**Sustainable Camera-Based Support for the Visually Impaired**

**Abstract**

This project aims to enhance the daily lives of visually impaired individuals by providing them with a wearable assistive technology solution. The prototype consists of a pair of smart glasses equipped with various sensors and features to aid navigation, object recognition, and communication. This project is integrated with various domains including Computer Vision (CV), Internet of Things (IOT), Machine Learning, App Development. This device recognises objects, obstacles, traffic and helps the person using the device to make further decisions.

**1. Introduction**

As visually impaired people face many difficulties in doing their day-to-day tasks. It is our responsibility to assist them, we propose an innovative idea using the emerging technologies like Computer Vision (CV) which acts as the eye, serves as the visual perception system, handling all necessary image processing tasks and process it with the help of Machine Learning (ML) to recognize the image and with the integration with Internet of Things (IoT) the signals or the recognized object can be heard by the user. If the idea is implemented it becomes very easy for the visually challenged people to walk around freely without any difficulties or hesitation. They can be cautious about their environment and seek assistance in case of emergencies when needed.

Our contributions to this paper are as follows:

• A literature review is made to identify methods used in this project.

• The working and ways to implement it.

• Summarize and conclude the daily uses of the device.

**2. Literature review**

This section presents a summary of the relevant resources that contributed to the project's development.

Deeshu et al. [1] explored the features of Analysis of object detection in OpenCV python. This paper provides an overview of object detection, face recognition and vehicle detection. This object detection can be done from the images or live video stream using multiple software like MATLAB, OpenCV using C++, OpenCV using Python etc. MATLAB provides powerful matrix library, toolboxes, debugging tools etc, yet it faces challenges in terms of accuracy and efficiency. In this research, object detection has been implemented using OpenCV and Python which help in improving accuracy and efficiency.

Ankita Rameshwar Mahajan et al. [2] conducted research on Movement Detection using OpenCV. This paper provides an overview of detection of any movement occurred in front of a camera. In the paper, the proposed method used OpenCV2 and Tracker, where OpenCV2 has been utilized for detecting movement of individuals or surrounding movement. Movement detection is technique of detecting occurrence of any movement within the camera's field of view. The article proposes the idea of which will count the movement of the object or person.

K. Vaishnavi et al. [3] conducted a study on the object detection linked to video and image analysis. This research offers a solution to the challenges encountered in the traditional methods such as relying on other computer vision methods to support deep learning, which in turn leads to slow performance. This paper presents a solution for object detection from a image using single layer of convolution network.

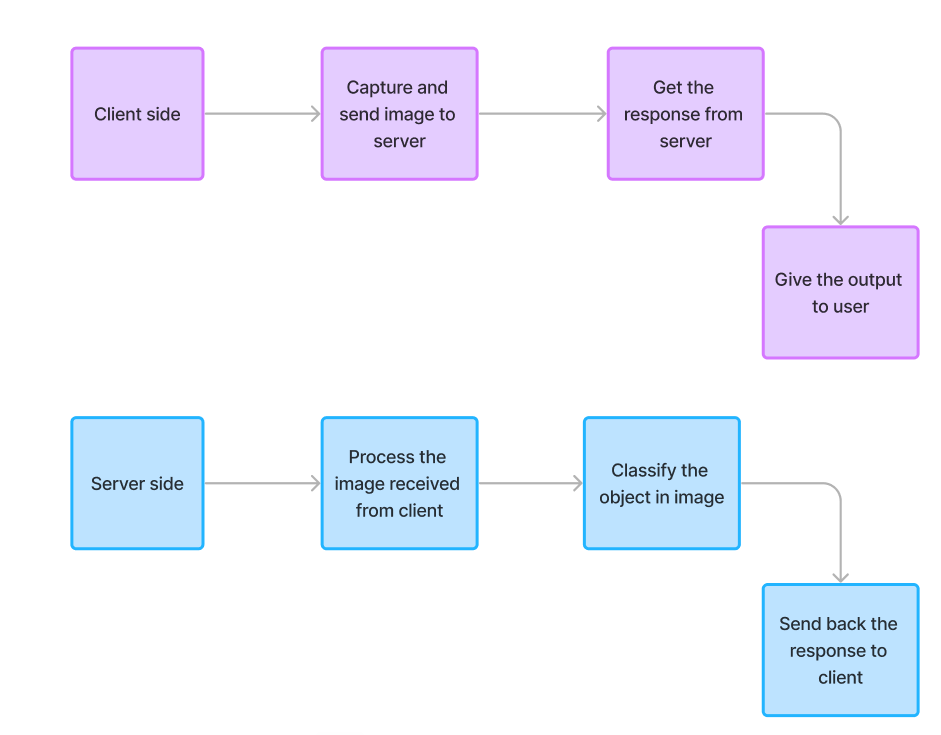
Yann LeCun et al. [4] researched Convolutional networks for images, speech, and time series. This paper provides a study on the utilization of back-propagation networks to learn complex, high-dimensional, nonlinear mappings from extensive datasets, making them enable for image recognition and speech recognition tasks. This paper introduces the concept of using multilayer networks as classifiers, and eliminate the features extractor by feeding the network with raw inputs and rely on backpropagation to transform initial layers into effective feature extractors.

Ferdin Joe John Joseph et al. [5] explored the features of Keras and TensorFlow: A Hands-on Experience. This offers a practical training experience with Keras in the TensorFlow library using Jupyter Notebook for Python. The use of Keras and TensorFlow has been explained to simplify the prerequisite knowledge required for working with Keras.

Nuruldelmia Idris et al. [6] conducted a study exploring a Generic Review of Web Technology: DJango and Flask. This study provided a concise overview of backend development in both web and app development contexts.

**2.1 Methodology**

This section provides the methodology to build the project of creating a glasses for visually impaired people. Figure 1 shows the conceptual framework for this project.



**Fig 1**: Conceptual frame work

**2.1.1 Establishing connection of client and server**

To get started we need to first connect the glasses with the application. The glasses act as the client and it is to be connected to server which is present in the application to which the glasses are paired. To establish the connection between client and server, Flask library [] in python is used. With the help of the host IP address the client is connected to it and requests for the response from the server by sending the image. The server accepts the request and gives the response to the client.

**Algorithm-1:** client and server communication

1. Begin

2. Importing request from flask

3. Send the captured image to URL of the host

4. Receive the response from server

5. Repeat till interrupted

Algorithm-1 represents how the client-side work is done to send images to server and get the response. The URL is the IP address generated by the host along with the port of connection.

Algorithm-2 represents how to generate the host IP using flask. After the IP is generated, the client can be connected and the communication can be done.

**Algorithm-2:** Generating host IP

1. Begin

2. Importing Flask, jsonify, request from flask

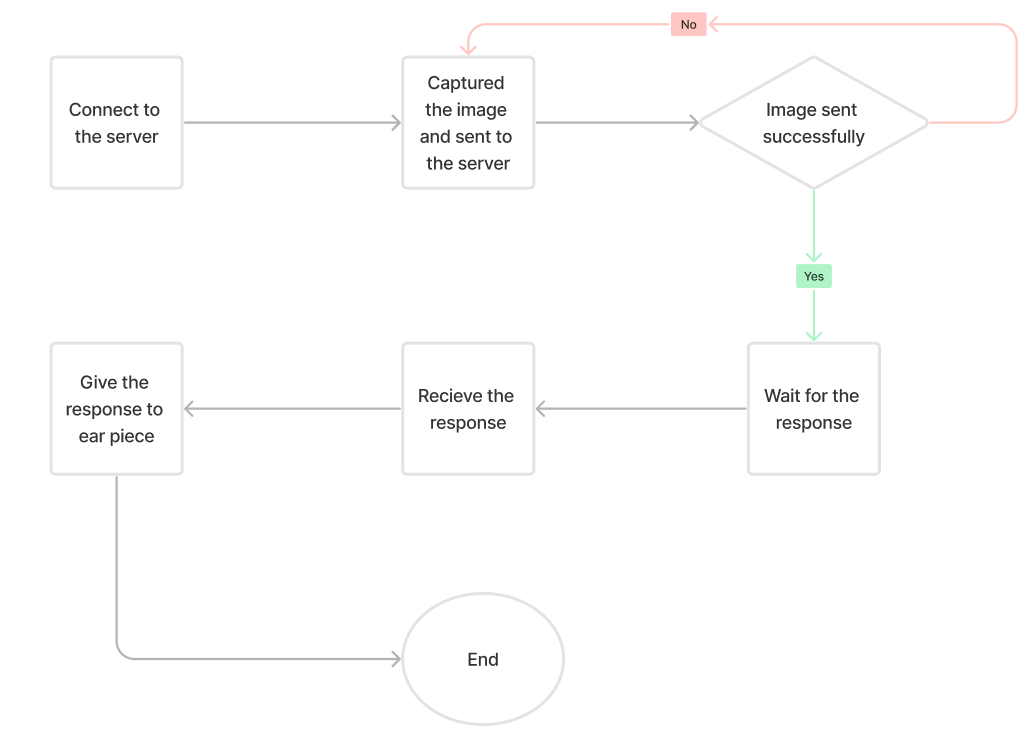
3. Creating Flask app

4. Root the app with Post method

5.Process the other work flow

**2.1.2 Capturing images at the client level**

With the help of IoT all the client level work is done. The core heart of the glasses is the micro controller used. Using the efficient and powerful controller is important along with its compact size because the client code runs in it and it must connect with the server with Wi-Fi and Bluetooth. According to us at present the suitable controller which can be used is Raspberry pi. A small camera is connected to it which captures the images in front of the person. The Raspberry pi reads the image and sends it to the server over Wi-Fi.

****

**Fig-2:** Work flow of the client-side

**2.1.3 Generating model for classification**

There are many objects around us which needs to be classified. Starting with the most common objects like obstacles, vehicles, humans, animals and traffic we collect the dataset of these images. The dataset is then cleaned and all the images are processed to have same size and pixels. On around the dataset consists of 20,000 images which is divided in training set and test set. Training set consists of around 70% of the images and the remaining 30% as test set. After processing the dataset EfficientNetB3 algorithm is used to classify the images in different classes and the generated model is saved with .h5 extension. This model is further used in server to predict the image sent by the client. EfficientNet is a Convolutional Neural Network [6] architecture that uses compound scaling method to improve performance. It uses a compound coefficient to scale up models in a simple but effective manner. It has ability to find a compromise between accuracy and processing cost.

EfficientNet has two parts:

* Creating an efficient baseline architecture using NAS.
* Use the compound scaling methos while scaling up to enhance performance.

**2.1.4 Processing in server**

Server is the back bone for this project which runs in the background of the application. All the processing and prediction part of the projects is done in the server. The images received from the client are preprocessed using OpenCV [1] according to the requirements of the trained model. Then the with the help of Keras and TensorFlow [5] the processed image is predicted by the model. The prediction of the image is given as the index along with confidence of the prediction. The class corresponding to the index is labeled and a response dictionary is created containing class and confidence. With the help of jsonify the response dictionary is sent to the client.

**Algorithm-3:** Processing in server

1. Begin

2. Importing required libraries

3. Load the pretrained model

4. Preprocess the image received

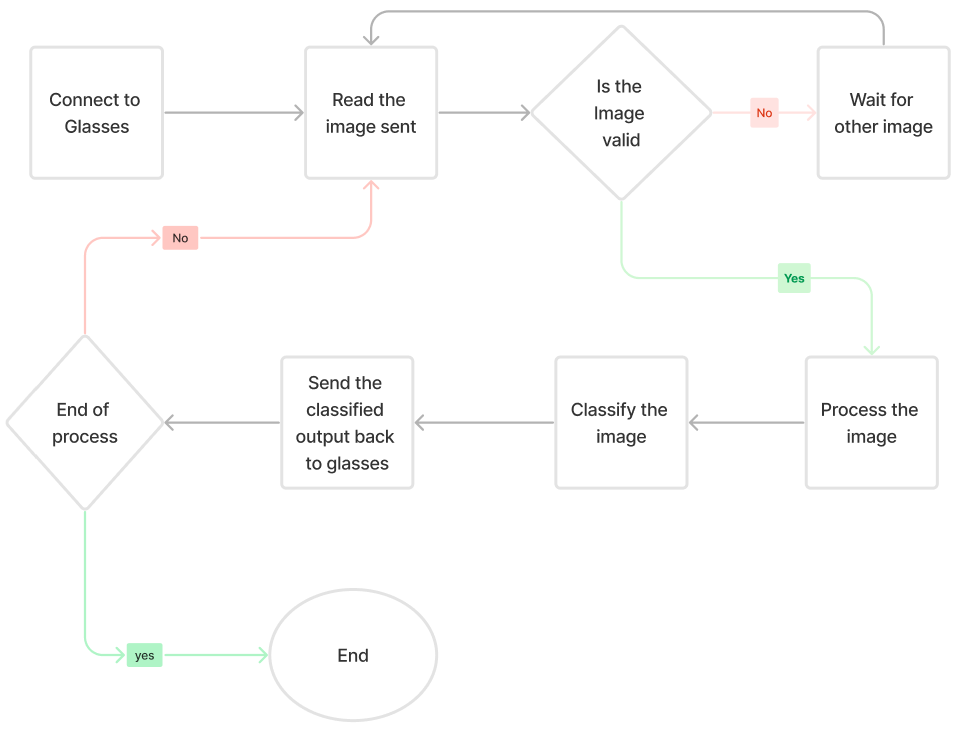
5. Predict the image using model

6. create response dictionary

7. jasonify the response and send to client

8. Repeat till interrupted

Algorithm-3 represents the work flow of the process done by the server for classifying the images and giving response.

****

**Fig-3:** Work flow in server

**2.1.5 Response given to user**

Now we come back to the client-side process where the response received is to be given to the user. Best way to give the output is using speech. The text output received is converted to speech and then given to user. This can be done by an ear piece connected to Raspberry pi. Through the ear piece the user can hear the assistance given by the device to stay alert.

\*Photo of glasses\*

**3. Future scope and conclusion:**

Improving Accuracy is the most import key feature to be developed further to classify the objects more precisely. To classify the multitude of objects, present in the environment the object recognition capabilities must be expanded beyond boundaries. In today's context, processing the things faster is very much crucial. Taking decisions within a fraction of milliseconds requires high processing speed. In this scenario, as number of pictures are processed per second, we require more processing speed and less complexity. Furthermore, integrating a feature to track the user's location using various sensors such as GPS modules could prove invaluable in emergency situations. Considering that the user lacks normal vision, there are many hazardous dangers that demands instant reflexes. Incorporating additional features for detecting the dangers can reduce the risk and increases the safety of the users. Efficient power management is essential as all processes must be powered through the glasses. Optimizing power usage can significantly extend usage time and enhance overall efficiency.

With the help of the proposed system, it makes the visually impaired people to complete their daily tasks with easy to go. One can use these glasses to walk around without any fear of falling or hitting by an obstacle or vehicles. Additionally, they offer the capability to read and comprehend text without visual assistance. This emerging technology holds great potential for societal benefit, offering cost-efficient solutions for noble causes. The system also holds significant market potential, providing opportunities for commercialization and widespread adoption.

**References**

1. Deeshu Akshay Aggarwal Dr. Sonal Sharma Pooja Gupta. (Year June-2019). Analysis Of Object Detection in OpenCV python. Journal of Emerging Technologies and Innovative Research. Volume 6(Issue 5), pp.309-318. [Online]. Available at: https://www.jetir.org/view?paper=JETIR1905N47 [Accessed 17 November 2023].
2. Ankita Rameshwar Mahajan, Vinod Agrawal. (May-2022). MOVEMENT DETECTION USING OPENCV. International Research Journal of Modernization in Engineering Technology and Science. Volume:04(/Issue:05). [Online]. Available at: https://www.irjmets.com/uploadedfiles/paper/issue\_5\_may\_2022/22939/final/fin\_irjmets1652552307.pdf [Accessed 15 November 2023].
3. Srivastava, S., Divekar, A.V., Anilkumar, C. et al. Comparative analysis of deep learning image detection algorithms. J Big Data 8, 66 (2021). https://doi.org/10.1186/s40537-021-00434-w
4. Sakshi Indolia, Anil Kumar Goswami, S.P. Mishra, Pooja Asopa. (2018). Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach. International Conference on Computational Intelligence and Data Science. Volume 132, Pages 679-688. [Online]. Available at: https://www.sciencedirect.com/science/article/pii/S1877050918308019 [Accessed 15 November 2023].
5. Ferdin Joe John Joseph, Sarayut Nonsiri, Annop Monsakul. Keras and TensorFlow: A Hands-On Experience doi: <https://doi.org/10.1007/978-3-030-66519-7_4>
6. Nuruldelmia Idris, Cik Feresa Mohd Foozy, Palaniappan Shamala. (April 2020). A Generic Review of Web Technology: DJango and Flask. International Journal of Advanced Computing Science and Engineering. Vol. 2(No. 1), pp.34-40. Available at: https://www.academia.edu/41538217/Efficient\_Way\_of\_Web\_Development\_Using\_Python\_and\_Flask [Accessed 18 January 2024].
7. Kaggle: - <https://www.kaggle.com/>