# Re-Identification in a Single Feed - Project Report

Internship Assignment - Liat.ai

Role: AI Intern

Task: Re-identify players in a single video feed.

Submitted by: Anitha Reddy

## 1. Objective

To simulate real-time person (player) re-identification within a single video feed by assigning consistent identities to players who disappear and later re-enter the camera view.

## 2. Project Pipeline

### 2.1 Detection (YOLOv8)

- YOLOv8 for detecting players in each frame.
- Frames extracted from video and passed through the detector.
- Bounding boxes saved and used to crop player images.

### 2.2 Cropping

- Extracted individual player crops from each frame.
- Saved crops for feature extraction.

### 2.3 Feature Extraction (Torchreid)

- Used OSNet (pretrained) from Torchreid to extract embeddings.
- Features saved as player_features.npy.

### 2.4 Re-ID Matching

- Cosine similarity used to compare all features.

- Grouped similar player images with consistent IDs.

## 3. Tools & Technologies

YOLOv8: Person Detection

Torchreid: Feature Extraction (OSNet)

NumPy: Feature Matrix Processing

Scikit-learn: Cosine Similarity Matching

Python 3.10+: Core Scripting

## 4. Experiments & Observations

- Input Video: 15sec_input_720p.mp4
- Frames: ~375, Crops: ~150
- Embedding Dim: 512-d, Groups Formed: ~10-15

## 5. Key Learnings

- Appearance-based features are critical for identity persistence.
- Frame-by-frame analysis scales better than real-time tracking.

## 6. Challenges Faced

- Uniform similarity caused re-ID confusion.
- Lighting changes affected accuracy.
- Large feature matrices needed careful memory handling.

## 7. Future Improvements

- Add temporal smoothing and tracking.
- Use domain-specific re-ID models.
- Visualize IDs in video output.

## 8. Final Deliverables

- extract_players.py: Crop players from frames

- extract_features.py: Extract features using OSNet

- reid_matching.py: Group players by ID

- player_features.npy: Feature embeddings

- reid_groups/: Grouped player folders

- requirements.txt, README.md, report.md

## Acknowledgments

Thanks to Liat.ai for the opportunity.

Libraries: Ultralytics YOLOv8, Torchreid, Scikit-learn, NumPy, OpenCV