MatchVision Project Report

Approach and Methodology

The objective of this project was to identify and match the same players appearing in two different video sources: a **tacticam** (close-up view) and a **broadcast** (wide-angle view). We followed a step-by-step pipeline:

1. **Player Detection using YOLOv5**
   We used a pre-trained YOLOv5 model (best.pt) to detect players in both videos. The output was a set of cropped player images from each frame.
2. **Feature Extraction**
   From these cropped images, we extracted visual embeddings using a deep CNN. These embeddings represent players in a feature space.
3. **Player Matching**
   We compared the embeddings from both videos using cosine similarity to find the closest matches.
4. **Annotation and Evaluation**
   We visualized the matches by overlaying unique IDs on the tacticam video. To measure performance, we generated a dummy ground truth CSV and calculated accuracy using scikit-learn metrics.

Techniques Tried and Their Outcomes

- **YOLOv5 for Detection**:
  Player detection worked well under good lighting and visibility. However, it missed detections when players were occluded or far from the camera.
- **Cosine Similarity for Matching**:
  Simple and fast method for comparing features, but didn't work well in challenging scenes. Accuracy was very low (almost 0%) when using auto-generated ground truth — likely due to incorrect or noisy matches.
- **Streamlit Frontend (planned)**:
  Though not fully implemented, the idea was to use a Streamlit app to visualize matches interactively. This would help with manual corrections and debugging.

Challenges Faced

- **Large File Sizes**:
  .mp4, .pt, and .npy files were too large to push to GitHub, leading to 408 errors. We had to re-clone the repo and properly use .gitignore.
- **Data Issues**:
  The auto-generated ground truth was not reliable. It treated predicted matches as true labels, leading to inaccurate accuracy scores.

- **Model Limitations**:
  The embedding model wasn't trained specifically for re-identification in sports, so mismatches were common.
- **Workflow Glitches**:
  Several manual fixes were needed (e.g., folder creation, file placement, virtual environment issues).

What's Left / Future Work

- **Ground Truth Curation**:
  Manually label correct matches to evaluate the model honestly.
- **Model Improvements**:
  Use or fine-tune a re-ID-specific architecture like a Siamese or Triplet network on player images.
- **Better Visualization**:
  Upgrade video annotation to make matches more understandable. Maybe allow a human to review and approve them.
- **Streamlit Interface**:
  Add a simple UI for uploading videos, viewing results, and correcting mismatches.
- **Performance & Scale**:
  Optimize the pipeline to handle longer videos with many players.