# IT5005 Project Description

Rajendra Prasad Sirigina

AY 2024/25 Sem 2

## 1 Project Overview

In this project, you will be tasked with implementing object tracking in videos. This document provides only general guidelines. Changes/revisions, if any, will be communicated to you in class or through Canvas announcements.

## 2 Project Tasks

Object detection and tracking in videos play a crucial role in sports analytics, intelligent traffic management, autonomous vehicles, surveillance, etc. In this project, you would implement an object tracking algorithm for video data using YOLO and Byte Track algorithms. Ultralytics [1] provides wrappers for the easy use of these architectures. The major tasks are highlighted below.

1. **Setting up pre-trained models for object tracking in Google Colab:** Set up a Google Colab environment to run pre-trained models for object tracking tasks. It includes understanding the dependencies, importing necessary libraries, loading pre-trained models, and executing them to track objects in video sequences. A Jupyter notebook is provided to guide the exploration of pre-trained models. The expected outcomes are:

   (a) Installation of Ultralytics library to access the pretrain models. Use a YOLO11 model with pre-trained weights to detect and track objects in video feeds. The focus will be on understanding how to use this model to predict and establish tracking of objects across video frames. The models are pre-trained on the COCO dataset [2].

   (b) Exploration of basic hyperparameter tuning, which might include adjustments to settings like confidence thresholds and IoU (Intersection over Union) thresholds, to see their effects on object detection and tracking performance.

[1] `https://docs.ultralytics.com/`
[2] `https://cocodataset.org/#explore`

2. **Training and testing object tracking model with a custom dataset** In this task, you will train and test the object tracking model using a custom dataset. It involves selection and preparation of the dataset, training of the model, validation of its performance, and application of the trained model to the test dataset. The expected outcomes:

   (a) Training and validation of the YOLO11 model with a custom dataset, focusing on interpreting performance metrics such as mean average precision (mAP), confusion matrices and precision-recall curves.

   (b) Apply the trained model to predict object tracking in new video data, demonstrating the model's ability to generalize to unseen data.

3. **Exploration of YOLO and Tracking Model Architectures** Conduct a detailed investigation of YOLO models and tracking algorithms. The objective is to understand the intuition and motivation behind the origin and evolution of YOLO models and tracking algorithms. Prepare a four-page report summarizing the architecture and algorithm details of these models. For example, you should be able to articulate why a specific block is implemented.

   The report should be a single PDF document not exceeding 4 pages (excluding the cover page, figures, tables, and bibliography). Use a minimum font size of 10, single line spacing, and single column format. There is no specific template for the report. The figures and tables are at the end of the report. The report should contain only the model description and not the source code. Additionally, record a 15 minute presentation, highlighting the architecture and algorithm details of the YOLO and ByteTrack algorithms.

# 3 Deliverables and Deadlines

The list of deadlines and deliverables is presented in Table 1.

Table 1: Milestones and Deliverables

| **Milestones/Tasks** | **Deliverables** | **Due Date** |
|---|---|---|
| 1: Setting up pretrained models and analysis | Zip file of your code (Jupyter notebook) | Week 11 |
| 2: Training and testing on custom dataset | Zip file of your code (Jupyter notebook) | Week 12 |
| 3: Presentation | The video recording of your presentation | Week 13 |
| 4: Final Report | Final report in PDF format | Week 13 |