

1. All Source Code:-

Make sure your GitHub repository includes:

- * `yash.py` → main script running detection and tracking.
- * `sort/sort.py` → SORT algorithm for real-time tracking
- * `players_yolov11.pt` → fine-tuned YOLOv11 model
- * `15sec_input_720p.mp4` → input video
- * `output_tracking.mp4` → generated video with ID overlays
- * `tracking_data.csv` → frame-wise data log.

Setup & Run Instructions:-

- * `git clone https://github.com/yaswanthikancharla99/soccer_game.git`
- * `cd soccer_game`
- * `pip install ultralytics opencv-python numpy filterpy scikit-image`
- * `python yash.py`

Dependencies Used:-

- * Python ≥ 3.8
- * `ultralytics` for YOLOv11
- * `OpenCV` for frame processing
- * `FilterPy`, `NumPy`, `scikit-image`.

3. Brief Report (PDF or Markdown)

❖❖ Approach and Methodology:-

- * Used YOLOv11 for fast player detection
- * Integrated SORT for re-identification and tracking stability
- * Built a frame-wise export log using CSV and OpenCV

❖❖ Techniques Tried:-

- * YOLOv11 fine-tuning.
- * ID mapping via Kalman filter in SORT
- * Frame annotation using OpenCV
- * Encountered file-size limits → resolved with Git LFS.

Challenges:-

- * Large .pt file blocked by GitHub' s upload cap
- * Push issues with Git LFS — resolved via migration and buffering.