Player Re-Identification in a Single Feed

Name: *Chidrawar Bhumika*
Date: *23-06-2025*
Assignment: Player Re-ID – Option 2 (Single-Feed)

---

1. Objective

Track players in a 15-second video (`15sec_input_720p.mp4`), assigning consistent IDs even if they leave and re-enter the frame.

---

2. Dataset & Model

- **Input video**: `15sec_input_720p.mp4` (provided by assignment — place it in the root directory)
- **Detection model**: YOLOv8-based `.pt` file (`best.pt` renamed and placed in `weights/`)
- **Tracking algorithm**: DeepSORT for frame-to-frame tracking and ID assignment

---

3. Methodology

3.1 Detection

- Utilized YOLOv8 via the `ultralytics` package
- Loaded `best.pt` model weights

3.2 Tracking & Re-ID

- Employed DeepSORT with `max_age=15` to handle temporary occlusions/exits
- Converted YOLO outputs (bounding boxes + confidence) into DeepSORT-compatible detections

3.3 ID Maintenance

- DeepSORT retains the same ID after re-entry if re-entered within 15 frames
- Tracking visualized by overlaying bounding boxes and IDs on output video

---

4. Code Walkthrough

`detect_and_track.py`:

1. Load YOLOv8 model
2. Initialize DeepSORT
3. Open input video
4. For each frame:
    - Detect players
    - Send detections to tracker
    - Draw bounding boxes with ID labels

5. Save output video to `results/`

---

5. Challenges

- **Short video duration** (15 seconds) limits tracking continuity testing
- **Occlusions & overlap** can cause brief mis-assignment
- **Uniform player appearances** (e.g. similar jerseys) make visual distinction harder

---

6. Future Work

- Integrate appearance-based Re-ID embeddings (e.g., TorchReID) for stronger identity retention
- Use OCR to detect jersey numbers for absolute matching
- Extend to longer feeds or cross-camera scenarios

---

7. Results

- **Output video**: `results/output_video_with_ids.mp4` shows players labeled with consistent IDs
- **Observations**: DeepSORT maintains IDs if players re-enter quickly; occasional ID swaps occur if entry is delayed or occluded

---

8. References

- YOLOv8 (Ultralytics)
- DeepSORT Real-time – `deep_sort_realtime`
- Ultralytics documentation, DeepSORT implementation guides