# DS5216: Artificial Intelligence

## Programming Assignment 02: Player Tracking in Sports Videos

**Name:** S.G. Amararathna - DTS2402 **Date:** October 18, 2025

---

### 1. Introduction

The primordial aim of this project was to develop a computer vision model that would allow detecting and following players on different sports video clips. First and foremost, the player detection mechanism was supposed to be implemented through the application of the YOLO object detection framework. An additional keypoint detection model was applied as a bonus to identify the main body joints of the players and gain a more insightful view of their posture and movement. For these purposes, pre-trained models were used, which allowed achieving the aims in a convenient and effective way.

---

### 2. Methodology

#### 2.1. Dataset

The dataset for this project was manually collected from publicly available sources like YouTube. It consists of 7 short video clips from two different sports: football and cricket. Each video clip has a duration of approximately 7-10 seconds and a resolution of 720p.

#### 2.2. Tools and Frameworks

The project was developed using the following tools and libraries:

- **Language:** Python
- **Environment:** Google Colab (utilizing GPU acceleration)
- **Core Libraries:**
  - **PyTorch:** The underlying deep learning framework.
  - **Ultralytics YOLOv8:** For loading and running the pre-trained detection and pose estimation models.
  - **OpenCV (cv2):** For video processing tasks such as reading frames, writing output videos, and handling image data.

### 2.3. Model Implementation

Two pre-trained models from the YOLOv8 family were used:

1. **Player Detection:** The `yolov8n.pt` (nano) model was used for the primary task of detecting players. This model is pre-trained on the COCO dataset and is highly effective at identifying the "person" class.
2. **Keypoint Detection (Bonus):** For the bonus task, the `yolov8n-pose.pt` model was implemented. This model is specifically designed to detect human keypoints (pose estimation) in addition to identifying the person.

---

## 3. Results

This section presents the visual results obtained from the models.

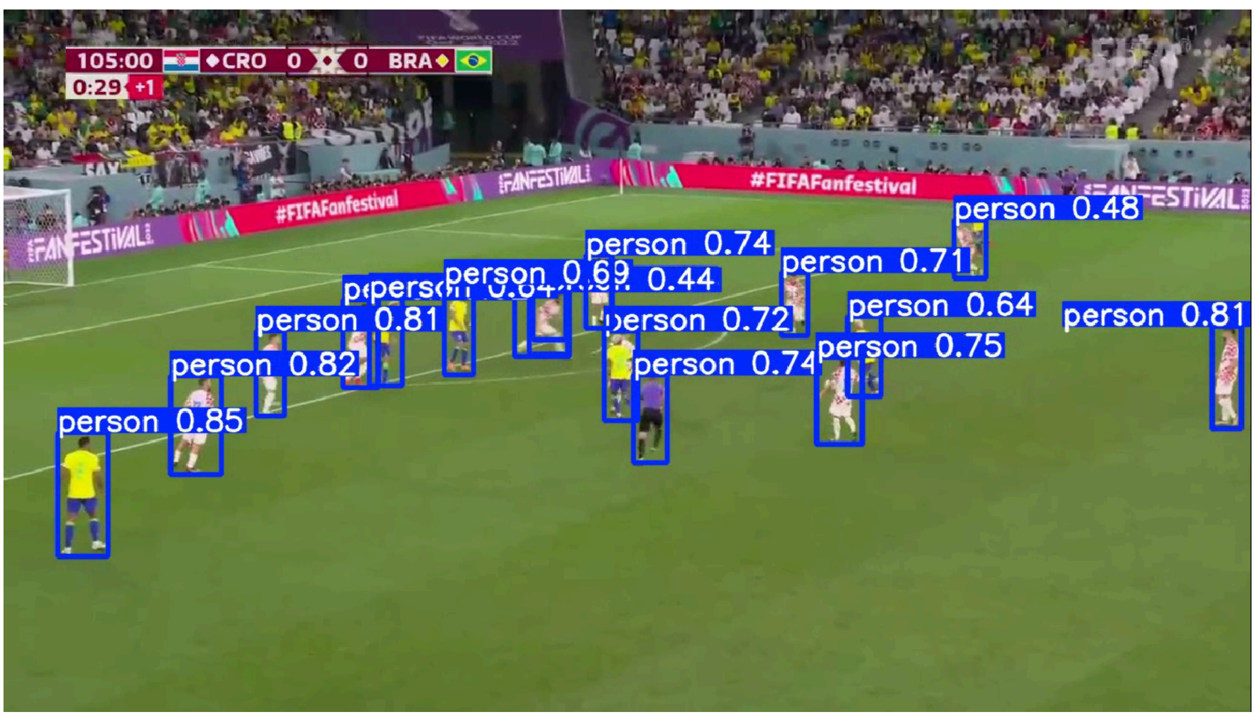


**Figure 1:** Successful player detection in a football video clip using the `yolov8n.pt` model.

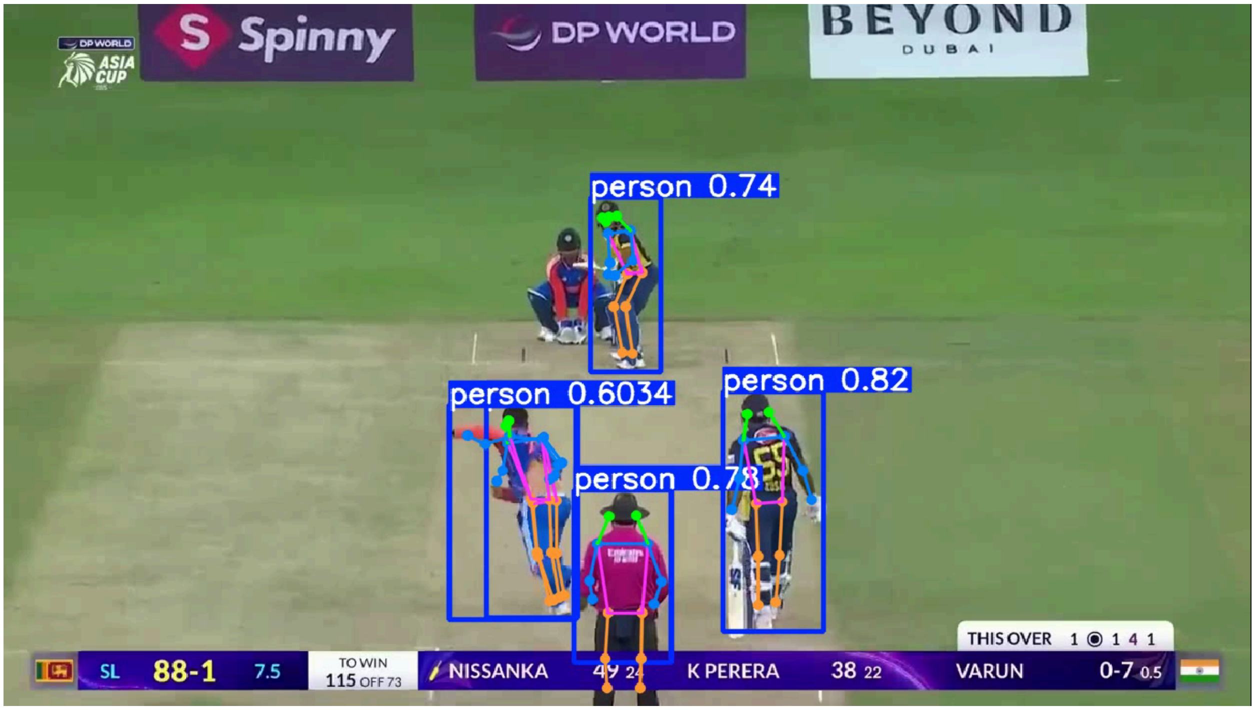


**Figure 2:** Keypoint estimation (pose) successfully applied to cricket players using the `yolov8n-pose.pt` model.

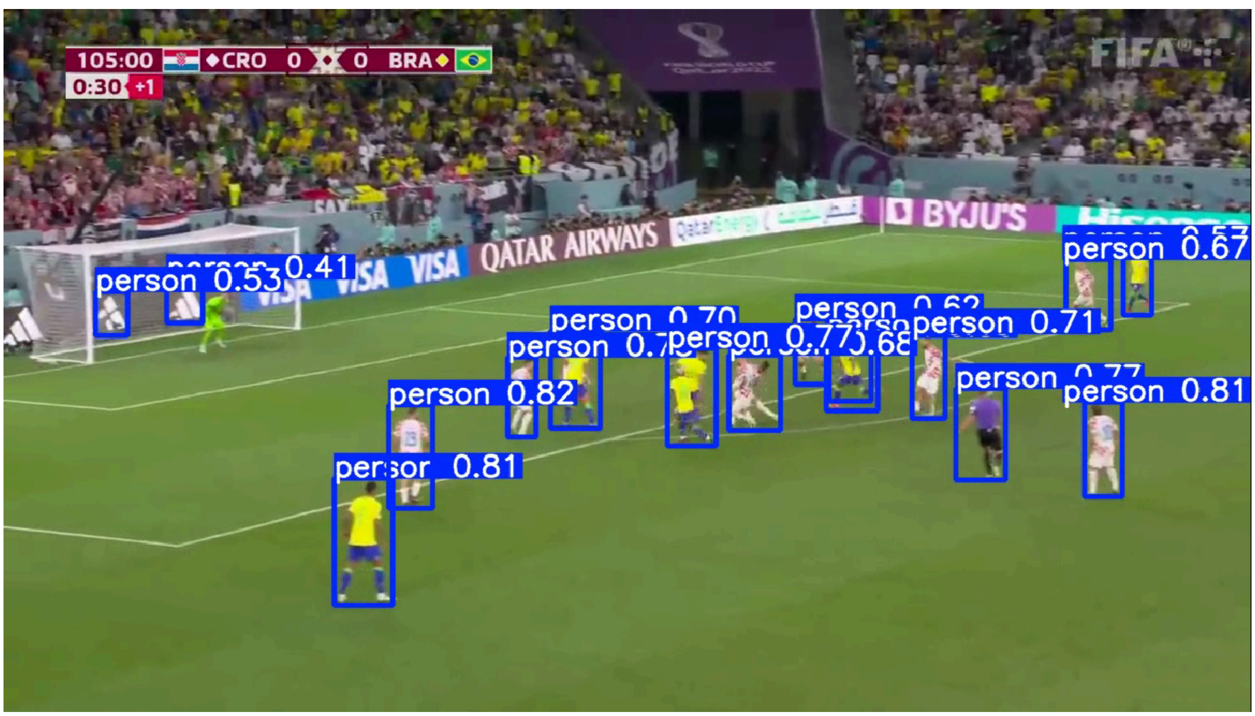


**Figure 3:** An example of a false positive, where the model incorrectly detects a figure on an advertisement hoarding.

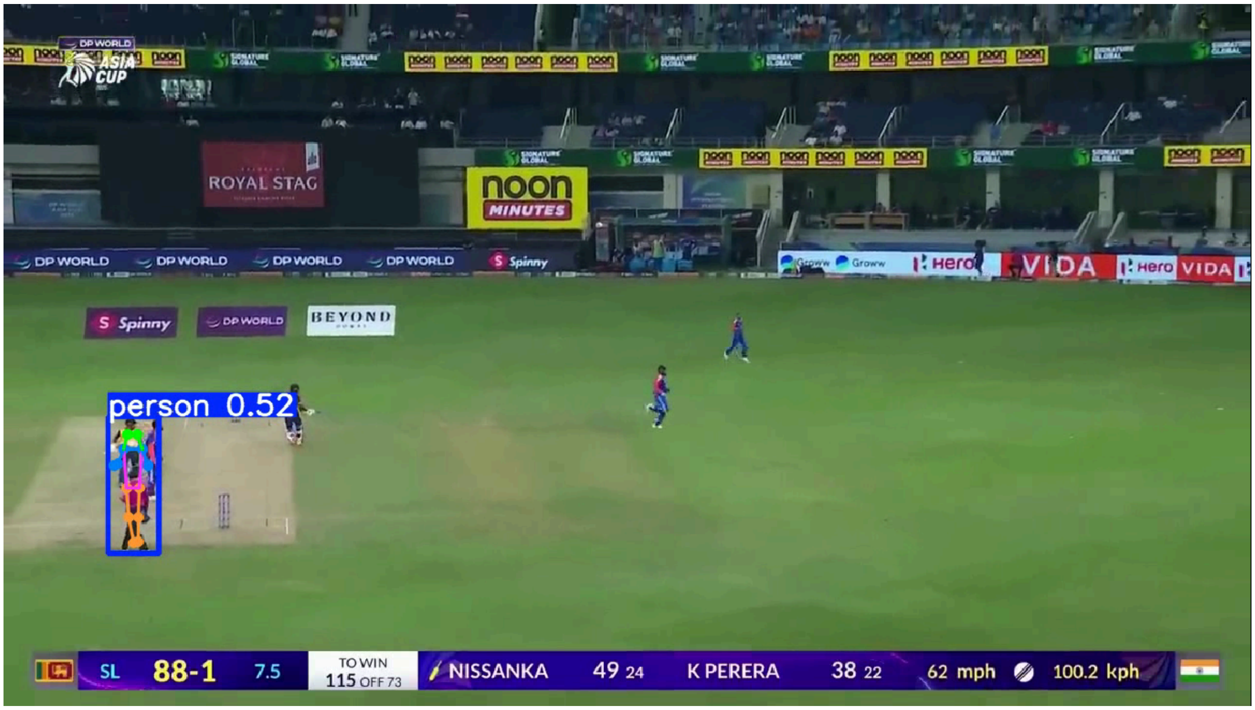


**Figure 4:** A limitation of the pose estimation model, failing to detect distant players in a cricket clip.

---

## 4. Discussion

### 4.1. Model Performance

The `yolov8n.pt` model demonstrated strong overall performance in detecting players. Qualitatively, the model exhibited **high precision**, meaning its detections were almost always correct. It also showed **good recall**, successfully identifying a large percentage of the visible players in most frames.

### 4.2. Limitations of the Models

Despite the strong performance, several limitations were observed:

- **False Negatives (Missed Detections):** The models failed to detect players under challenging conditions such as significant distance from the camera (small scale), heavy player clustering (occlusion), and motion blur.
- **False Positives (Incorrect Detections):** A notable limitation was the incorrect detection of human-like figures on electronic advertisement hoardings as "persons", as shown in Figure 3.
- **Pose Estimation Performance:** The `yolov8n-pose.pt` model's performance was significantly more limited. While it worked well for close-up players, it frequently failed to detect distant players (as seen in Figure 4), suggesting it is less robust to changes in scale and requires higher subject clarity than the standard detection model.

### 4.3. Possible Improvements

To address the identified limitations, the following improvements could be implemented:

1. **Fine-Tuning:** The most effective improvement would be to **fine-tune** the models on a custom dataset of sports players. This would teach the model to better distinguish between actual players and figures in advertisements.
2. **Using a Larger Model:** Using a larger model like `yolov8m.pt` (medium) could increase accuracy, especially for smaller, distant players, at the cost of slower processing speed.
3. **Confidence Threshold Adjustment:** Experimenting with the confidence threshold could provide a trade-off; increasing it might reduce false positives but could also lead to more missed detections.

---

## 5. Conclusion

In conclusion, the project was successfully able to show how a YOLOv8-based computer vision model can be implemented for player detection in sports videos. The pre-trained models performed remarkably well at teaching the primary object detection task, but restrictions around subject distance and clarity, as well as false positives were noted. The bonus pose estimation task further emphasized the greater complexity and difficulties with more comprehensive analysis. The findings offer a strong basis and areas for improvement, e.g., fine-tuning, that serve as straightforward directions to enhance the model's capabilities for further work.