Name: Vishal Narayan Deka | Roll No: 234156031 | MTech 1st Year Robotics & AI

Video Analytics (EE-722) Course Project Report

Title

Single Football Player Detection and Tracking in a Match

Objectives

This project is an attempt to perform real time object detection and tracking of football players in a match using Yolov5, a very effective vision AI model for classification (detection) and ByteTRACK, a multi object tracker for estimating the bounding boxes. By integrating these technologies, I aim to enhance football analysis, offering insights into player movements and team dynamics. The report outlines the methodology, discusses challenges, presents experimental results, and highlights future directions in sports analytics.

Approach

I have followed the work of Piotr Skalski who has authored a notebook which performs multiple football players tracking. The player/object detection is carried out by YOLOv5. YOLOv5, an evolution of the YOLO (You Only Look Once) architecture, offers state-of-the-art object detection capabilities with remarkable speed and accuracy.

Initially, Piotr had hoped to bypass the custom model training process by using the popular COCO dataset which has 'humans' and 'sports ball' classes on the ready. This YOLOv5 model pretrained on COCO was tested on several dozen brief football game videos from the Bundesliga Data Shootout Kaggle competition. The model's performance fell short as it flagged numerous extraneous objects beyond the field—coaches, fans, maintenance staff, camera crews, and even spare balls—necessitating additional detection filtering logic, an undesirable prospect. Despite the dataset encompassing the requisite classes, disparities between the videos used for inference and the images employed for training became apparent. Faced with this predicament, the only recourse was to construct a bespoke dataset and embark on training a custom model. Frames at regular intervals in the videos were therefore annotated with bounding boxes and model weights were obtained. This weights file was made available for download, and the very same file was used to set the weights of the model in this project.

Post building a detector, the decision was made to utilize ByteTRACK, recognized as one of the state-of-the-art multi-object trackers, for the task of object tracking within the video. Internally, ByteTRACK doesn't rely on neural networks and derives its efficacy solely from elementary mathematical computations, particularly in comparing bounding box positions across individual frames.

Notably, ByteTRACK assumes responsibility only for tracking and not detection. Consequently, it circumvents the need for separate training, a characteristic not shared by some other trackers.

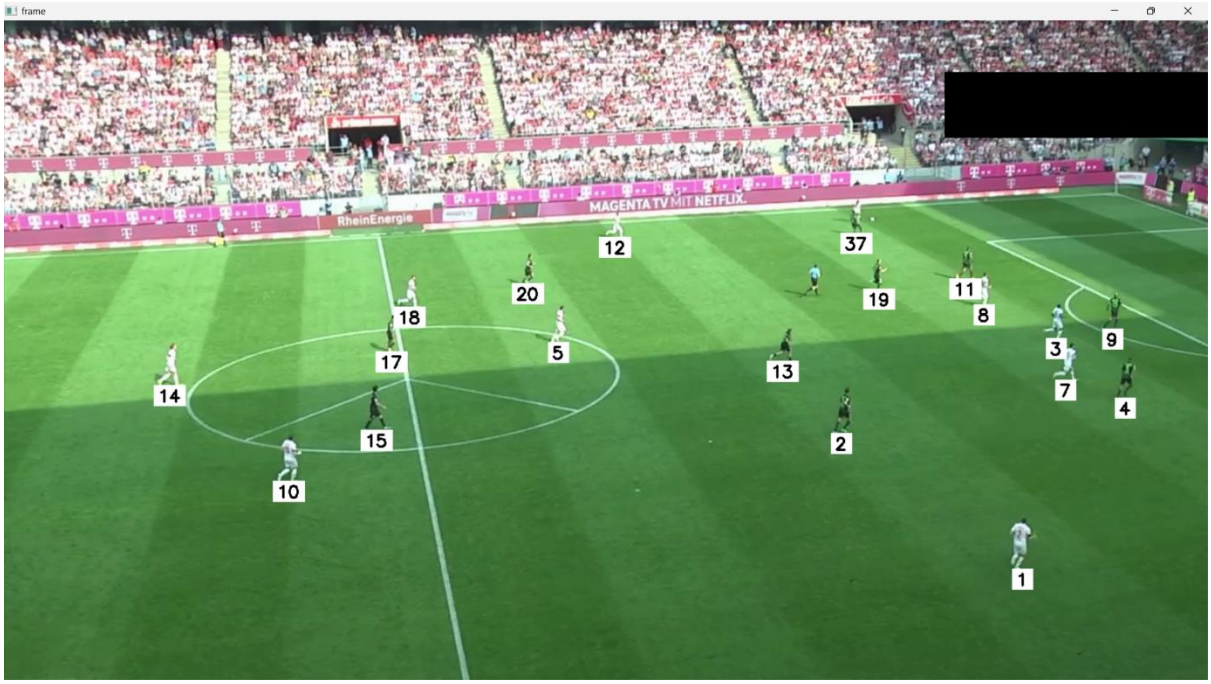
ByteTRACK offers the flexibility to interchange detectors seamlessly without necessitating a revamp of the entire code architecture.

Highlighted Points

After borrowing from the RoboFlow notebook, I had to make some changes in order to enable it to track a single player only. YOLOv5 would classify all objects on the field as players, which were then tracked by ByteTRACK. Since ByteTRACK would return Detection class objects (which had bounding box coordinates, tracking ids etc as member variables), it made logical sense to annotate only the selected tracking id into the image, consequently tracking only a single player. At the first frame, I provided the ability for a user to select a player to track, via a cv2 window which showed all detected players on the field and their respective tracking ids. From the first selection onwards, only the selected tracking id and corresponding bounding boxes are drawn, the rest are simply ignored. In case an object is lost, i.e. the selected tracking id no longer appears on the list of detections, the program would prompt the user to re-select the player of interest. It's worth noting that YOLOv5 and ByteTRACK form a rather robust pair and an object is mostly lost when it goes out of frame, and tracking was found to be satisfactory when players were occluded with each etc.

Conclusions

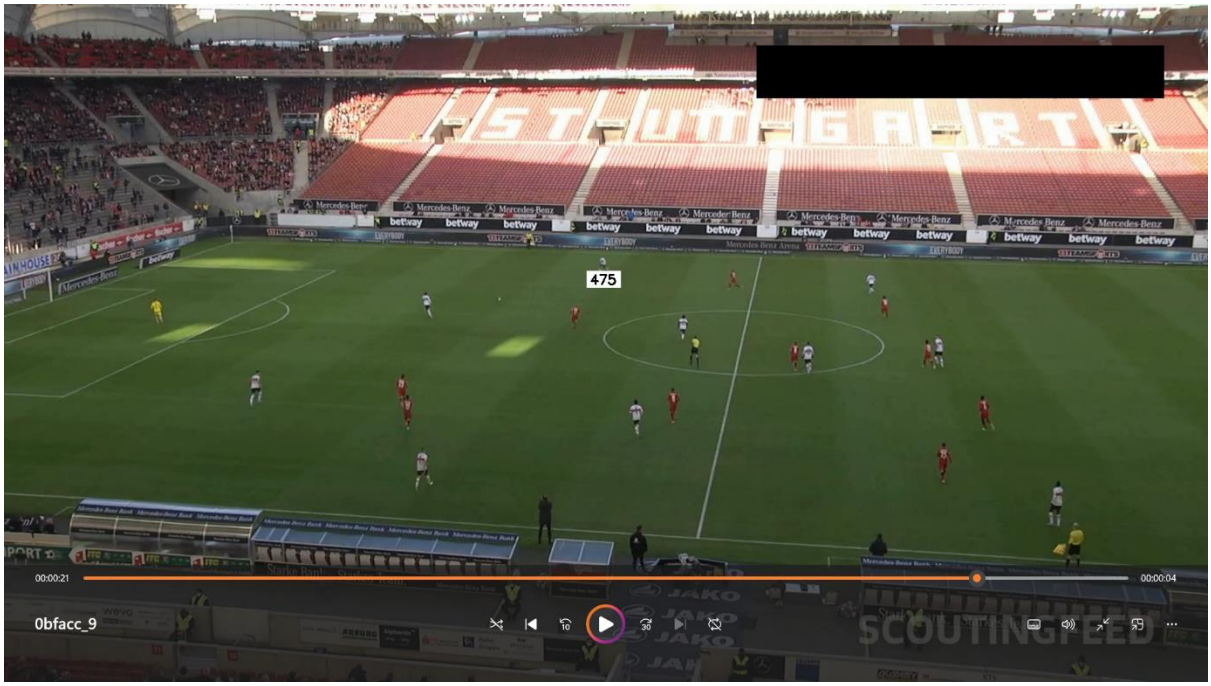
A frame on which YOLOV5 + ByteTRACK is performing object tracking looks like the screen capture shown below. Each player is marked with their tracking id. Notice how referees, people outside the play area and the crowds in the stadium aren't being tracked.



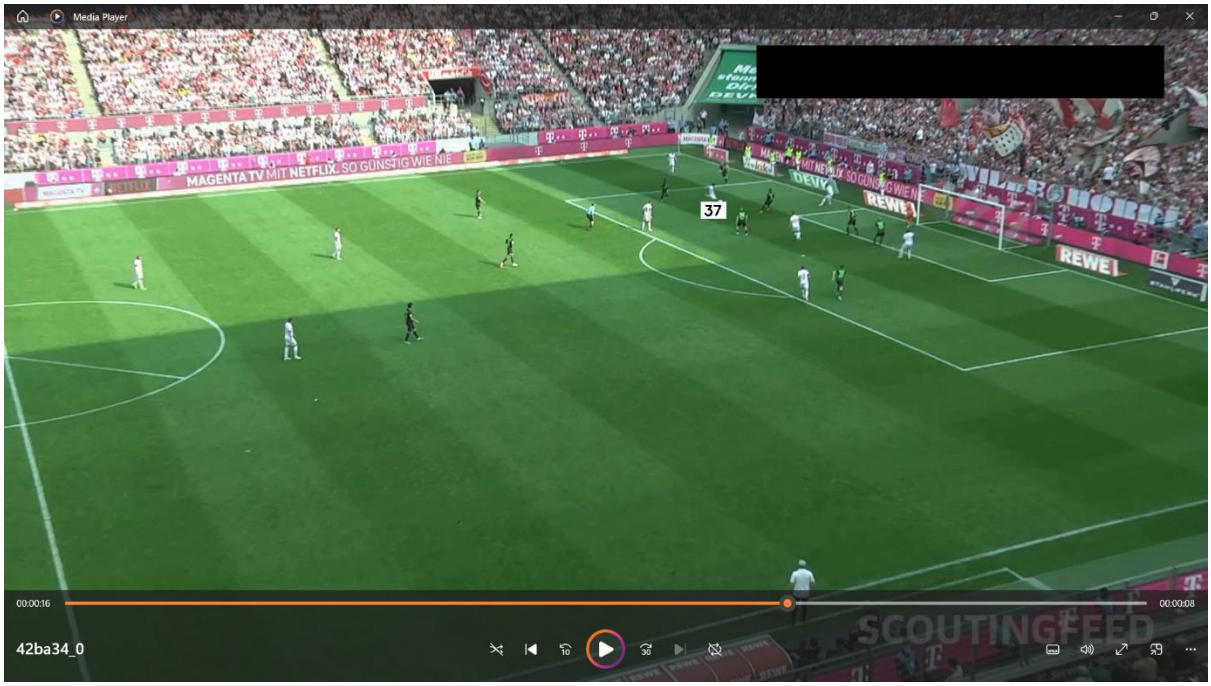
Tracking using the technologies mentioned in this project works best when the tracked object remains within frame and sufficiently away from other objects' bounding boxes. Passing through regions where there's close proximity between other bounding boxes can cause tracking ids to get lost or mixed up. One way to mitigate this problem would be to maintain a list of active and passive

trackers, but a bottleneck is that players of the same team are wearing same colors, hence have pretty identical color histogram information.

The following image is a snapshot of the single player tracker in action. The number 475 was selected as a target tracking id at some previous frame before this instant. The tracker performs very well when the target player is in an isolated region.



This next image demonstrates robustness of the tracker. It can be observed that bounding box is anchored well to the player given the presence of other players (objects of interests around him).



Links

The Jupyter Notebook for this project can be accessed by following this link. All necessary packages to be installed are mentioned in the notebook for easier replication of the results obtained here. A small subset of the dataset used is also provided in the zip file. The full dataset was part of a competition and can be accessed on Kaggle here.