International Journal of Research in Engineering, Science and Management
Volume 8, Issue 10, October 2025
https://www.ijresm.com | ISSN (Online): 2581-5792

Assisting Visually Impaired People to Detect Objects Using Machine Learning

Harditya Shah[1*], Sahil Nannaware[2], Rohit Singh[3], Jyoti Ramteke[4]
[1,2,3]Student, Department of Computer Engineering, Sardar Patel Institute of Technology, Mumbai, India
[4]Professor, Department of Computer Engineering, Sardar Patel Institute of Technology, Mumbai, India

***Abstract*: Navigating the world poses significant challenges for the visually impaired, compounded by the limited availability of suitable technological solutions. Traditional aids often fall short in providing real-time assistance and fail to address the nuanced complexities of everyday tasks. Our project aims to bridge this gap by introducing a comprehensive solution tailored to the unique needs of visually impaired individuals. By capturing live footage of their surroundings, our system offers immediate access to visual information previously inaccessible to the user. Through an advanced image processing module, the captured images are analyzed and interpreted in real-time, enabling the identification of objects, obstacles, and spatial cues. This information is then seamlessly converted into spoken feedback using a text-to-speech module, empowering users with clear auditory guidance to navigate their environment confidently. With a focus on usability and effectiveness, our project represents a significant step forward in enhancing the independence and quality of life for the visually impaired community. By leveraging cutting-edge technology, we aspire to create a more inclusive world where everyone has equal access to information and opportunities.**

***Keywords*: Machine Learning, YOLO, CNN.**

1. Introduction

In today's digital age, software plays a crucial role in enhancing accessibility for individuals with disabilities. Among these, visually challenged people face unique challenges that require innovative technological solutions to ensure they can interact with digital environments effectively. This report explores the various software applications designed specifically to assist visually challenged individuals, examining their features, benefits, and impact on daily living and professional activities. The aim of this report is to provide a comprehensive overview of current software solutions, highlighting advancements in accessibility technology. It delves into screen readers, magnification tools, voice recognition software, and other assistive technologies that empower visually challenged users. Additionally, the report discusses the ongoing development and future potential of these technologies in making digital content more accessible. By understanding the current landscape and identifying areas for improvement, this

Report seeks to contribute to the ongoing conversation about digital inclusivity. It underscores the importance of developing and implementing software that not only meets the needs of visually challenged individuals but also enhances their overall quality of life and independence.

This project focuses on leveraging deep learning techniques to tackle two significant computer vision challenges: object detection and text detection. By integrating state-of-the-art algorithms, the project aims to deliver robust and efficient solutions for real-world applications such as surveillance, document digitization, and autonomous systems.

A. Objective

1) *Object Detection*: To identify and classify objects in images or video streams with high accuracy using the YOLO (You Only Look Once) algorithm.
2) *Text Detection*: To detect and recognize text in images using a Convolutional Neural Network (CNN)-based model.

B. Scope

This project has applications in various domains, including:

1) Real-time monitoring and automation.
2) Text recognition for digitizing documents or assisting visually impaired individuals.
3) Enhancing visual data analysis in smart systems.

2. Literature Review

Artificial intelligence vision for visually impaired was published in 2021 by D. Tamilarasi, Pavitra P, Manish Kumar, Pawan Kalyan, Nawazish Manzar and their Findings were Use of ultrasonic sensors, Object detection (YOLO and SSD), GPS and GMS modules but the device is completely focused on object detection and has no scope in face recognition.

Audio assistance for visually impaired teachers using image processing published in 2023 by Sujit Prasad, M. Athif Ahamad, M. Thangatamilian, I. Charumithra, R. Manjula Devi, S. Jagadeshwaran for Portable smart assistance, Face Detection (audio information about classroom activities and mischiefs of the pupils, HaarCascade classifier. The U=use of OpenCV for detection which limits the dataset for model training.

Mobile application for visually impaired published in 2023 by M.A.M Mahir, Upandra, M.N.M Hussain, C.J. Wickramarathne, Perera R.D.D, Dharshana. They find out that Physical volunteer radar, Smart document reader, Audiobook Rack can be implemented but, in this system, it can't recognize

*Corresponding author: harditya.shah@spit.ac.in

the mathematical equations, handwritten text and special symbols.

In [?], Finger vision for visually impaired published in 2021 by Kasthuri N, Nethra Krupa A, Naveen Kumar S, Madhvan R. They proposed method enables the visually impaired user to read the document on the go and helps them to reduce the dependency on others for reading anything with an efficiency of nearly 87 percent. The gaps were the use of physical devices which requires a person to volunteer and also a lot of errors made by the users.

Guidance System for Visually Impaired People released in 2021 by Kanchan Patil, Avinash Kharat, Pratik Chaudhary, Shrikant Bidgar, Rushikesh Gavhane. They worked on the Emotion Recognition, Face Recognition, Image Captioning, Object Detection Module. The problems were in Navigation through surrounding using ultrasonic sensors and the voice-over assistant module without limiting the English language only.

Blind Assist published in 2021 by Shreya P, Shreyas N, Pushya D, Uma Maheswar Reddy N. They find out multiple functionalities including optical character recognition (OCR), currency recognition, Users can Scan any text (document, textbook, address boards, etc.) and the assistant will be able to read it out clearly, can successfully send e-mail through Gmail Server with the help of voice assistant. The problem was with face recognition for login, traffic board sign reader and E-mail inbox reader through voice assistant.

A Google Glass Based Real-Time Scene Analysis for the Visually Impaired published in 2022 by Hafeez Ali A., Sanjeev U. Rao, Swaroop Ranganath, T. S. Ashwin, Guddeti Ram Mohana Reddy. The system they designed was to be highly portable, easy to wear, and works in real-time, Embedded sensors that can achieve these functionalities with little to no need for external sensors. The issues were with highly dependent on a strong internet connection, the device is relatively expensive in developing countries.

Internet of Things Facilitating Blind People's Daily Lives (IoT) published in 2023 by Pyush Bhanote, Lekha Rani. They worked in Infrared sensors for stairs, ultrasonic sensors to object detection, or water sensors for puddle detection, responds quickly, is lightweight, and is reasonably priced. The gaps were with the un implementation of GPS and GSM system.

Android Integrated Voice-based Intimation Via GPS with Panic Alert System published in 2023 by D. Rajalakshmi, P. Shobha Rani, Talapaneni Lakshmi Bhavana, Viveka Emani, Reddy Swetha. There working were on the technology that allows the vision handicapped to safely and autonomously navigate their environment, other dangers may be detected by water and fire sensors on board, SOS Button helps in case of emergency. They could also work for better object detection is possible with the use of computer vision.

3. Methodology

The project employed a dual approach to address object and text detection tasks. For object detection, YOLO (You Only Look Once) was utilized due to its ability to process images in real-time and deliver accurate predictions. The model was trained on a diverse dataset, leveraging transfer learning to improve detection efficiency and accuracy for various object categories.

For text detection, a Convolutional Neural Network (CNN) architecture was used, designed to identify and extract text features from images. The model was trained using a labeled dataset containing varied text styles, sizes, and orientations. Preprocessing steps, including resizing and normalization, were applied to ensure consistency across the data. Hyperparameters such as learning rate and batch size were optimized to achieve the best performance.

Both models were evaluated using standard metrics such as accuracy, precision, and recall to assess their effectiveness, with the results analyzed to identify strengths and areas for improvement.

A. Data Loading

1) *Object Detection*: The COCO dataset and benchmark are used in a wide range of AI vision tasks and disciplines. Models trained on COCO are used for object detection, people detection, face detection, pose estimation, and many more computer vision tasks. The COCO (Common Objects in Context) dataset is a large-scale image recognition dataset for object detection, segmentation, and captioning tasks. It contains over 330,000 images, each annotated with 80 object categories and 5 captions describing the scene.
2) *For Text Detection*: The MNIST dataset, or Modified National Institute of Standards and Technology dataset, is a collection of handwritten digits that's used to train and test image. The main dataset is the letters EMINST subset, which consists of 103600 characters with 26 classes. Each image is 28x28 pixels, grayscale. Each image is a crude 28 x 28 (784 pixels) handwritten digit from "0" to "9" and "A" to "Z". Each pixel value is a grayscale integer between 0 and 255.

B. Algorithms

Non-Maximum Suppression (NMS) is a crucial algorithm in object detection tasks to refine the predictions made by a model, such as YOLO. Its primary goal is to eliminate redundant or overlapping bounding boxes, ensuring that only the most relevant detections are retained for each object.

Fig. 1. Non-Max suppression algorithm

Steps in Non-Maximum Suppression:

1) The model predicts multiple bounding boxes for objects, each associated with a confidence score indicating the likelihood of the box containing the object. Sort by Confidence Score:
2) All bounding boxes are sorted in descending order based on their confidence scores. Select the Box with the Highest Score:
3) The box with the highest confidence score is selected as the most probable detection. Calculate Intersection Over Union (IoU):
4) For all remaining boxes, compute the IoU between the selected box and each of the other boxes. IoU measures the overlap between two bounding boxes relative to their combined area. Suppress Overlapping Boxes:
5) Any box with an IoU above a predefined threshold (e.g., 0.5) is suppressed (removed) as it likely represents the same object. Repeat:
6) The process is repeated with the remaining boxes until no further boxes are left.

4. Model Architecture

A. For Object detection

Model: The model here is the You Only Look Once (YOLO) algorithm that runs through a variation of an extremely com plex Convolutional Neural Network architecture called the Darknet. We are using a more enhanced and complex YOLO v3 model. Also, the python cv2 package has a method to setup Darknet from our configurations in the yolov3.cfg file. COCO has already been trained on YOLO v3 by others, so we will be using a pre-trained model and we have already obtained the weights stored in a 200+mb file. Input Data: We will be using live web camera detection and feed them to this trained model. API: The class prediction of the objects detected in every image will be a string e.g. “cat”. We will also obtain the coordinates of the objects in the image and append the position “top”/“mid”/“bottom” and “left”/“center”/“right” to the class prediction “cat”. We can then send the text description to the Google Text-to-Speech API using the gTTS package. Output: We will be getting voice feedbacks in the form e.g. “bottom left cat” — meaning a cat was detected on the bottom-left of my camera view using Google Text-to-Speech API using gTTS package by giving text description of the object.

B. For Text Detection

1) *Input Layer*: This is the first layer in the neural network, which defines the shape of the input data. Shape = (28, 28, 1): The input shape represents the dimensions of the input data:
   - 28, 28: The image size is 28x28 pixels (common for grayscale image datasets like MNIST)
   - 1: The number of channels (for grayscale images, there is only 1 channel, whereas for RGB images, there would be 3 channels).
2) *Convolutional Layers (Conv2D)*: These layers are used to detect features in images, such as edges, textures, or shapes. They apply a set of filters to the input data to extract useful patterns. 32, 64, 128 Units (Filters): These refer to the number of filters in each convolutional layer. More filters mean the network can learn more complex features. For example: The first layer has 32 filters (32 different feature detectors). The second layer has 64 filters. The third layer has 128 filters. Kernel Size = 3: This refers to the 14 size of the filter (3x3) that slides over the image to detect patterns. The kernel size is small enough to capture fine details but large enough to capture meaningful spatial relationships. Activation = ReLU (Rectified Linear Unit): The ReLU activation function is applied after the convolution operation to introduce non-linearity. It sets negative values to 0 and keeps positive values unchanged. This helps the network model complex patterns and ensures that the layers learn non-linear relationships.
3) *MaxPooling Layer*: This layer is used to reduce the spatial dimensions (height and width) of the input while retaining the most important features. It helps reduce the computa- tional cost and prevent overfitting. Max Pooling Operation: It takes the maximum value from a small region (typically 2x2) of the input, effectively downsampling the image. For example, after pooling, the feature map’s size is reduced, but it retains the most prominent features (like edges).
4) *Flatten Layer*: This layer converts the 2D matrix (output from the convolution and pooling layers) into a 1D vector. This step is necessary before passing the data to fully connected (dense) layers, as they require a 1D input. For example, if the output from the previous layer is of shape (7, 7, 128), the Flatten layer will convert this into a single vector of size 7*7*128 = 6272.
5) *Dense Layer*: A fully connected (dense) layer connects every neuron from the previous layer to every neuron in the current layer. It’s typically used after the convolu tional and pooling layers for classification or regression. First Dense Layer with 512 Units and ReLU Activation: This layer has 512 neurons (or units). Each neuron receives input from all 6272 neurons (or however many there are from the previous Flatten layer) and performs a weighted sum. The ReLU activation is applied to introduce non-linearity. Second Dense Layer with 128 Units and ReLU Activation: This layer has 128 neurons and also uses the ReLU activation function. It reduces the number of neurons to help the model learn more abstract features. Third Dense Layer (Output Layer) with Softmax Activation: This is the final layer, and the number of units is equal to the number of letters (i.e., the number of classes in your classification task). Softmax Activation: Softmax is used in the output layer of a classification network when the task involves multi

class classification. It converts the raw scores (logits) into probabilities, and the sum of all probabilities across the output units will equal 1. The class with the highest probability is the predicted class.

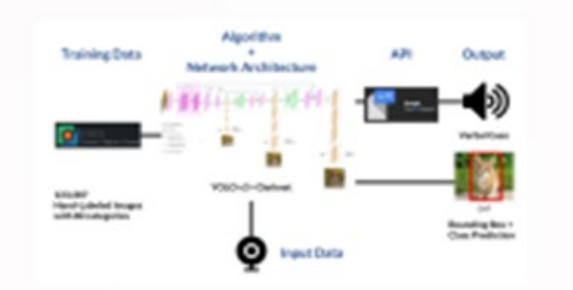

Fig. 2. Design

5. Design

Fig. 3. Model results for classifiers

1) *Image Acquisition*: It is the collection of images for conversion into printed format from other sources. It retrieves the input text or detects the image to start the initial process.
2) *Pre-Processing*: Preliminary processing of data enhances the quality of an image. Each binary image contains a threshold value and they can be set as a local and global value. The technique of preparing (cleaning and organizing) the raw data to make it suitable for building the training models to a readable format.
3) *Segmentation*: The read image is converted into normalized image to get accurate data. This process can be processed as explicitly and implicitly both in the classification phase. The characters are segmented on the basis of pre-processing and RGB images.
4) *Feature Extraction*: After the segmentation the characters are separated based on their features which are extracted from high quality images and are observed with the help of inter-class variations and those characters are selected which are efficiently computable.
5) *Classification*: This step helps the segmented characters to arrange them into different categories and classes. After evaluating their result, they will divide into two categories such as structural pattern classification and statistical pattern classification.
6) *Post Processing*: The final step is to provide a better quality image to the system and to provide accurate results. Contextual and lexical processing is done to reduce the chances of errors. Differentiates the training phase and testing phase which are the essential requirements to develop an OCR. At last, the characters are recognized and converted into speech.

6. Result and Analysis

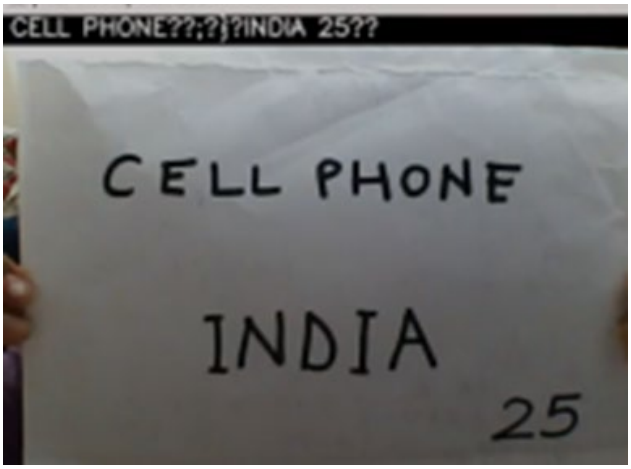

Fig. 4. Text detection

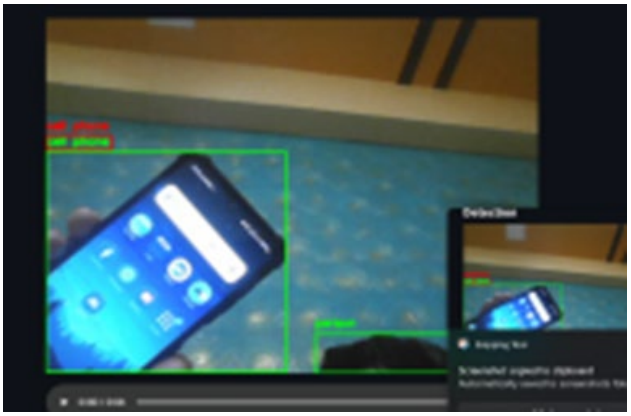

Fig. 5. Object detection

The object detection model developed using YOLO achieved an overall accuracy of 82 percent, showcasing its ability to detect and classify objects effectively in diverse scenarios. The model performed particularly well with medium to-large objects in clear environments but faced challenges with smaller or heavily occluded objects, which impacted its overall accuracy. This highlights potential areas for improvement, such as fine-tuning the model or increasing the dataset diversity.

For text detection, the CNN-based model demonstrated impressive results, achieving a training accuracy of 98 percent and a validation accuracy of 92 percent. The high training accuracy reflects the model's capacity to learn complex text features, while the slight reduction in validation accuracy suggests effective generalization with minimal overfitting. The CNN model excelled in detecting text under various conditions, such as differing font styles and sizes, but occasional inaccuracies were noted in low-contrast or highly distorted text regions.

Overall, the combination of YOLO for object detection and CNN for text recognition provides a complementary approach for robust visual analysis tasks. While both models show strong potential, further optimization, such as augmentation of training data or refining hyperparameters, could enhance their performance and adaptability.

7. Conclusion

In conclusion, the development and implementation of software tailored for visually challenged individuals is not just a technological necessity but a societal imperative. This report has highlighted the critical role that accessible software plays in empowering visually challenged users, enabling them to engage more fully with digital environments. Through the examination of various assistive technologies such as screen readers, magnification tools, and voice recognition software, it is evident that significant strides have been made in enhancing digital accessibility. However, the journey towards complete digital inclusivity is ongoing. Continuous innovation, com bined with a dedicated focus on user-centric design, is essential to address the remaining barriers that visually challenged individuals face. The insights and recommendations provided in this report underscore the importance of collaborative efforts among developers, designers, policymakers, and advocacy groups to create a more inclusive digital landscape, we can ensure that visually challenged individuals have equal opportunities to leverage the benefits of technology, ultimately enhancing their quality of life and fostering a more inclusive society.

8. References

[1] Mahesh, Ananth and Dheepthi, "Using Machine Learning to Detect and Classify URLs: A Phishing Detection Approach," 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2023, pp. 1285-1291.

[2] Kanchan Patil, Avinash Kharat, Pratik Chaudhary, Shrikant Bidgar, Rushikesh Gavhane, "Guidance System for Visually Impaired People," 2021.

[3] Shreya P, Shreyas N, Pushya D, Uma Maheswar Reddy N., "Blind Assist," 2021.

[4] Kasthuri N, Nethra Krupa A, Naveen Kumar S, Madhvan., "Finger vision for visually impaired," 2021.

[5] M.A.M Mahir, Upandra, M.N.M Hussain, C.J. Wickramarathne, Perera R.D.D, Dharshana, "Mobile application for visually impaired," 2023.

[6] Sujit Prasad, M. Athif Ahamad, M. Thangatamilian, I. Charumithra, R. Manjula Devi, S. Jagadeshwaran, "Audio assistance for visually impaired teachers using image processing," 2023.

[7] D. Tamilarasi, Pavitra P, Manish Kumar, Pawan Kalyan, Nawazish Manzar, "Artificial intelligence vision for visually impaired," 2021.

[8] Eko Didik Widianto, Peiqi Yang, Anggilia Nur Safitri, Khairun Nisa Maulani, Arseto Satriyo Nugroho, "A Mobile Application CODEC (Color Detection) for Color-blind People using KNN," 2023.

[9] Jiayi Wu, Peiqi Yang, Yicong Chai, Kexin Zhang, Siyu Liang, Jingwen Mo, Cheng Liu, "Smart Blind Guide Device Based on Raspberry Pi," 2023.

[10] D. Rajalakshmi, P. Shobha Rani, Talapaneni Lakshmi Bhavana, Viveka Emani, Reddy Swetha, "Android Integrated Voice-based Intimation Via GPS with Panic Alert System," 2023.