NIRF-2024
Engineering Rank Band (151-200)
Pharmacy Rank - 77
Innovation Rank Band (11-50)

www.kiet.edu
Delhi-NCR, Ghaziabad

KIET GROUP OF INSTITUTIONS
Connecting Life with Learning

Accredited by
NAAC with Grade "A+"

AI Project (AI101B)

Even Semester
Session 2024-25

Object Detection In Images

Ansh Mishra (202410116100030)
Abhishek Sharma(202410116100009)
Ankit Sisodia (202410116100029)
Ashutosh Singh(202410116100043)

Project Supervisor:

Mr.Apoorv Jain
Asst.Professor

Object Detection in Images

Object Detection is a key task in Computer Vision that combines classification and localization.

It allows systems to identify what an object is and where it is located in an image.

Widely used in autonomous vehicles, security systems, healthcare, and more.

Classical methods (like Haar Cascades) and modern deep learning models (like YOLO and SSD) are used.

This project focuses on implementing object detection using YOLOv8 and Python on Google Colab.

Understanding Object Detection Techniques

Exploring AI's Role in Object Detection

- **Definition of Object Detection**

  Object Detection combines Image Classification with Localization.

- **Real-world Applications**

  Used in self-driving cars, security systems, and medical imaging.

- **AI Models in Focus**

  Exploring how AI detects objects using YOLO models.

- **Implementation Tools**

  Project focuses on Python programming and Google Colab for implementation.

- **Importance of Localization**

  Localization helps identify the precise location of objects in images.

Comparing Object Detection Techniques

A comparative analysis of detection methods

- **Implement object detection techniques**

  Utilize Haar Cascades and YOLOv8 for detection tasks.

- **Analyze performance metrics**

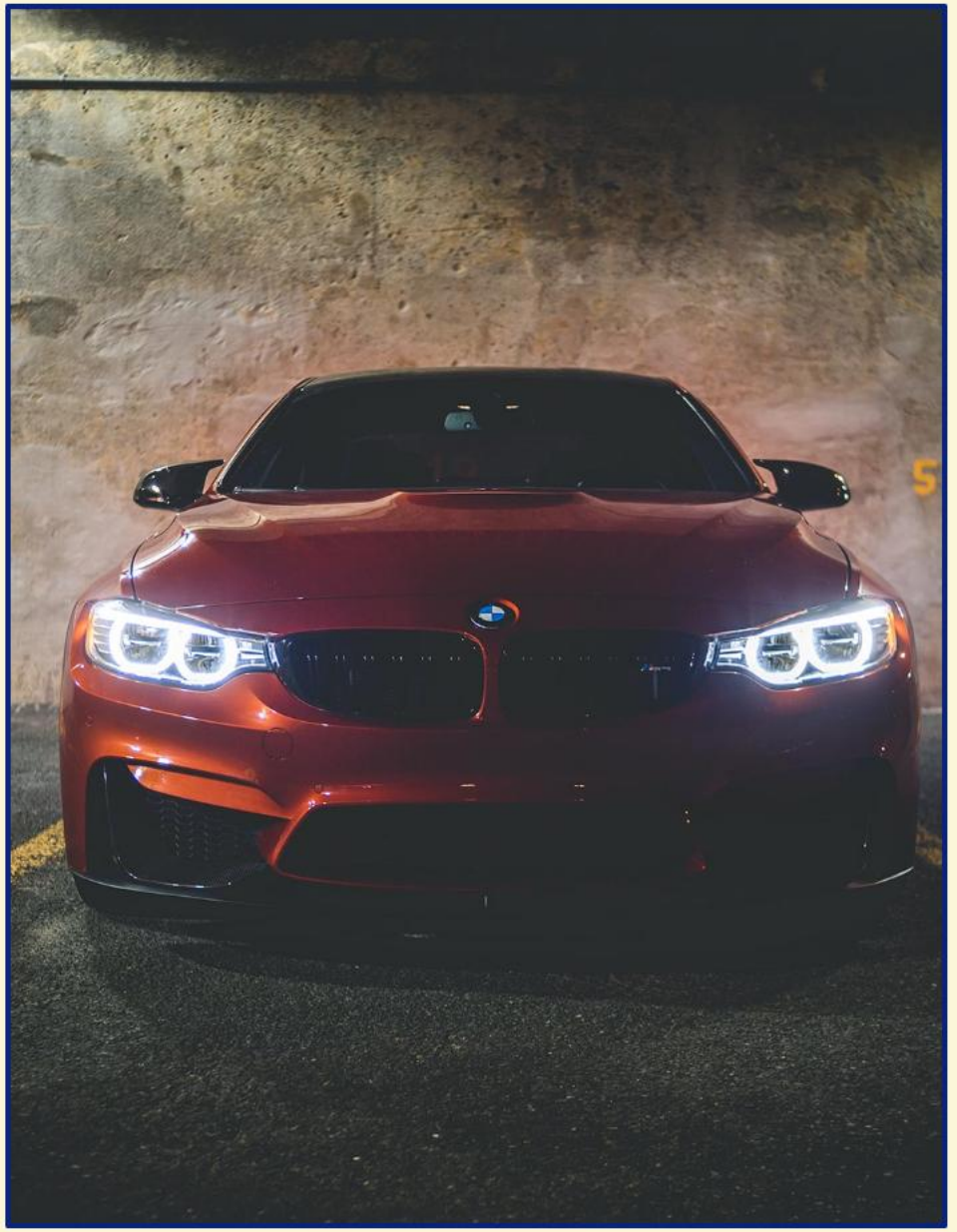
  Focus on comparing accuracy and speed of detection methods.

- **Use of Google Colab**

  Leverage Google Colab for GPU-accelerated development.

- **Gain practical skills**

  Enhance your AI and computer vision skills through hands-on practice.



Essential Tools for Image Detection

- **Python as the primary language**

  Python is the main programming language utilized for object detection tasks.

- **Google Colab for GPU access**

  Google Colab provides a cloud-based platform with GPU capabilities for enhanced processing.

- **OpenCV for image processing**

  OpenCV is used for handling images and implementing Haar Cascades for detection.

- **YOLOv8 for object detection**

  YOLOv8 from Ultralytics is employed as a deep learning-based object detector.

- **Matplotlib for visualization**

  Matplotlib is utilized for visualizing results and outputs from the object detection process.

Advanced Object Detection Techniques

Overview of methodologies for object detection

Image Collection

Collected a diverse range of images including faces, vehicles, and objects for analysis.

Face Detection with Haar Cascades

Implemented Haar Cascades for effective face detection in images.

Object Detection with YOLOv8

Utilized YOLOv8 for advanced and accurate object detection in various scenarios.

Bounding Box Annotation

Detected and labeled objects using bounding boxes for clear visualization.

Results Visualization

Visualized results through Colab, making findings accessible and understandable.

Code Workflow for YOLOv8 Detection

Install ultralytics

Begin by setting up the ultralytics library for YOLOv8 functionality.

Import libraries

Include necessary libraries to enable image processing and model functionality.

Upload and preprocess images

Prepare images for detection by uploading and preprocessing them correctly.

Load yolov8n.pt model

Load the YOLOv8 model file to initiate object detection tasks.

Run detection

Execute the model to detect objects within the uploaded images.

Show & download output

Display the results and provide an option to download the processed images.

Performance Comparison of Detection Models

Comparative analysis of object detection models

YOLOv3 Performance

High accuracy in detecting small and overlapping objects; mAP score 0.80, inference time 45ms.

SSD MobileNet Performance

Faster detection with slightly lower accuracy; mAP score 0.73, inference time 30ms.

Haar Cascades Limitations

Effective only for clear, frontal faces; best suited for simpler detection tasks.

Diverse Applications of Object Detection

- **Autonomous Vehicles**

  Utilize object detection to identify pedestrians and traffic signs, enhancing road safety.

- **Healthcare**

  Identifies tumors and fractures in medical scans, aiding in diagnostics and treatment planning.

- **Surveillance**

  Detects suspicious activities in real-time, improving security measures in public spaces.

- **Retail Analytics**

  Employs people counting and shelf analytics to optimize inventory and enhance customer experience.

- **Gaming & AR**

  Facilitates real-time interaction with the environment, enhancing the immersive gaming experience.

Understanding Limitations of Object Detection

Generalization Issues

Pre-trained models may not generalize well to all data types encountered in real-world scenarios.

Lighting and Visibility

Effective object detection requires optimal lighting and clear visibility of objects to function accurately.

Resource Intensive

High resource usage is a challenge as GPU is often required for fast processing speeds in model training.

Data Annotation

Preparing datasets is often annotation-heavy, requiring significant time and effort to label data accurately.

Ethical Concerns

The use of object detection raises ethical issues, including surveillance, bias, and privacy considerations.

Insights on Computer Vision Project

Hands-on Exposure to AI

The project provided practical experience in computer vision using AI technologies.

Effectiveness of YOLOv8

YOLOv8 was found to be effective and easily deployable using Google Colab.

Future Enhancements

Future plans include implementing real-time webcam detection capabilities.

Object Tracking Integration

Exploration of integrating object tracking features for advanced analysis.

Mobile Device Deployment

Plans to deploy the solution on mobile devices or Raspberry Pi for wider accessibility.