Artificial Intelligence (BCSE306L) Digital Assignment-2

Team Members:
Prabhu Pachisia(23BCE1305)
Shridhan Suman(23BCE1279)

Team Name: Nexus

Image Recognition and Information Extraction

1. Project Introduction

Our project is designed to analyze images and extract useful information in an automated way.
It combines the power of **computer vision** and **natural language processing** to make sense of both the visual and textual content in an image.

It performs three main tasks:

- **OCR (Optical Character Recognition):** Extracts text from images, making it editable and searchable.
- **NLP (Natural Language Processing):** Identifies key entities such as dates, names, organizations, and numbers from the extracted text.
- **Object Detection:** Detects and labels objects present in the image using the YOLOv8 model.

By integrating these technologies, the project can turn raw images into structured, meaningful insights.

2. Objective and Motivation

- **Objective**: To develop a tool that can automatically read, understand, and describe the content of images.
- **Motivation**:
    - Images often contain both text and objects.
    - Manual extraction is time-consuming and error-prone.
    - Our system helps automate this process, making it useful for document analysis, research, and accessibility.

3. Technical Details & Implementation

Technologies Used

- **Python (Core language)**
  Python is the main programming language which we have used to develop our project. It is simple, powerful, and has a large collection of libraries for AI, ML, and image processing.
- **Flask (Web framework for UI & backend)**
  Flask provides a lightweight web framework to build our application's interface.
  It allows users to upload images and see the processed results in a clean web page.
- **OpenCV (For image preprocessing)**
  OpenCV (Open Source Computer Vision Library) is used to process images.
  In this project, it converts images to grayscale to improve OCR text detection.
- **Tesseract OCR (For text extraction)**
  Tesseract is an open-source OCR engine developed by Google.
  It reads characters from images and extracts them as editable text.
- **spaCy (For entity recognition from extracted text)**
  spaCy is a Natural Language Processing (NLP) library.
  It identifies important entities like dates, organizations, numbers, and locations from the extracted text.
- **YOLOv8 (For object detection in images)**
  YOLO (You Only Look Once) v8 is a deep learning model for real-time object detection. It scans the image once and detects multiple objects (like person, car, and laptop) with high accuracy.

Implementation Approach

i. **Upload Image**: User uploads an image through a Flask web app.
ii. **OCR (ocr_module.py)**: Extracts text using Tesseract.
iii. **NLP (nlp_module.py)**: Analyzes text using spaCy and extracts entities like dates, numbers, and organizations.
iv. **Object Detection (object_detection.py)**: YOLOv8 identifies objects in the image.

v. **Integration (image_analysis.py)**: Combines OCR, NLP, and Object Detection into a structured report.

vi. **Flask App (main.py + index.html)**: Displays extracted text, entities, and detected objects in a user-friendly GUI.

4. Demonstration

- **Steps in Demo Video**:
  1. Brief introduction of team members.
  2. Explanation of project motivation & objectives.
  3. Technical walkthrough (modules and workflow).
  4. Running the Flask app: upload an image and show results (text + entities + objects).
  5. Summary of results.

Demo Video Link (ctrl + click): click here

5. Key Results & Observations

- Successfully extracts text from clear printed documents and invoices.
- Named entities (like dates, organizations, and numbers) are recognized accurately.
- Objects like person, car, laptop, dog, etc. are detected with YOLOv8.
- Works best with high-quality images (less noise, clear text).
- Fast and efficient – processes image text extraction and object detection in just a few seconds.

6. Future Improvements

- Extend support for handwritten text recognition in addition to printed text.
- Improve OCR accuracy for noisy/blurred images using advanced preprocessing.
- Add multilingual support to extract and analyze text in multiple languages.
- Enhance the web interface with result downloads (PDF/CSV reports).
- Deploy the project on cloud platforms for real-time access anywhere.

7. Conclusion

The project demonstrates how OCR, NLP, and Object Detection can be integrated into a single application for extracting rich information from images. This system can be extended to domains like document processing, digital archives, accessibility for visually impaired users, and intelligent automation.