1. Approach and Methodology

I implemented a soccer player re-identification system using **YOLOv7 for object detection** and **Deep SORT** for tracking. The goal was to detect and track players and the ball across frames in a soccer match video, assigning consistent IDs to each detected entity throughout the game.

- **YOLOv7** was used for fast and accurate detection.
- **Deep SORT** helped maintain identity tracking of players and the ball.
- I customized the detection class to include the soccer ball and players, saving the final output as a video.

2. Techniques Tried and Outcomes

✅ YOLOv7 pretrained weights were used to detect all objects, including the ball.

✅ Integrated **Deep SORT Realtime** with YOLOv7 detection results.

✅ Output was saved as a video (`output_tracked.mp4`) for visualization.

⛔ I initially attempted to use `torch.hub.load()` to load the model, which failed due to PyTorch version issues — resolved using `attempt_load()`.

3. Challenges Encountered

❗ **Model loading error** due to `torch.hub.load()` and incompatible keyword `path`.

❗ **Virtualenv issues** — had to create and isolate Python 3.10 environment due to version-specific conflicts.

❗ **UnpicklingError** due to PyTorch 2.6 defaulting to `weights_only=True` — resolved with updated loading logic.

❗ **Multiple tracking IDs** appeared briefly due to occlusion or players leaving/re-entering the frame.

❗ **UI features like score display and unique team colors** are not yet added due to time constraints.

4. What Remains (Future Work)

If i had more time/resources, we would:

🎯 Add **score detection** using OCR or scoreboard logic.

🎨 Assign **team-based colors** to tracked players (via color clustering or jersey color detection).

💡 Improve ID stability with ReID embeddings or deeper feature matching.

📈 Add analytics (pass count, player heatmaps, possession time).