KENT STATE UNIVERSITY

Assignment #4

Total Points: 150

Office location: CAED 210H

Instructions

Collaboration Policy: For "Individual assignments," Collaboration is expected within the limits of discussing concepts and problems. However, each student must produce his/her own solution to the problems. For "Group assignments," each student should have specific contributions to the homework –by default, the instructor assumes equal contributions; if any disagreement between team members about the contributions of individuals, please talk to the instructor to initiate a peer evaluation process.

ChatGPT Policy: In general, please be transparent if you use ChatGPT and highlight the parts of the homework generated by ChatGPT. When you choose to use ChatGPT to provide some answers, please *1) use an online document to save the ChatGPT sessions that helped you produce the answer*; 2*) critically review the answers generated by ChatGPT, highlight the parts that you found that ChatGPT's answer needs improvements*.

General Expectations and Requirements for Homework:

- Please use considerable and uniform font sizes throughout the document to maintain consistency of the document. You may choose to highlight subheadings with either a bold or underlined feature.
- Please use bullet points wherever possible to make the answers clear and easy to follow by an educated reader.
- Please do not forget to reference additional data, hyperlinks or literature used as evidence or background information to support your claims and solutions in the document. Please list those references below your answer or at the end of the document.
- Please refer to the textbook and provide descriptive answers wherever possible.
- Please communicate with the instructor to clarify questions about the homework description BEFORE the submission; after the homework submission deadline, the students are responsible for the point losses due to different ways of interpreting the homework requirements.

All homework submissions should be submitted electronically on Canvas.

Construction Equipment Detection using YOLOv5

Student Name:
Date:
Course/Instructor:

In this assignment, you will develop an object detection model using YOLOv5 to detect construction equipment in images. You will utilize a dataset from Kaggle, preprocess the data, train the model, evaluate its performance, and visualize detection results.

By completing this assignment, you will:

- Gain experience in data preprocessing, including annotation format conversion.
- Understand dataset splitting for machine learning applications.
- Train an object detection model using YOLOv5.
- Evaluate and visualize model performance.

Submission Guidelines:

- Submit your **Jupyter Notebook (.ipynb)** with the following:
  - Clean, well-documented code.
  - Visualizations of training metrics and detection results.
  - Discussion of model performance and any observed trends.
- Provide a brief **report (PDF)** summarizing:
  - Steps taken in data preprocessing and training.
  - Model evaluation results.
  - Insights and potential improvements.

Here are some hints:

Step 1: Download the Dataset

- Use gdown to download the dataset from a shared Google Drive link.
- Unzip the dataset into a structured directory.

Step 2: Split the Dataset

- Divide the dataset into **training**, **validation**, and **test** sets.
- Ensure each split has separate images and labels folders.

Step 3: Convert Annotations to YOLO Format

- Read JSON annotation files and extract bounding box information.
- Convert annotation coordinates to YOLO format (.txt files).
- Save YOLO-formatted labels in the corresponding split directories.

Step 4: Configure YOLOv5 Dataset

- Create a construction_equipment.yaml configuration file.
- Define training, validation, and test data paths.
- List all class names and the number of classes (nc).

Step 5: Train YOLOv5 Model

- Clone the YOLOv5 repository and install dependencies.

- Train the model using **YOLOv5s** pretrained weights.
- Monitor loss and accuracy metrics during training.

Step 6: Evaluate Model Performance

- Compute performance metrics: **precision, recall, and mAP**.
- Use YOLOv5's val.py script to validate performance on the test set.

Step 7: Run Inference

- Apply the trained model to images from the test set.
- Visualize detection results by overlaying bounding boxes on images.

Step 8: Recommendations for Improvement (optional)

- Try data augmentation techniques for better generalization.
- Adjust hyperparameters such as **batch size, learning rate, and epochs**.
- Explore real-time applications using live camera feeds.