|  |
| --- |
| Title: Medicine Label Reader Using OCR (PaddleOCR) |
| Done by : Pritesh Ram Keshri Project Number : 12, Submission Date: 05th July 2025 |
| **Overview:**  This project implements an Optical Character Recognition (OCR) system that reads and extracts textual information from medicine labels and pill containers. The system uses **PaddleOCR**, a high-performance OCR library, to detect and recognize text in images. It aims to assist users (e.g., patients, pharmacists, or healthcare providers) in easily understanding printed medicine information such as names, dosages, and usage instructions. |
| **Libraries used:**   * **paddleocr:** High-accuracy OCR engine developed by PaddlePaddle with multilingual support and built-in direction classifiers. * **matplotlib:** Used for visualizing images and OCR results. * **cv2 (OpenCV):** Used for reading and displaying images, and basic image manipulation. * **os:** For file management and accessing image directories. |
| **Dataset Details:**  No predefined dataset was used. The application is designed to read medicine labels from user-provided images. For testing purposes, sample label images were used to verify the OCR system's effectiveness. |
| **APIs Integrated:**  No external APIs were used. Instead, the project uses **PaddleOCR**, which is installed and run locally. It internally leverages deep learning models pre-trained on large datasets for text detection and recognition. |
| **Source code 1: File Name : OCR Initialization and Text Detection**  from paddleocr import PaddleOCR  import cv2  import matplotlib.pyplot as plt  ocr = PaddleOCR(use\_angle\_cls=True, lang='en') # Enable angle classifier and English language  img\_path = "sample\_label.jpg"  image = cv2.imread(img\_path)  result = ocr.ocr(img\_path, cls=True) |
| **Source code 2: Visualization of Detected Text**  for idx in range(len(result)):  res = result[idx]  for line in res:  box, (text, confidence) = line  print(f"Text: {text}, Confidence: {confidence:.2f}") |
| **Source code 3: Display Results on Image**  for line in result[0]:  points = line[0]  text, confidence = line[1]  pts = [(int(pt[0]), int(pt[1])) for pt in points]  cv2.polylines(image, [np.array(pts)], isClosed=True, color=(0,255,0), thickness=2)  cv2.putText(image, text, pts[0], cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (255,0,0), 2)  plt.figure(figsize=(10,8))  plt.imshow(cv2.cvtColor(image, cv2.COLOR\_BGR2RGB))  plt.title("Detected Text on Medicine Label")  plt.axis('off')  plt.show() |
|  |
|  |
| **Output screenshots:**   1. **Input Image:** Raw medicine label uploaded for OCR processing.      1. **Detected Text:** Console output showing recognized text and confidence levels.     **3. Bounding Boxes:** Image with yellow boxes highlighting detected words/phrases. |
| **What you learned:**  Through this project, I gained practical knowledge of:   * Using **PaddleOCR**, a modern OCR framework with multilingual support. * Performing image preprocessing and result visualization using **OpenCV** and **Matplotlib**. * Understanding how to integrate deep learning-based OCR engines in real-world applications. * Interpreting and parsing OCR output for further usage in real-time systems. |
| **What the Skills you gained:**   * **OCR Implementation** using pre-trained models (PaddleOCR). * **Image processing** and manipulation with OpenCV. * **Python programming** for computer vision. * **Visualization of OCR results** for validation and debugging. * **Confidence-based text filtering** to improve result reliability. |
| **Real Time Applications:**   * **Medical Accessibility:** Helps elderly or visually impaired users to read medicine labels easily. * **Pharmaceutical Automation:** Automated systems to verify label information on medicine packaging. * **Telemedicine:** Integration with health apps to extract medicine info during consultations. * **Inventory Management:** Automatic logging of medicine names and batch numbers in hospitals. * **Drug Identification:** Quick scanning of unknown medicine labels for safety checks. |
| **Further Enhancement Suggestions:**   | **Area** | **Improvement** | | --- | --- | | Accuracy | Preprocess images using denoising and contrast  Enhancement. | | Multilingual OCR | Include Hindi, Tamil, or other regional languages. | | Mobile Support | Build an Android/iOS app using Flutter + Paddle  Lite. | | Real-time | Add voice output for visually impaired users. | | UI | Create a user-friendly dashboard for medicine info  Logging. |  |  |  | | --- | --- | |  |  | |  |  | |  |  | |  |  |  |  |  | | --- | --- | |  |  | |  |  | |  |  | |  |  | |