Object Detection Module Project

Introduction

In this group project, students will embark on an engaging journey to develop an object detection model for identifying various road conditions and turns in images captured on roads in Gilgit. The primary goal is to enable the model to detect right turns, left turns, straight roads, and unexpected road conditions such as landslides. This project is designed to provide students with hands-on experience in data gathering, data labeling, and the implementation of object detection models, specifically YOLO (You Only Look Once).

Project Objectives

1. Gather a dataset of road images in Gilgit, with a minimum requirement of 50 images per group, captured by the students themselves.
2. Annotate and label the collected data with information about road conditions and turns.
3. Train a classifier to categorize images into predefined classes: right turn, left turn, straight, and unexpected.
4. Implement the YOLO (You Only Look Once) object detection model to precisely locate and classify turns and unexpected road conditions in images.
5. Evaluate the model's performance and fine-tune it as needed.
6. Document the entire process, including data collection, labeling, model training, and evaluation.
7. Create a GitHub repository for the project and ensure that all project requirements, code, documentation, and datasets are added to the repository.

Project Phases

Phase 1: Data Collection

- In groups of up to 3 students, gather a diverse dataset of road images from Gilgit. These images should cover a variety of road types, conditions, and weather.
- There should be at least 10 image per class.

Phase 2: Data Labeling

- Annotate and label the collected dataset, specifying whether each image represents a right turn, left turn, straight road, or an unexpected road condition (e.g., landslides).
- Number of classes and objects in labeling process is not hard rule, but you can create at least above-mentioned classes.
- Utilize annotation tools or platforms to streamline this process.

Phase 3: Classifier Training

- Train a classification model (e.g., a convolutional neural network) using the labeled dataset to categorize images into the predefined classes.
- Evaluate the classification model's performance using appropriate metrics such as accuracy, precision, recall, and F1-score.
- You can show one example image per class for evaluation.

Phase 4: YOLO Implementation

- Introduce students to the YOLO (You Only Look Once) object detection algorithm.
- Preprocess the dataset to convert labels into YOLO-compatible format.
- Train the YOLO model to detect and classify turns and unexpected road conditions in images.

Phase 5: Model Evaluation

- Assess the YOLO model's performance using metrics such as Mean Average Precision (mAP) and Intersection over Union (IoU).

- Fine-tune the model based on the evaluation results to improve its accuracy and robustness.
- You can show one example image per class for evaluation.

Phase 6: Documentation

- Create a comprehensive document detailing the entire project, including the data collection process, annotation techniques, model training procedures, and evaluation results.
- Include code snippets, visualizations, and any challenges faced during the project.
- Ensure that all project requirements, code, and datasets are added to the GitHub repository.

Assessment and Grading

- Assessment of the project will be based on the quality of the dataset collected, accuracy of the classification model, performance of the YOLO object detection model, completeness of the project documentation, and the presence of the GitHub repository with all project materials.

Resources and Tools

- Python programming language
- Deep learning frameworks (e.g., TensorFlow, PyTorch).
- YOLOv3 or YOLOv4 for object detection.
- Annotation tools (e.g., LabelImg, RectLabel, or any of yours choice).
- Data collection resources (captured by students).
- GitHub or similar version control systems for code management and project repository.
- Documentation tools for creating project reports (e.g., Microsoft Word).

Deadline

- The project deadline is **September 30, 2023**.