

Project Documentation

Team: Py 02

Project Title:

Object Detection Using YOLO

Project Overview:

This project is a web-based object detection application using Flask and YOLOv3 (You Only Look Once version 3). The main objective is to provide users with a simple interface to upload images and receive detected objects in the images with their respective bounding boxes and confidence scores.

Objectives

- **Image Upload and Handling:** Allow users to upload images via a web interface.
- **Object Detection:** Utilize YOLOv3 to detect objects within the uploaded images.
- **Result Display:** Show the detected objects with bounding boxes and labels on the image, along with confidence scores.

Scope

- **Frontend:** A simple HTML page (index.html) for image upload.
- **Backend:** A Flask server to handle image uploads, process the images using YOLOv3, and return the detection results.
- **Object Detection:** Integrate YOLOv3, configured to use pre-trained weights and configuration files (yolov3.cfg, yolov3.weights, coco.names for class labels).

Project Team:

1. Sai Prasad Patchipulusu(TL) - Designing and developing User Interface
2. Shlok Yadav - Model Selection and Training
3. Komal S Kallanagoudar - Model Implementation
4. Surekha Vijay Pawar -Dataset Collection and documentation
5. Priya Sanjay Sarmalkar- Dataset Collection and Data Preprocessing

Project Technologies:

- **Flask:** Flask is a micro web framework written in Python, known for its simplicity and flexibility. It provides the essentials needed to build web applications, such as routing, request handling, and

rendering templates, without imposing a specific structure or requiring large libraries.

- **OpenCV:** OpenCV (Open-Source Computer Vision Library) is designed to accelerate the use of machine perception in commercial products. It includes several hundred computer vision algorithms and provides functionalities such as image processing, video capture, and analysis, which are crucial for the image manipulation tasks in this project.
- **YOLOv3 (You Only Look Once version 3):** YOLOv3 is a real-time object detection system that processes images using a single neural network. It divides images into regions and predicts bounding boxes and probabilities for each region, offering a high-speed and high-accuracy method for detecting multiple objects in an image.
- **NumPy:** NumPy is a library for Python that supports large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays. It is essential for handling the numerical computations involved in image processing and manipulation.
- **HTML/CSS:** HTML (HyperText Markup Language) and CSS (Cascading Style Sheets) are the core technologies for building and designing web pages. HTML provides the structure, while CSS styles the appearance of the web pages, making the user interface more appealing and user-friendly.
- **JavaScript:** JavaScript is a versatile scripting language used to create dynamic and interactive effects within web browsers. In this project, it enhances the user experience by providing immediate feedback and interactivity, such as handling image uploads and displaying detection results without reloading the page.
- **cv2.dnn:** The cv2.dnn module in OpenCV provides deep neural network functionality, enabling the loading and execution of pre-trained models like YOLOv3. It allows for efficient image analysis and object detection by leveraging the power of deep learning.
- **OS Library:** The OS library in Python provides a way of using operating system-dependent functionality such as reading or writing to the filesystem, creating directories, and handling file paths. It is crucial for managing the uploaded and output images in this project.

Project Setup:

- Download YOLOv3 Weights and Configuration Following Files
 1. YOLOv3 configuration file (yolov3.cfg)
 2. YOLOv3 weights file (yolov3.weights)
 3. COCO names file (coco.names)
- Install the following requirements using pip
 1. Flask==2.0.2
 2. numpy==1.21.2
 3. opencv-python==4.5.3.56 or Use the command *pip install -r requirements.txt*

- Create Uploads and Output Directories
- Run the Application using the command

Python app.py

- Access the Application

Open your web browser and navigate to: <http://127.0.0.1:5000/>

Project Resources/ Reference:

- [1].Shaukat Hayat, She Kun, Zuo Tengtao, Yue Yu, Tianyi Tu, Yantong Du worked on A Deep Learning Framework Using Convolutional Neural Network for Multi-class Object Recognition paper released in 2018 3rd IEEE International Conference on Image, Vision and Computing <https://ieeexplore.ieee.org/document/8492777>
- [2].Zewen Li, FanLiu,Wenjie Yang, Shouheng,Peng and Jun Zhou worked on A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects paper released in IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, VOL. 33, NO. 12, DECEMBER 2022 <https://ieeexplore.ieee.org/document/9451544>
- [3].Rahul Chauhan, Kamal Kumar Ghanshala, R.C Joshi worked on Convolutional Neural Network (CNN) for Image Detection and Recognition paper released in 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC) <https://ieeexplore.ieee.org/document/8703316>
- [4].Prasoon Bharat Mishra1, Abdul Malik2, M.Safa,Saranaya G4,Arun D5 worked on Enhanced Object Detection with Deep Convolutional Neural Networks for Vehicle Detection paper released in 2022 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS) <https://ieeexplore.ieee.org/document/10047323>
- [5].Xin Hu, Hua Ouyang and Yang Yin worked on Image Recognition based on Convolution Neural Network paper released in 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC) <https://ieeexplore.ieee.org/document/9339197>

Project Risks:

- The Image format should be JPEG and PNG
- The application detects the objects which are present in the coco.names

□ Thank you □