can use this to look for specific medications or read documents that are not written in Braille. Finally, we aim to develop a smaller gadget that will allow us to incorporate a camera inside the user's eyewear. This camera will transmit a live feed to a special device that will perform the computing.

REFERENCES

[1] Aralikatti, A., Appalla, J., Kushal, S., Naveen, G. S., Lokesh, S., & Jayasri, B. S. (2020, December). Real-time object detection and face recognition system to assist the visually impaired. In Journal of Physics: Conference Series (Vol. 1706, No. 1, p. 012149). IOP Publishing.
[2] Amit, Y., Felzenszwalb, P., & Girshick, R. (2021). Object detection. In Computer vision: A reference guide (pp. 875-883). Cham: Springer International Publishing.
[3] Mathurabai, B., Maddali, V. P., Devineni, C., Bhukya, I., & Bandari, S. (2022). Object detection using SSD-MobileNet. Int Res J Eng Technol.
[4] Tomar, P., & Haider, S. (2022). A study on real time object detection using deep learning. International Journal of Engineering Research & Technology (IJERT), 11(05).
[5] Cao, S., Liu, Y., Lasang, P., & Shen, S. (2018). Detecting the objects on the road using modular lightweight network. arXiv preprint arXiv:1811.06641.
[6] Agarwal, K. (2018). Object detection in refrigerators using Tensorflow (Doctoral dissertation).
[7] Om Patil, O. P., & Vijay Gaikwad, V. G. (2018). Classification of vegetables using TensorFlow.
[8] Ruedeeniraman, N., Ikeda, M., & Barolli, L. (2020). Tensorflow: a vegetable classification system and its performance evaluation. In Innovative Mobile and Internet Services in Ubiquitous Computing: Proceedings of the 13th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2019) (pp. 132-141). Springer International Publishing.
[9] Sun, J., Radecka, K., & Zilic, Z. (2019). Foodtracker: A real-time food detection mobile application by deep convolutional neural networks. arXiv preprint arXiv:1909.05994
[10] Zhang, X., Yang, L., & Sinnott, R. (2019, February). A mobile application for cat detection and breed recognition based on deep learning. In 2019 IEEE 1st International Workshop on Artificial Intelligence for Mobile (AI4Mobile) (pp. 7-12). IEEE.
[11] Tomar, P., & Haider, S. (2022). A study on real time object detection using deep learning. International Journal of Engineering Research & Technology (IJERT), 11(05).
[12] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
[13] Salunkhe, A., Raut, M., Santra, S., & Bhagwat, S. (2021). Android-based object recognition application for visually impaired. In ITM Web of Conferences (Vol. 40, p. 03001). EDP Sciences.
[14] Jung, K. (2012, September). Object recognition on mobile devices. In 2012 IEEE Second International Conference on Consumer Electronics-Berlin (ICCE-Berlin) (pp. 258-262). IEEE.
[15] Pawar, D. R., & Yannawar, P. (2024). Advancements and Applications of Generative Adversarial Networks: A Comprehensive Review. Int. J. Res. Appl. Sci. Eng. Technol, 12.
[16] Pawar, D., Borde, P., & Yannawar, P. (2024). Generating dynamic lip-syncing using target audio in a multimedia environment. Natural Language Processing Journal, 8, 100084.
[17] Pawar, D. R., & Yannawar, P. (2023, July). Recent advances in audio-visual speech recognition: Deep learning perspective. In First International Conference on Advances in Computer Vision and Artificial Intelligence Technologies (ACVAIT 2022) (pp. 409-421). Atlantis Press.
[18] Younis, A., Shixin, L., Jn, S., & Hai, Z. (2020, January). Real-time object detection using pre-trained deep learning models MobileNet-SSD. In Proceedings of 2020 6th International Conference on Computing and Data Engineering (pp. 44-48).

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)