CROSS-CAMERA PLAYER MAPPING

1. Approach and Methodology:

This project aimed to match players shown in two videos captured from different camera angles (Tacticam and Broadcast).

- A deep learning model called YOLOv8 from the Ultralytics library was used. This model was applied to each frame to detect players by drawing bounding boxes around them.
- Once players were detected, the areas within the bounding boxes were cropped. These cropped images were then passed into another model called ResNet50. Rather than being used for classification, ResNet50 was employed to extract visual features which are unique vectors that describe each player's appearance. These vectors were used to represent each player numerically.
- After features were extracted from Tacticam and Broadcast videos, cosine distance was used to measure the similarity between feature vectors. A lower cosine distance indicated a higher visual similarity between players.
- Based on these distances, the Hungarian algorithm was applied. This algorithm determined the optimal one-to-one matches between players across the two videos while avoiding duplicate assignments.
- Selected video frames with detected player boxes and IDs were saved to validate the results visually. Final matched pairs were stored in a .csv file.

2. Techniques used and their outcomes:

- YOLOv8 was found to work reliably in most frames. In well-lit and clear conditions, accurate detections were achieved. However, some players were either missed or incorrectly detected in blurry, crowded, or low-quality frames.
- ResNet50 was effective in extracting distinguishable features for many players. Nevertheless, players wearing identical uniforms with similar poses or appearing in low-quality frames often produced nearly identical feature vectors, making the matching process more difficult.
- Cosine distance proved helpful in comparing visual features by quantifying similarity in appearance. However, its performance was limited when players appeared highly similar.
- The Hungarian algorithm performed well when applied to small groups of players and clean data. It ensured one-to-one matching without duplication.
- OpenCV was used effectively for the visual validation of player detection and matching.

3. Challenges:

- No labelled player identities or tracking IDs were available in the videos, which made it challenging to assess matching accuracy or implement data-driven improvements automatically.
- Large model and video files (over 25MB) could not be uploaded directly to GitHub due to size restrictions.
- Bounding boxes were sometimes found too tight or loose, resulting in cropped images that either excluded parts of the player or included unnecessary background.
- Extracting ResNet features for every frame and player crop was computationally intensive and time-consuming.

4. Possible Improvements:

- A labelled dataset could be used to evaluate the accuracy of player matching quantitatively.
- A small application or user interface could be developed to facilitate visual inspection and verification of matched players across videos.