VISION-AID

Empowering sightless vision through advanced technology.

Our Team

Abdul Kader Mohamed Moosa

Chetty B Indira

Dharanidharan Ekambaram

Senthil Kumar Aswin

Omarkathaf Asira

Duraipandi Hari Vignesh

Kesavan Dinesh

Anusha Gundlapalli

INTRODUCTION

VisionAid is an innovative assistive tool designed to enhance the lives of visually impaired individuals. By combining advanced technologies like Paddle OCR for text recognition and YOLOv5 for object detection, VisionAid processes captured or uploaded images and provides real-time audio feedback using Google Text-to-Speech (gTTS). This seamless integration enables users to interact with their surroundings, access information, and navigate environments with greater independence and confidence. VisionAid is a step toward a more inclusive and accessible world.

PROBLEM STATEMENT

Millions of visually impaired individuals encounter significant challenges in their daily lives, such as reading text, identifying objects, and navigating unfamiliar environments. These barriers limit their independence, confidence, and ability to access crucial information. Despite advancements in technology, there remains a lack of affordable, real-time solutions tailored to address these needs. This highlights the urgent necessity for an innovative assistive tool that empowers visually impaired individuals by leveraging cutting-edge technologies to enhance accessibility and promote inclusion in society.

OBJECTIVES

- To develop an innovative assistive tool for visually impaired individuals that provides real-time accessibility.
- To enable text recognition using advanced OCR technology (Paddle OCR).
- To identify objects in the environment using deep learning-based object detection (YOLOv5).
- To convert extracted information into audio feedback using text-to-speech (gTTS).
- To enhance independence and promote inclusivity for visually impaired users by bridging the gap between them and their surroundings.

DATASET DESCRIPTION

Vision Aid Dataset Description

- Uses COCO-Text v2.0 (for scene text recognition) and COCO (for object detection).
- Offers high-quality annotations, diverse real-world images, and complex scenarios for deep learning training.

COCO-Text v2.0: Scene Text Recognition

- Large dataset: 63,686 images with 239,506 annotated text regions.
- Comprehensive annotations: Includes machine-printed & handwritten text.
- Real-world diversity: Captures natural scene text in various environments.
- Metadata: Provides legibility, language, occlusions, curved text, and distortions for robust OCR training.

Dataset Structure

- Image-Level: Image ID, dimensions, and filename.
- Text Annotations: Bounding boxes, legibility, text type, and language.
- Additional Flags: Occlusions, distortions, vertical text, and artificial/natural content sources.

Vision Aid

Input Layer
Captures images or videos from a camera or allows uploading.

Image Enhancement
Adjusts brightness and contrast to improve image quality.

Noise Reduction
Applies filters to remove unwanted noise and enhance clarity.

Edge Detection
Identifies edges to distinguish important features.

Resizing & Normalization
Resizes and normalizes images for model compatibility.

Final Output
Optimized image ready for OCR or object detection.

PREPROCESSING MODULE

1 Image Acquisition: Captures images/videos via camera or allows users to upload images.

2 Image Enhancement: Adjusts brightness, contrast, and sharpness for better clarity.

3 Noise Reduction: Applies filters to remove unwanted artifacts and improve text visibility.

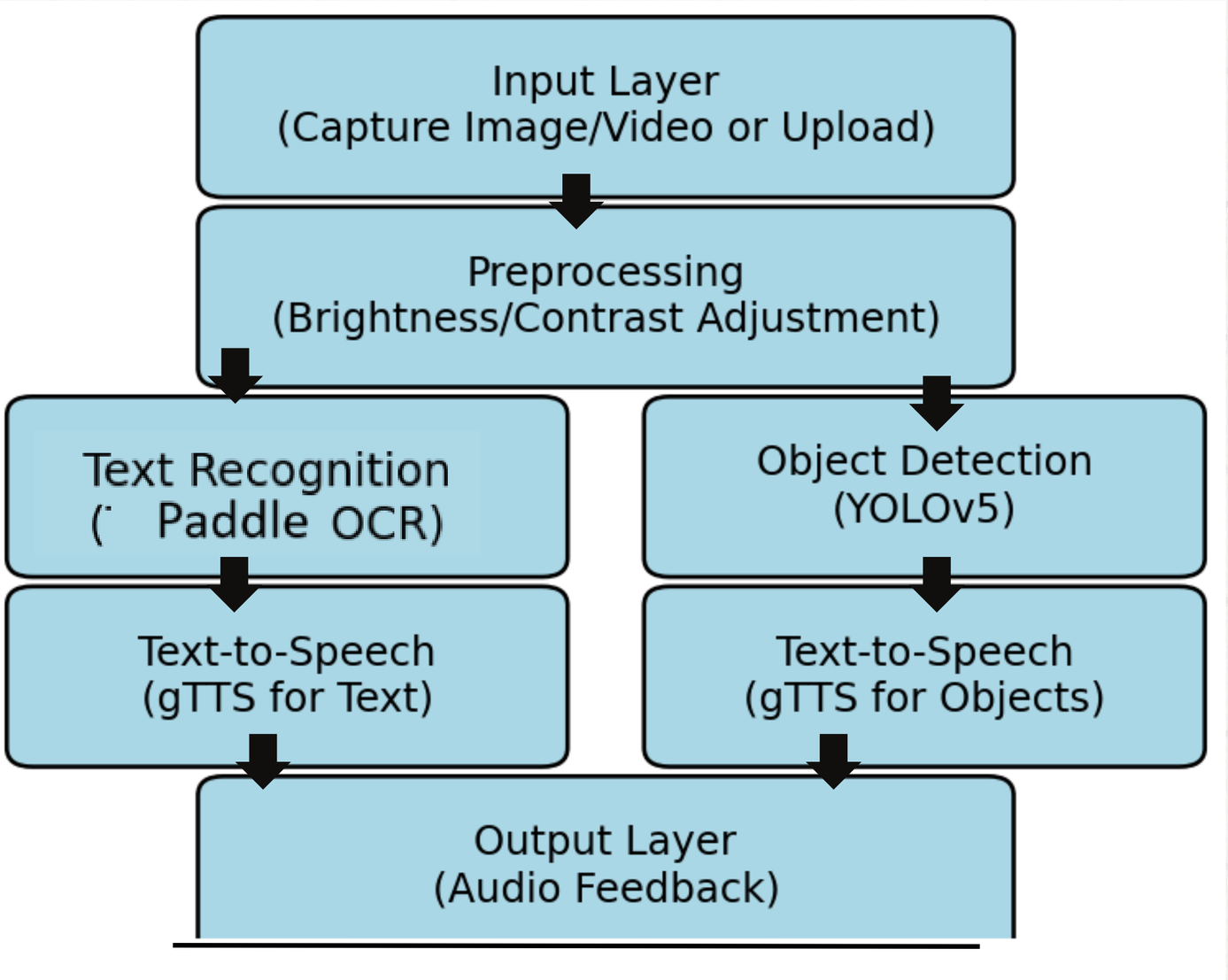
4 Grayscale & Binarization: Converts images to grayscale and enhances text regions for OCR.

5 Edge Detection & Segmentation: Identifies text areas and object boundaries for recognition.

6 Image Resizing & Normalization: Scales images while maintaining aspect ratio for deep learning models.

7 Final Preprocessed Output: Generates optimized images ready for OCR and object detection.

IDEATION PROCESS



The flowchart outlines the workflow of an assistive tool for visually impaired users. Input is captured via a camera or uploaded by the user and then preprocessed to enhance quality. Text recognition is performed using Paddle OCR to extract text, while object detection uses YOLOv5 to identify objects in the scene. Both outputs are converted to audio descriptions using gTTS (Google Text-to-Speech). Finally, the system provides real-time audio feedback, enabling users to interact with their environment effectively.

Your paragraph text

MODULE DESCRIPTION

PREPROCESSING MODULE

- Captures images or videos through a camera or allows users to upload an image.
- Enhances the quality of the input by adjusting brightness and contrast.
- Prepares the data for accurate text recognition and object detection.

TEXT RECOGNITION MODULE

- Utilizes Paddle OCR to extract text from the preprocessed image.
- Processes the text for further audio conversion.

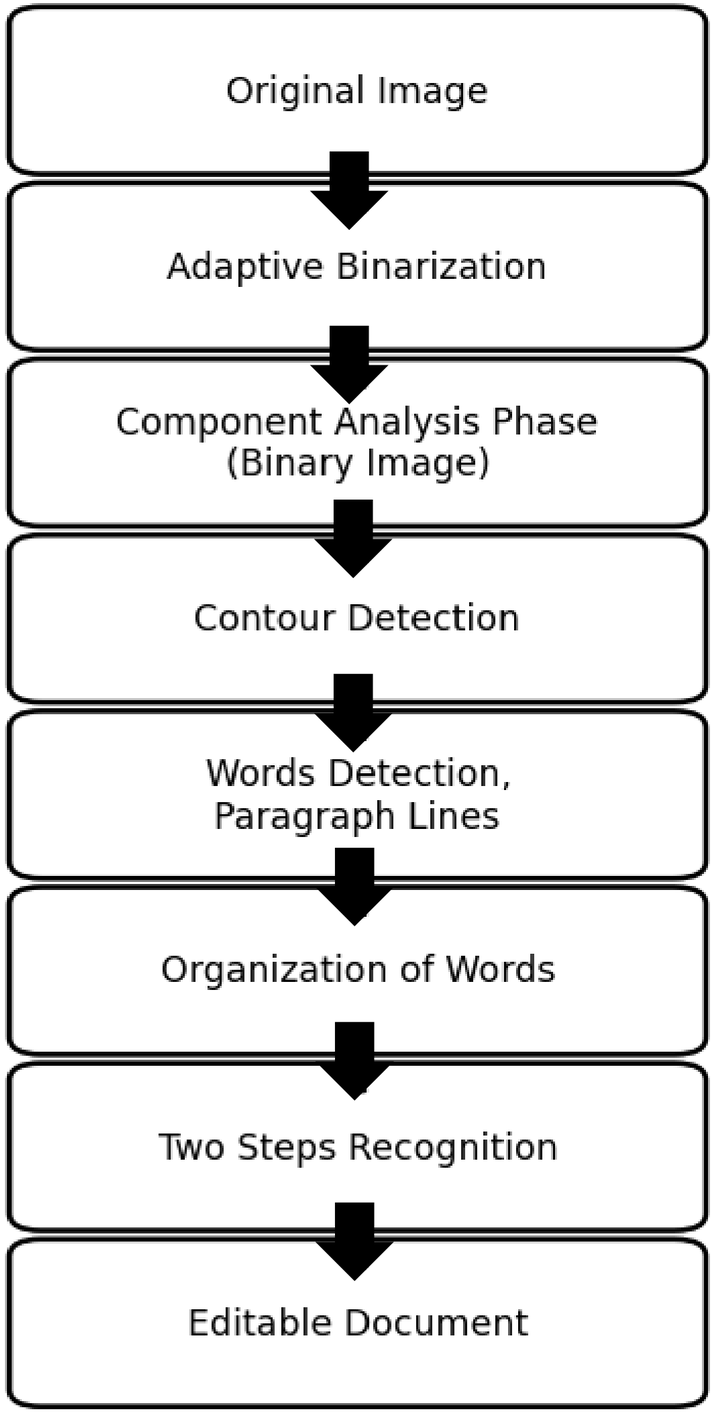
OBJECT DETECTION MODULE

- Leverages YOLOv5, a deep learning model, to identify objects in the image.
- Outputs object labels and their locations for audio generation.

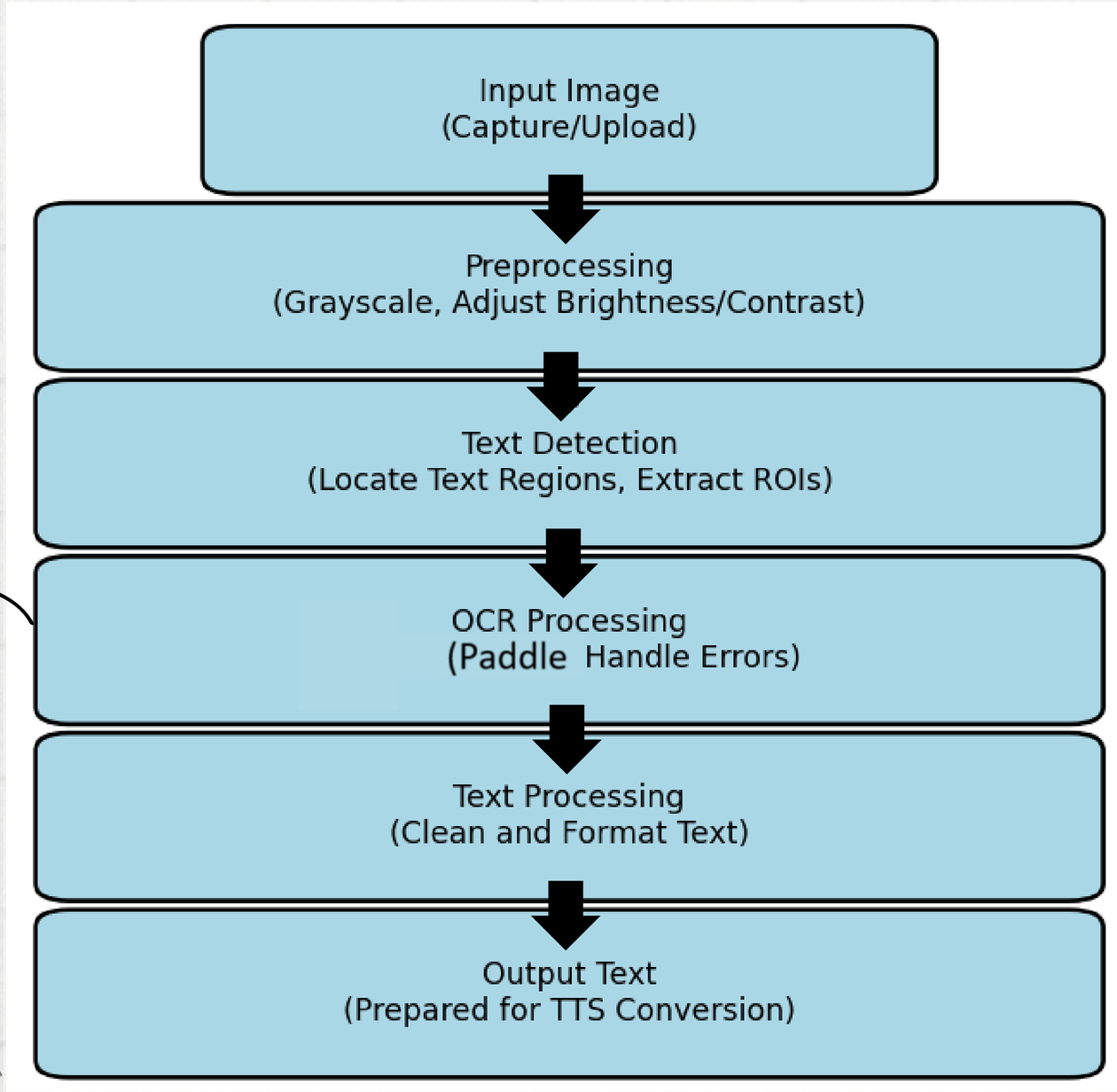
TEXT-TO-SPEECH MODULE

- Converts extracted text or identified objects into audio feedback using Google Text-to-Speech (gTTS).
- Delivers clear and real-time audio output to ensure effective communication with the user.

TEXT RECOGNITION MODULE



OCR



TEXT RECOGNITION MODULE

The Text Recognition Module is designed to process images containing text and convert them into readable, editable text.

1. The process begins with capturing or uploading an image, which is then preprocessed by converting it to grayscale and adjusting brightness and contrast to enhance text visibility.
2. Next, the system detects text regions by extracting regions of interest (ROIs) further processing. Optical Character Recognition (OCR) is then performed using Paddle OCR, where the system recognizes and converts the detected text while handling errors to improve accuracy.
3. After OCR processing, the extracted text undergoes cleaning and formatting to remove unwanted characters and fix formatting inconsistencies.
4. Finally, the processed text is prepared for various applications, including Text-to-Speech (TTS) conversion, allowing visually impaired users to access information audibly. This module ensures an efficient and adaptive pipeline that enhances text clarity, minimizes errors, and supports diverse text styles and backgrounds, making it useful for applications like document editing, voice output, and digital archiving.

TEXT RECOGNITION MODULE

1. PaddleOCR is an open-source OCR tool developed as part of the PaddlePaddle deep learning framework by Baidu. It efficiently recognizes and extracts text from images, making it ideal for tasks like document digitization, text recognition in complex backgrounds, and multilingual text processing (supports 80+ languages).

Text Detection:

- The first step is finding areas in an image that contain text. PaddleOCR uses models like DB (Differentiable Binarization) or EAST (Efficient and Accurate Scene Text Detector) to detect and locate these text regions.
- Text Recognition:
- Once text regions are identified, PaddleOCR converts the text into machine-readable form using CRNN (Convolutional Recurrent Neural Network) or sequence-based models. These models process characters and words, even in challenging formats like handwritten text or curved text.
- Post-Processing:
- After recognition, the extracted text is cleaned and formatted to fix errors or inconsistencies, ensuring the output is clear and accurate.

OBJECT RECOGNITION MODULE

Start: Input Image

Preprocessing
(Resize, Normalize)

Feature Extraction
(Convolutional Layers)

Bounding Box Prediction
(Object Localization)

Classification
(Label Assignment)

Non-Max Suppression
(Remove Redundant Detections)

Final Object Detection
(Labels & Bounding Boxes)

Output: Recognized Objects

YOLOv5

Start: Input Image
(Capture/Upload)

Preprocessing
(Resize, Normalize, Enhance)

Feature Extraction
(CNN Layers, YOLO Backbone)

Object Detection
(YOLOv5 Model, Bounding Box Prediction)

Confidence Filtering
(Thresholding & Non-Max Suppression)

Object Classification
(Label Assignment)

Post-Processing
(Refinement & Error Handling)

Final Output
(Detected Objects & Labels)

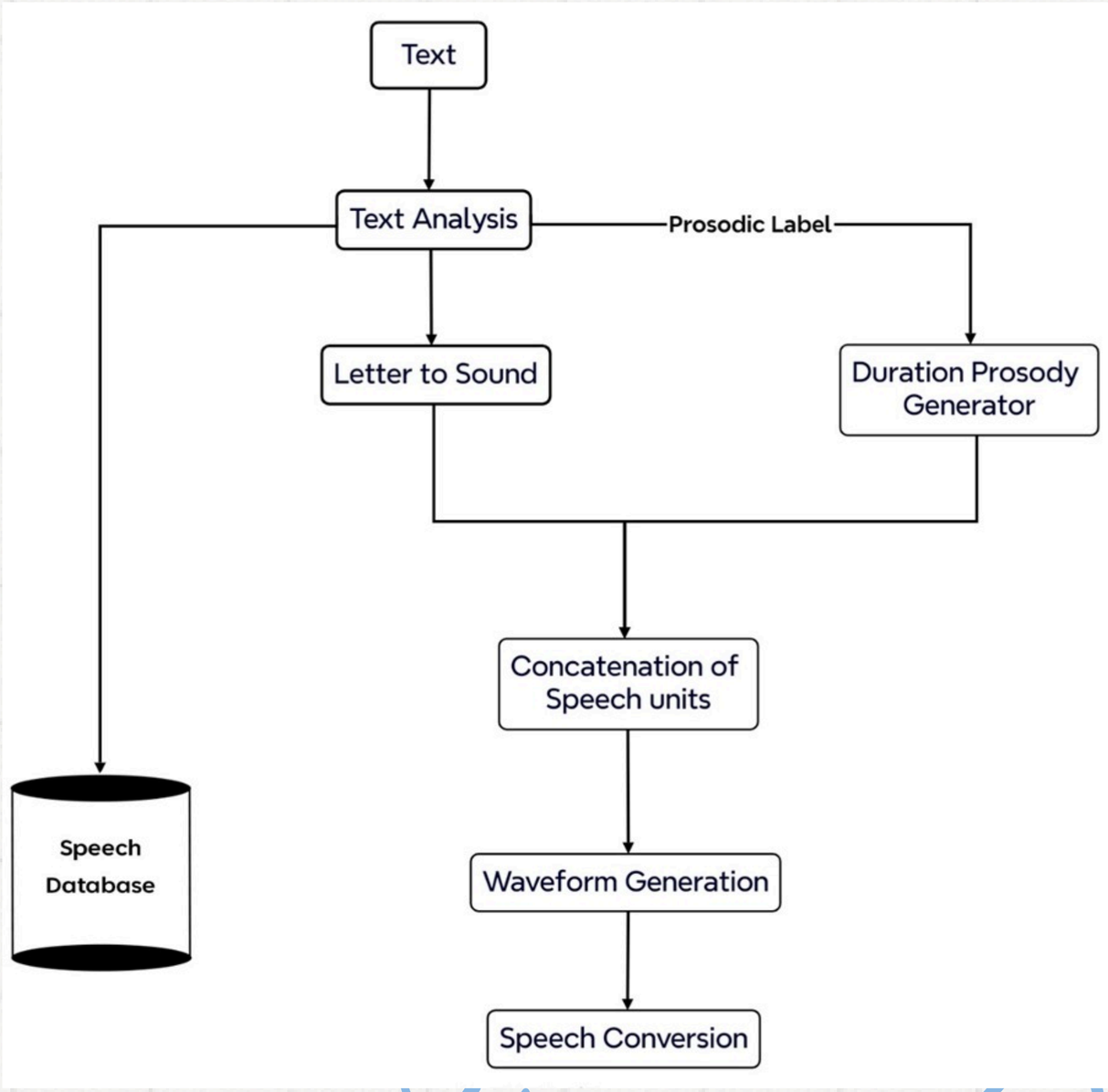
OBJECT RECOGNITION MODULE

The Object Detection Module is designed to identify and classify objects within an image, enabling users to better understand their surroundings.

1. Image Acquisition & Preprocessing – The system captures or uploads an image, which is then resized, normalized, and enhanced to improve detection accuracy.
2. Object Detection & Localization – Using YOLOv5, the model detects objects by predicting bounding boxes and assigning confidence scores to ensure precise identification.
3. Filtering & Classification – Non-Max Suppression (NMS) removes duplicate detections, and classification algorithms assign appropriate labels to detected objects.
4. Post-Processing & Output – The detected objects are refined, errors are handled, and the final output is prepared for applications like Text-to-Speech (TTS), navigation assistance, and real-time alerts.

GOOGLE TTS

Real time speech and text generation using TTS engine: Creating a real-time system for speech and text generation, along with text copying using a Text-to-Speech (TTS) engine, for sign language training and translation.text



HYPERPARAMETER TUNING

Key Parameters Tuned:
det_db_box_thresh, det_db_unclip_ratio, use_angle_cls, rec_algorithm, and drop_score.

Tested on Four Text Types:
Handwritten, CAPTCHA, Printed, and Digital Text.

Hyperparameter Tuning & Results
Each parameter was tested and optimized for best performance.

Final Tuned Values & Adjustments:
det_db_box_thresh (0.25) → Improved recall for handwritten & CAPTCHA text.
det_db_unclip_ratio (1.9) → Prevented text cropping.
use_angle_cls (True) → Corrected rotated text for better readability.
rec_algorithm (SVTR_LCNet) → Transformer-based model improved accuracy.
use_direction_classify (True) → Helped with character alignment.

HYPERPARAMETER TUNING

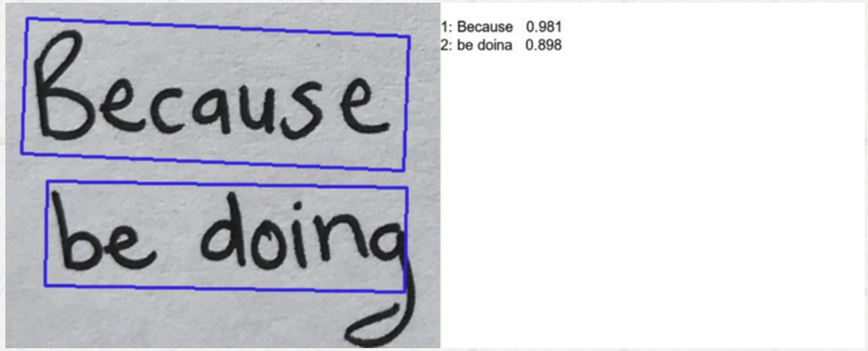
Key Observations:

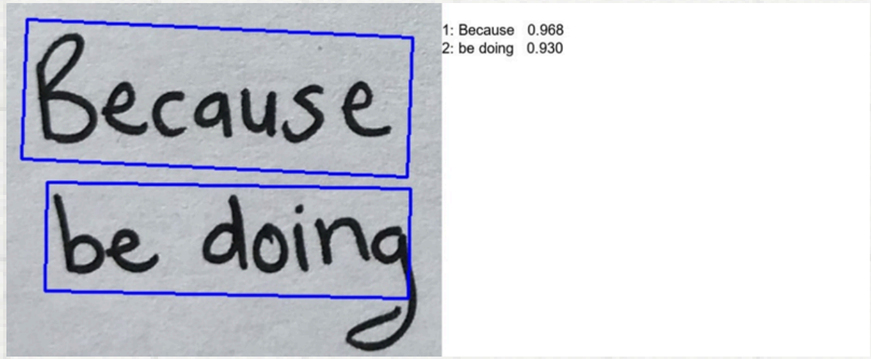
Bounding Box Detection

Improved precision and alignment after tuning.
Better coverage around text contours.

Text Recognition Accuracy

Before Tuning: Misrecognized words (e.g., "be doina" with 0.898 confidence).
After Tuning: Corrected to "be doing" (0.930 confidence).





Effect of Hyperparameter Tuning
Lower det_db_box_thresh (0.25) → Prevented missing text.
use_angle_cls (True) → Fixed minor rotations in text.
use_direction_classify (True) → Enhanced word structure.
Overall Accuracy Improved → Stronger recognition of handwritten text while maintaining reliability

MODEL EVALUATION

Evaluation Metrics

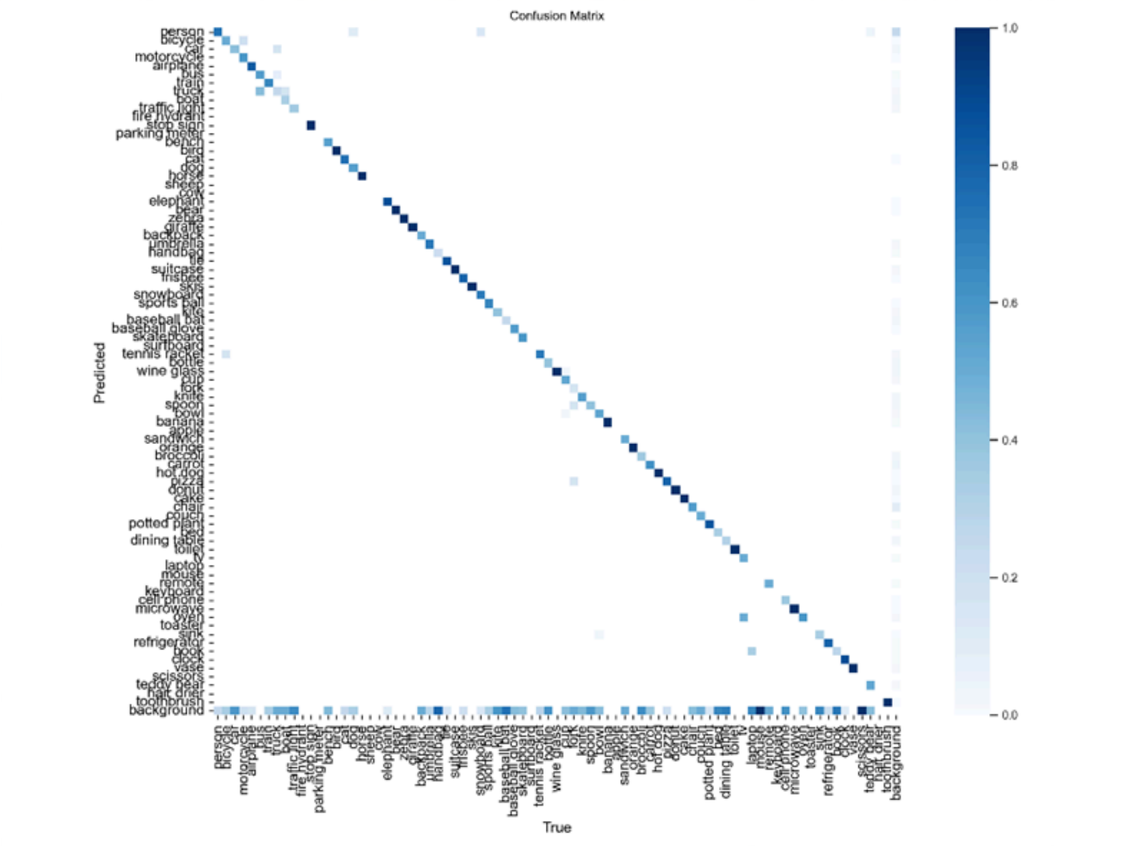
- ***Precision (P):*** Measures the proportion of correctly predicted objects among all detections.
- ***Recall (R):*** Indicates the model's ability to detect all relevant objects in an image.
- ***Mean Average Precision (mAP):***
  mAP@0.5: Measures the precision-recall area under the curve at IoU = 0.5.
  mAP@0.5:0.95: Evaluates model performance across IoU thresholds (0.5 to 0.95).
- ***F1-Score:*** Balances precision and recall.

Quantitative Results

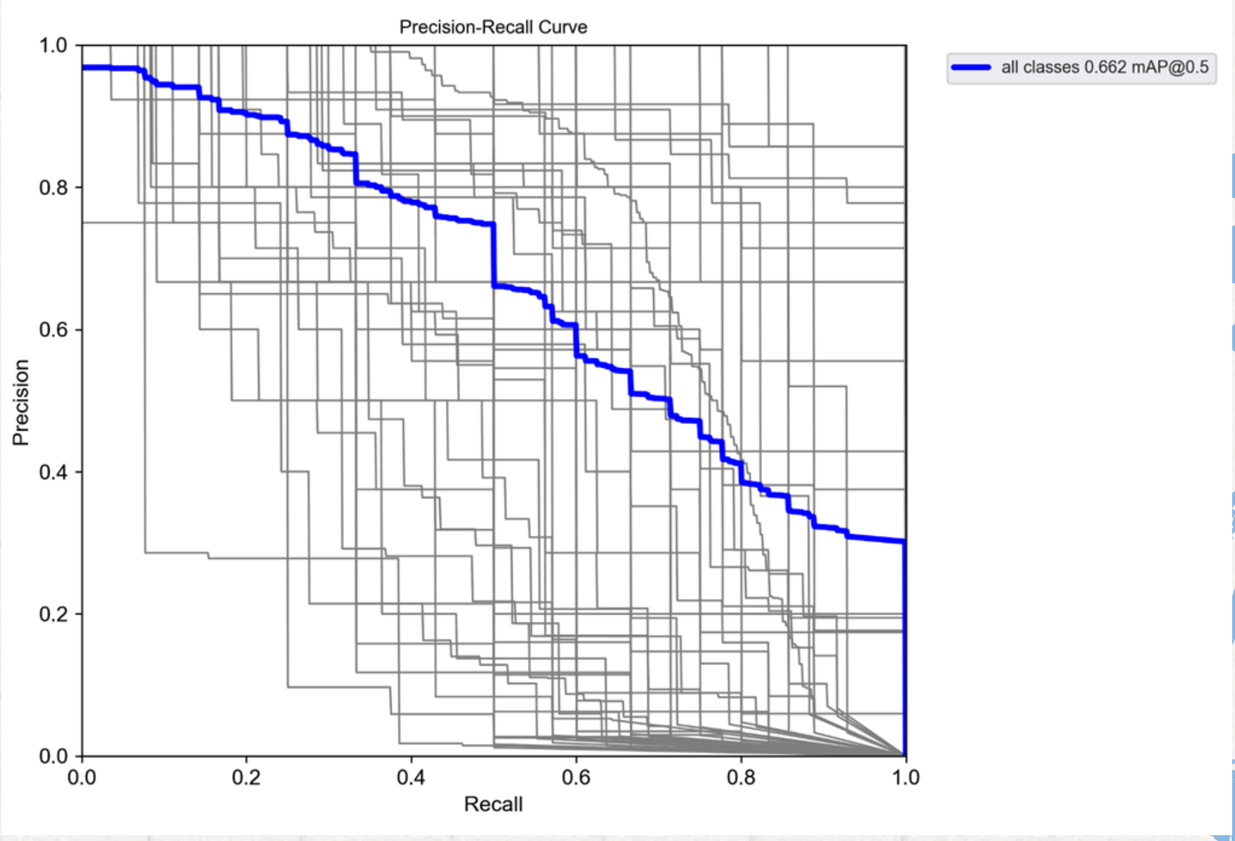
Metric	*Value*
Precision (P)	0.71
Recall (R)	0.80
mAP@0.5	0.66
mAP@0.5:0.95	0.62
F1-Score	0.63

MODEL EVALUATION

Confusion Matrix



Precision-Recall and F1-Confidence Analysis



CONCLUSION

- The developed system successfully integrates advanced technologies like Paddle OCR and YOLOv5 to provide a comprehensive assistive tool for visually impaired individuals.
- By enabling real-time text recognition, object detection, and audio feedback, the tool bridges accessibility gaps, empowering users to interact more confidently with their environment.
- VisionAid demonstrates the potential of AI to foster inclusivity and independence, paving the way for future enhancements to make daily life more accessible for all.

FUTURE ENHANCEMENTS

- **Multilingual Support:** Incorporate OCR and text-to-speech capabilities for multiple languages to cater to a diverse user base globally.
- **Improved Object Detection:** Upgrade to more advanced object detection models for enhanced accuracy and recognition of smaller or complex objects.
- **Mobile Application:** Develop a mobile-friendly version of the tool for easy access and portability.
- **Cloud Integration:** Enable cloud-based processing for faster and more efficient analysis of large datasets.
- **Voice Commands:** Introduce voice-based interactions to enhance user experience and control.

Thank you