**Topics in Deep Learning: Hackathon Assignment**

**Day 1 Question Paper**

**Scope of this hackathon**

This is a hackathon style examination as a part of the elective course “Topics in Deep Learning”. The examination is conducted over two days and this paper is the day 1 assignment. Students are required to submit the final deliverables by 6 pm, 20th Apr 2023.

**Objective**

In this exam you are required to develop an Object Detection System using the Drone-detection-dataset as given by the examiners of the hackathon. You are free to use TensorFlow or PyTorch framework and any other library publicly available. The detailed problem statement is described below. You are required to train, validate and test your implementation using the dataset specified in Kaggle for this exam and achieve appropriate results ethically.

**Problem Statement**

Recently object detection and tracking have found widespread usage ranging from delivery services to military applications and many more. Colour (RGB) videos having objects such as Drones, Helicopter, Birds, and Airplanes are available and provide an opportunity to visually identify and interpret objects. Detecting objects that are important for the given domain (Such as security applications) and in real time is extremely critical. In this hackathon, we challenge you with the problem of detecting objects using the drone detection dataset. The task aims to detect objects of predefined categories (e.g., Drones, Airplane, Birds, Helicopter) from individual Videos.

1. Dataset:

You are required to use the dataset provided at Kaggle provided under the subheading “Task 1”.

1. Download the dataset (125 MB) having 280 RGB videos and 280 CSV files.
2. Description about CSV:

Every video has corresponding CSV File with number of rows = duration of the video/0.033334 and 5 columns. < ex: If video is of 10 sec, then 10sec/0.03334 = 300 rows >

Row entry start with ‘0 sec’ having description of bounding boxes in columns for the respective label in the frame.

Column description is as follows: 1st column is time of the frame, 2nd column is bounding box coordinates for Airplane, 3rd column is bounding box coordinates for Bird/Birds, 4th column is bounding box coordinates for Drone/Drones and 5th column is bounding box coordinates for Helicopter.

Note: Bounding Box Coordinators are [xmin, ymin, width , height]. If a frame has multiple objects of same type, ex: 2 birds then 3rd column will have 2 entries of bounding box [xmin1, ymin2, width1, height1; xmin2, ymin2, width2, height2]. Suppose a frame has Bird as well as Airplane then column 2 and 3 will have the bounding box values.

1. Use Pandas and Jupyter Notebook and divide the dataset into training and testing. (split ratio of your choice).
2. Building the Model:

Select a suitable pretrained model from the framework you have chosen. It could be any popular model (e.g Resnet 50) as the backbone. You can choose it based on accuracy, number of parameters, and also considerations of training time. (A highly complex model may take huge amount of time to train). Create your own convolutional and output layers based on number of labels. Make sure your model runs in Google Colab (Intel’s ONEAPI) in a reasonable time. Train the system using the trainset data. You can start with a smaller subset of data (to ensure the model is correct and to get an idea about the training time). Measure the training accuracy and loss, and ensure it converges and find the objects at every frame in given video. Finetune the hyper parameters and obtain the best possible accuracy. This is an object detection problem in videos. You are required to report the training loss and convergence on both label as well as the bounding boxes. Create visualizations on the test accuracy where you draw bounding boxes against the original image and also report accuracies, confidence score (label, boxes) against the required outputs. You should download test files from the URL provided (or Kaggle) and use them for improving performance. Submit the final test file with your results to Kaggle.

1. Checkpoints:

Morning 9 am to 1 pm: - Dataset understanding, Model building and Comparison.

Afternoon 2 pm to 6 pm: - Training Convergence - Final Results - Visualization - Upload files to Kaggle Deliverables.

1. The deliverables are: 1. Uploading final test results to Kaggle 2. Your final results shown to invigilator 3. Submission of the source code.

Link to original Dataset:<https://github.com/DroneDetectionThesis/Drone-detection-dataset>

Link to the Paper: <https://arxiv.org/pdf/2007.07396.pdf>

All the very best,

Faculty