**Visual Object Detection in Domestic Settings**

This project focuses on object detection in domestic environments using state-of-the-art deep learning models. The goal is to detect and classify objects in various indoor settings such as bedrooms, bathrooms, living rooms, and children's rooms. The project compares the performance of several models, including YOLOv5, YOLOv8, YOLOv11, Detectron2, and MobileNet SSD, to determine the best model for indoor object detection.

**Project Overview**

The project evaluates the performance of different object detection models in domestic settings. The models are trained and tested on indoor environments, and their accuracy, confidence scores, and ability to handle visual clutter and lighting variations are compared. The results are presented in Excel files, which include detailed probability analysis for each detected object.

**Key Features**

* Object Detection: Uses advanced deep learning models to detect and classify objects in images.
* Room-Specific Predictions: Predictions are made for different rooms (e.g., bedroom, bathroom, living room) with probabilities for each detected object.
* Probability Analysis: Includes detailed probability analysis for each detected object, including whether the object is facing the door or not.

**Models Evaluated**

The following models were evaluated in this project:

1. YOLOv5
2. YOLOv8
3. YOLOv11
4. Detectron2
5. MobileNet SSD

**Results**

* Detectron2 performed the best in indoor settings, with high confidence and clear bounding boxes, though it has some biases (e.g., classifying all tables as dining tables).
* YOLOv11 achieved higher accuracy with fewer parameters, making it suitable for resource-constrained devices.
* MobileNet SSD is lightweight and fast but less accurate compared to other models.

**Project Structure**

The project is organized into the following components:

**Colab Notebooks**

1. **Code for running YOLOv5 model:**

<https://colab.research.google.com/drive/1VtSlOKf7A95QHX6ds51UlafNzS_wrcIt?usp=sharing>

1. **Code for running YOLOv8 model:**

<https://colab.research.google.com/drive/1w06IdcsvDyeDBcBtvt4Ga2UGr7LLymI8?authuser=1#scrollTo=Y8cDtxLIBHgQ>

1. **Code for running YOLOv11 model:** <https://colab.research.google.com/drive/15DkE8sTp24OYaOiRPHVQ1YpQP07AzT6Z?usp=sharing>
2. **Code for running Detectron2 model:** <https://colab.research.google.com/drive/1liOG6JU7UxJP72Uba06W3ESJnTPd8EQi>
3. **Code for running MobileNet SSD model :**

<https://colab.research.google.com/drive/19eVa5PbZ_BYStbY22beua_UzJI8VpDZF?usp=sharing>

**Colab Notebooks**

* **YOLO 5**: [Link to Colab Notebook](https://colab.research.google.com/drive/1VtSlOKf7A95QHX6ds51UlafNzS_wrcIt?usp=sharing)
* **YOLO 11**: [Link to Colab Notebook](https://colab.research.google.com/drive/15DkE8sTp24OYaOiRPHVQ1YpQP07AzT6Z?usp=sharing)
* **YOLO 8**: [Link to Colab Notebook](https://colab.research.google.com/drive/1w06IdcsvDyeDBcBtvt4Ga2UGr7LLymI8?authuser=1#scrollTo=Y8cDtxLIBHgQ)
* **Detectron Sample Code**: [Link to Colab Notebook](https://colab.research.google.com/drive/1liOG6JU7UxJP72Uba06W3ESJnTPd8EQi)
* **MobileNet-SSD**: [Link to Colab Notebook](https://colab.research.google.com/drive/19eVa5PbZ_BYStbY22beua_UzJI8VpDZF?usp=sharing)

**Excel Files**

* Detectron\_Predictions.xlsx: Contains predictions made by the Detectron2 model across different rooms.
* Yolo11\_Predictions\_Girl.xlsx: Contains predictions made by the YOLOv11 model across different rooms.
* YOLOV11\_Predictions\_Consolidated1.xlsx: Consolidated predictions from **YOLOv11 across various environments.**

**How to Use**

1. Colab Notebooks: Open the provided Colab links to explore the code and run the models. Each notebook contains detailed instructions on how to set up and run the object detection models.
2. Excel Files: The Excel files contain the predictions made by the models. You can open these files to view the probabilities and classifications for each object detected in the images.

**Future Work**

* Model Optimization: Further optimize the models to improve accuracy and reduce inference time.
* Real-Time Detection: Implement real-time object detection using webcam or video streams.
* Deployment: Deploy the models on edge devices for real-world applications.

**Acknowledgments**

* YOLO: Thanks to the creators of YOLO for providing an efficient and accurate object detection framework.
* Detectron: Thanks to Facebook AI Research for the Detectron framework.
* MobileNet: Thanks to the creators of MobileNet for providing a lightweight and efficient model for object detection.

**Example Repository Structure**

Object-Detection-Capstone/

│

├── Colab\_Notebooks/

│ ├── YOLO\_5.ipynb

│ ├── YOLO\_8.ipynb

│ ├── YOLO\_11.ipynb

│ ├── Detectron\_Sample.ipynb

│ └── MobileNet\_SSD.ipynb

│

├── Predictions/

│ ├── Detectron\_Predictions.xlsx

│ ├── Yolo11\_Predictions\_Girl.xlsx

│ ├── YOLOV11\_Predictions\_Consolidated1.xlsx

│ └── README.md

│

├── requirements.txt

├── LICENSE

└── README.md

**Example requirements.txt**

torch

torchvision

opencv-python

numpy

pandas

matplotlib