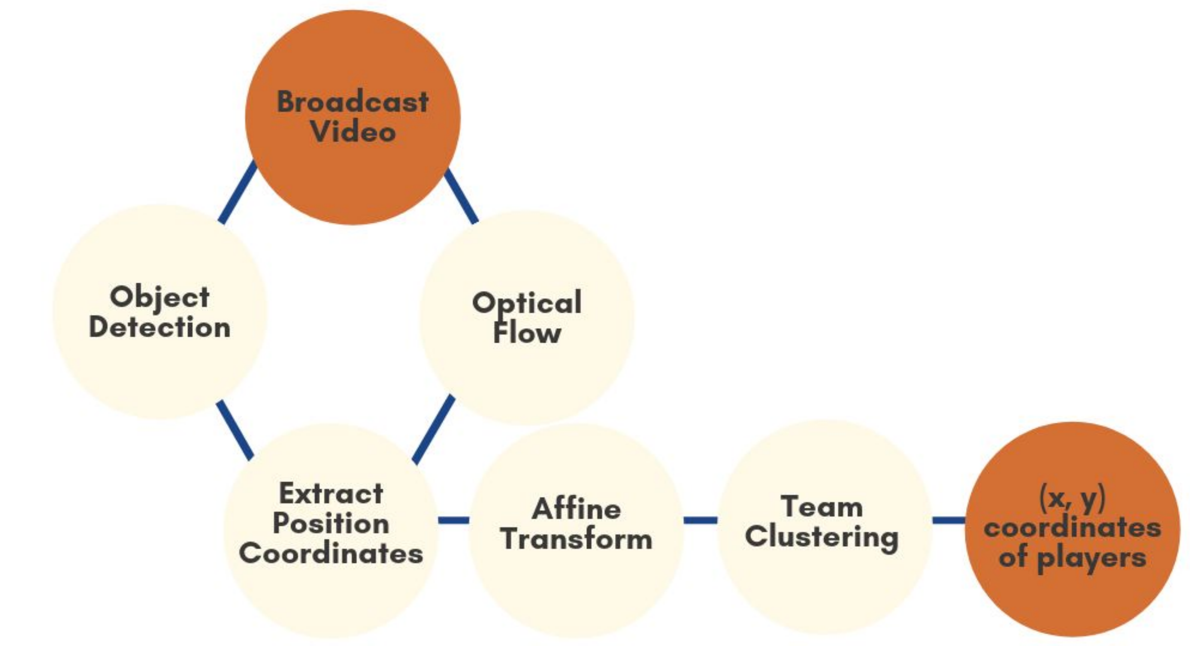
Basketball Data Extraction from Broadcast-Angle Video Footage

Alexander Bendeck and Niyaz Nurbhasha, Duke University

Introduction

This project is one of four in a "Basketball Analytics Pipeline" which seeks to study and consolidate varied approaches in basketball data analysis. Other teams were provided player and ball movement data from a company called SportVU and asked to analyze offensive and defensive schemes. Our goal was to extract this desired data from freely available broadcast-angle video footage.

To extract data, we focused on two concepts: Keypoint Detection and Instance Segmentation. We implemented Keypoint Detection using an API called OpenPose, a pre-trained neural network model which recognizes key points on the human body. We implemented Instance Segmentation via Mask R-CNN, a more generalized neural network for recognizing multiple classes of objects. While OpenPose worked well on the GPU (Graphics Processing Unit) computing nodes of the Duke Compute Cluster (DCC), training our Mask R-CNN network was more problematic. We were ultimately unable to train our model through the DCC, so we switched to a GPU environment provided by Amazon Web Services and trained the neural network through an online Computer Vision engine called Supervisely.



Object Detection

Players

- Mask R-CNN
  - Used weights file pre-trained on COCO dataset
    - 85 total objects
  - Mostly able to recognize people on the court
    - Players and referees
    - Also detects fans off the court
  - Sometimes struggles with overlapped players
- OpenPose
  - Detects 25 keypoints on the human body
  - Comparison: Less sensitive than Mask R-CNN
    - Still picks up noise (fans), though less prone to this than Mask R-CNN
    - Occasionally fails to detect some players

OpenPose player detection

Mask R-CNN player detection

Objects

- Trained network on Referees, Ball, Court, and Painted Area (Paint)
  - Split images into 80% training set and 20% validation set
  - Minimum detection confidence of 0.8 for object to be considered
  - Unclear images not considered
    - 24 from basketball, 12 from referees (out of 100 total)

Model	Precision	Recall	F1 Score
5-epochs-referees	49.9%	68.9%	57.9%
10-epochs-referees	50.2%	63.9%	56.2%
5-epochs-ball	13.1%	12.5%	12.8%
10-epochs-ball	16.8%	31.5%	21.9%

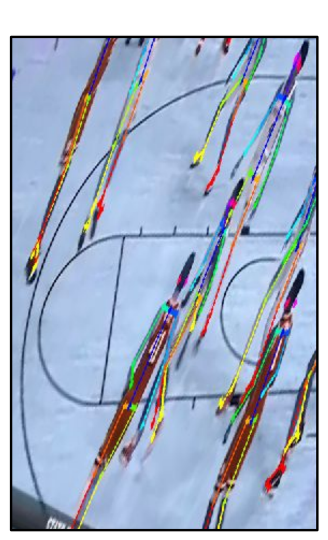
Mask R-CNN Basketball Detection

Mask R-CNN referee detection

Data Cleaning & Player Clustering

Affine Transform

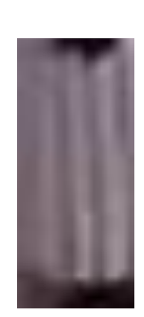
- Affine Transformation: Linear mapping between images preserving parallel lines
  - Used to map a player's location in an image frame to an absolute court location
  - Easy to then clean noise (fans, etc.) from data with sideline and baseline cutoffs



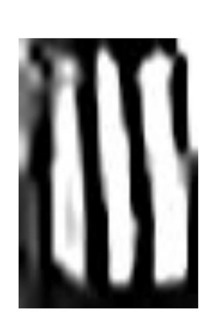
Above: Pre-transformation
Right: Post-transformation

Referee Detection

- We wanted an alternate method for detecting referees besides Mask R-CNN
  - Sharpen images with custom filters and detect vertical lines using a Hough Line Transform
- Results: On small dataset (30 players, 7 referees), all images classified correctly as referee or player



Raw image of referee

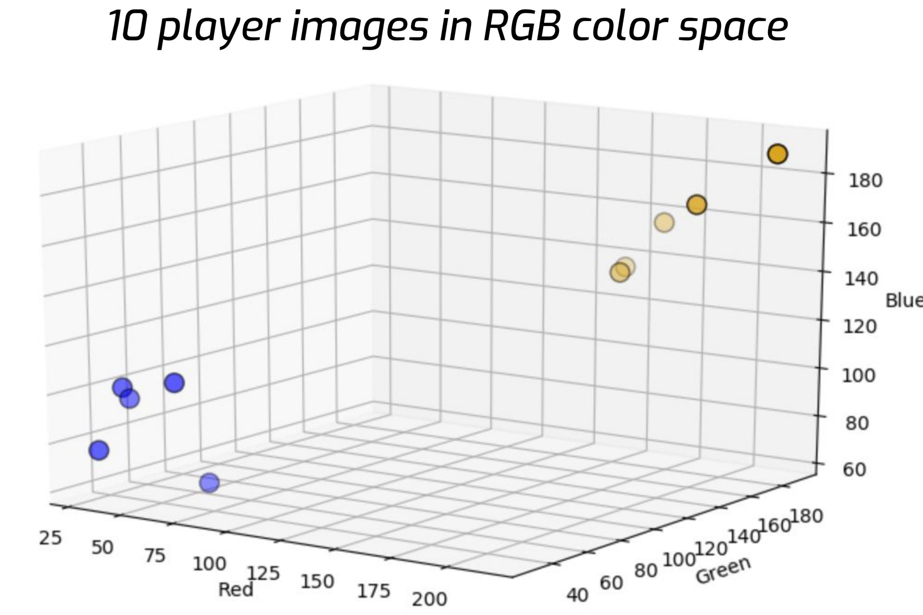


Sharpened image

Team Clustering

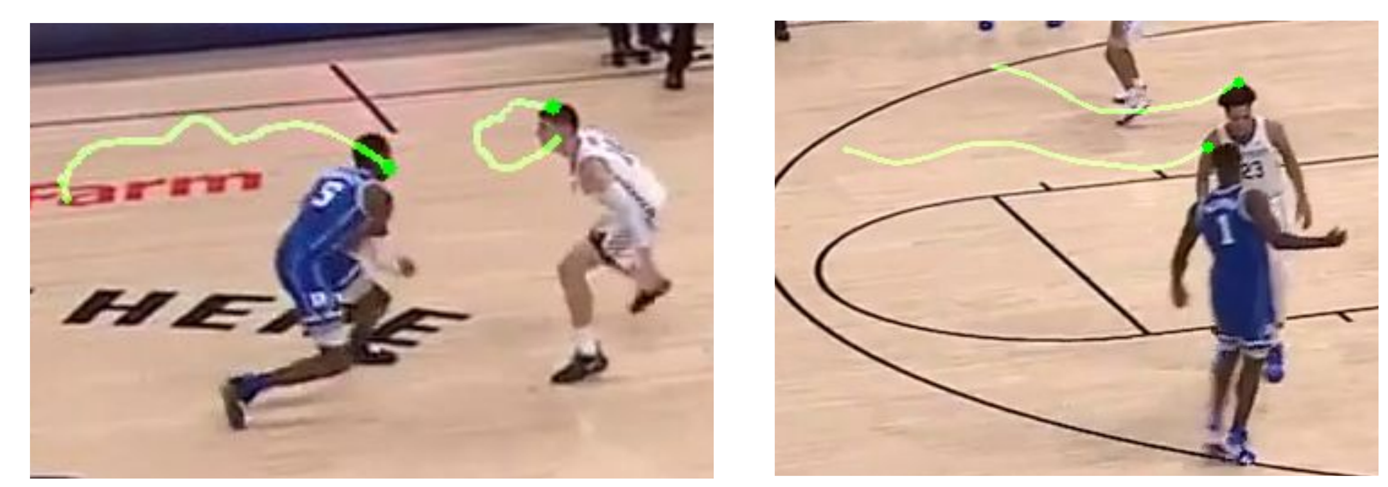
- Algorithm classifies 29 of 30 players correctly
  - K-Means clustering (k = 3) to identify top pixel colors within each image
  - K-Means clustering again (k = 2) to group by most prevalent color vector

10 player images in RGB color space



Optical Flow

- Goal: "follow" players between frames
  - Would only have to identify or label each person once
- Algorithm: Given a keypoint location in one frame, predict that the keypoint in the next frame that is closest to the current one is the same person
  - Very accurate for short intervals (< 25 frames)
- OpenCV algorithm follows points given by OpenPose across frames
  - Accurate for a few frames (< 10)
  - Not viable for long stretches of video
    - Subject to error when players criss-cross each other



Output of OpenCV Optical Flow algorithm

Future Work

- Recognizing basketballs is challenging due to poor resolution of ball
  - Idea 1: Use keypoint detection to identify players doing specific actions (i.e. shooting the ball) to estimate where ball is located
  - Idea 2: Track basketball with optical flow techniques
    - Need video with more frames per second (ball travels quickly)
- Refine team clustering and referee detection algorithms
  - Recent success training Mask R-CNN to detect the paint will allow us to automate the Affine Transformation
- Expanding our neural network training dataset
  - Now: One Duke vs. UNC game; need to improve the model's ability to identify objects in a variety of situations.
  - Access to better-quality video will translate to better models

References and Acknowledgements

1. Cao, Zhe, et al. "Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields." *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, doi:10.1109/cvpr.2017.143.
2. He, Kaiming, et al. "Mask R-CNN." *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, doi:10.1109/iccv.2017.322.

Special thanks to faculty mentor Alexander Volfovsky and project manager Harsh Parikh, both from Duke, for their support on this project.