**⚽ Player Re-Identification from Single Broadcast Feed**

**🔬 Submitted by:**

**Nakul Limbani**

**💼 For:**

**Stealth Mode ML Internship Assignment**

**🗓️ Date: 15 July 2025**

**📧 Contact:**

**✉️** [**nakulramesh2@gmail.com**](mailto:nakulramesh2@gmail.com) **✉️** [**nl0027@srmist.edu.in**](mailto:nl0027@srmist.edu.in)

**💻 GitHub:**

[**github.com/NakulLimbani/Player-Re-Identification**](https://github.com/NakulLimbani/Player-Re-Identification)

**💼 LinkedIn:**

[**linkedin.com/in/nakul-limbani-a65261240**](https://www.linkedin.com/in/nakul-limbani-a65261240/)

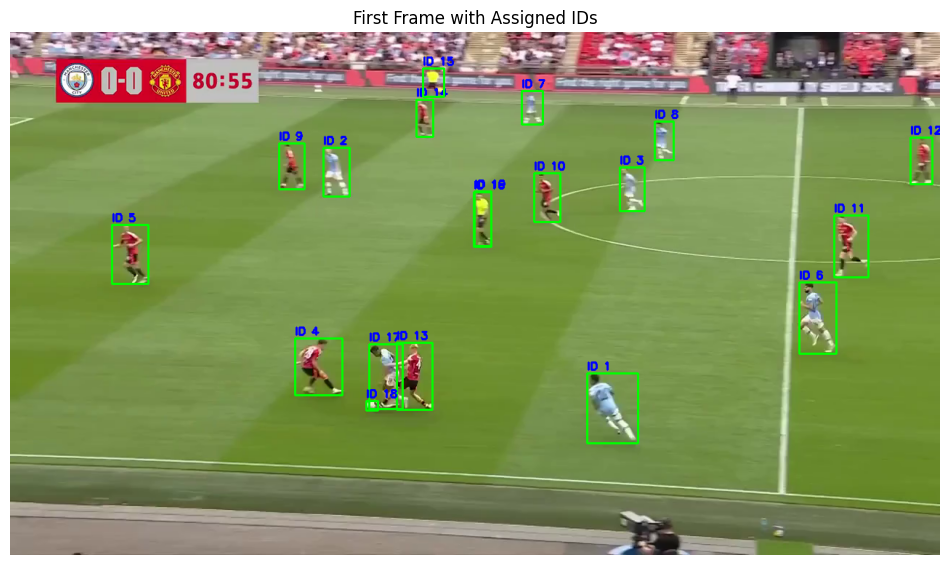
**🏅 Stealth Mode ML Internship Submission**

**📄 Final Report — Player Re-Identification from Single Broadcast Feed**

**✅ Overview**

This report details the approach, methodology, techniques, challenges, and future improvements for the **Player Re-Identification from a Single Broadcast Feed** project, implemented as part of the Stealth Mode ML Internship assignment.

The goal was to detect and uniquely identify football players throughout a single camera feed, while maintaining consistent IDs and preserving the original match commentary in the output video.



**🟢 Approach & Methodology**

**🎯 Objective**

* Detect all players in each frame of the broadcast video.
* Assign each player a **unique ID** and keep it consistent as long as possible.
* Handle player occlusions, motion blur, and crowded formations.
* Produce a final output video with IDs overlayed and original commentary audio preserved.

**⚙️ Detection**

* **Model Used**: YOLOv11 (Ultralytics YOLOv5 variant).
* **Class Filtering**: Only class index 2 (players) was used to filter out referees, crowd, and other objects.
* **Pre-processing**: Adjusted confidence and non-max suppression thresholds to balance detection accuracy and false positives.

**🔁 Re-Identification Logic**

* **ID Assignment**: Based on Intersection over Union (IoU) matching across consecutive frames.
* **Memory Mechanism**: Previous frame’s bounding boxes were stored, and IDs were reassigned if IoU > 0.3.
* **New IDs**: Assigned when no adequate match is found, ensuring every new player gets a unique ID.

**🖥️ Output Generation**

* Bounding boxes and IDs were drawn on each frame.
* Frames were written into a new video file.
* **Audio Preservation**: Original commentary audio was merged into the final output video using ffmpeg.

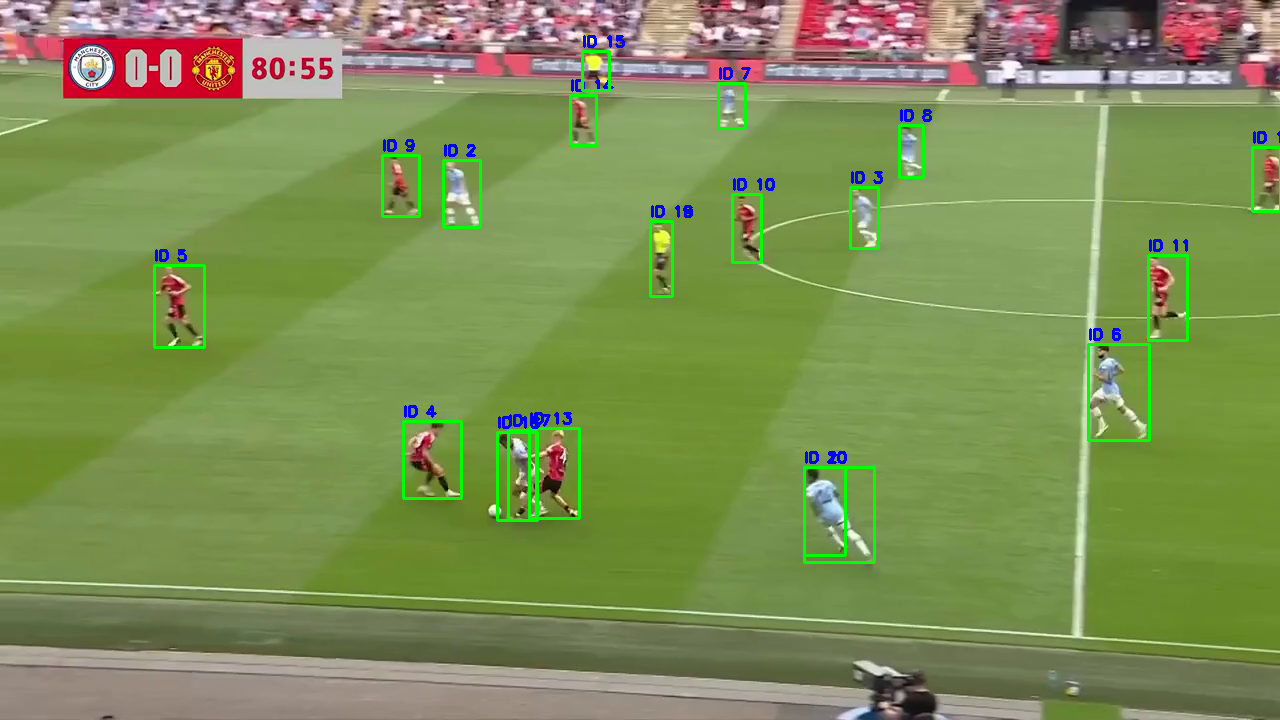
**🔎 Techniques Tried & Outcomes**

**1️⃣ IoU-based Tracker**

* **Outcome**: Provided a straightforward and fast way to track players frame-to-frame.
* **Limitations**: IDs may fluctuate slightly when players overlap heavily or during fast, erratic movements.

**2️⃣ Possible Techniques Considered**

* Appearance-based re-ID using deep features or jersey color histograms for improved consistency.
* Kalman filters for motion prediction and smoother trajectory estimation.
* OCR-based jersey number recognition to enforce hard ID constraints.



A football game on a field

AI-generated content may be incorrect.

**⚠️ Challenges Encountered**

* **Occlusion and Grouping**: Frequent overlap of players caused ID switches, which is a typical challenge in broadcast sports footage.
* **Fast Motion**: Rapid changes in player positions strained simple IoU-based matching.
* **Model Size Constraints**: The trained YOLO model (best.pt) exceeds 100 MB, requiring external hosting instead of direct repository push.
* **Audio Integration**: Merging video and audio streams while maintaining synchronization required careful handling with ffmpeg.

**🚀 Future Improvements**

* **Appearance Embeddings**: Incorporate player color features or deep visual descriptors to strengthen identity tracking.
* **Motion Prediction**: Use Kalman filters or advanced motion models to reduce ID switching.
* **OCR Integration**: Recognize jersey numbers for definitive ID assignment.
* **Extended Multi-camera Support**: Merge tracking across different camera angles or multiple feeds.

**📦 Deliverables Summary**

* ✅ Source code — Jupyter notebook and standalone Python script.
* ✅ Detailed README.md file with setup, run instructions, virtual environment guidance, and model link.
* ✅ Final report (this document).
* ✅ Final output videos:
  + With audio commentary.
  + Without audio (intermediate references).
* ✅ Intermediate archived outputs for comparison.
* ✅ Clean requirements.txt with minimal essential dependencies.

**💬 Contact**

👤 **Nakul Limbani**  
✉️ Email: [nakulramesh2@gmail.com](mailto:nakulramesh2@gmail.com) | [nl0027@srmist.edu.in](mailto:nl0027@srmist.edu.in)  
💼 LinkedIn: <https://www.linkedin.com/in/nakul-limbani-a65261240/>  
💻 GitHub: <https://github.com/NakulLimbani>

**⚽ Closing Note**

Thank you for this opportunity and for reviewing my submission! This project pushed my ability to handle real-world computer vision challenges, and I’m excited to continue building on this foundation with your team.