Tennis Game Analysis: Simultaneous Detection of Players, Balls, and Court Geometry Using Deep Learning

Aneri Soni
University of Alabama at Birmingham
asoni2@uab.edu

Anusha Nandy
University of Alabama at Birmingham
nandya@uab.edu

Abstract

We developed a computer vision system to detect the tennis court, players, and ball in fixed angle broadcast videos. Our goal is to eliminate the need for multiple high-speed cameras and expensive equipment while still accurately tracking the key elements of the game. Using tools such as YOLO and TrackNet, our system analyzes ball trajectories, court positioning, and player locations, enabling applications such as coaching insights, match analysis, highlight detection, and fan-friendly visualizations. Our system achieves reliable detection of players and the ball, demonstrating that single-camera analytics is feasible and effective. Although our implementation focuses mainly on detection, it is the most crucial component for reliable analytics since it serves as a foundation that can be extended to a more comprehensive system for real-time tennis analysis.

1 Introduction

Analyzing sports with computer vision can unlock a wide range of insights for coaching, semi-automated refereeing, performance prediction, and engaging visualizations. Several commercial systems already exist for this purpose, Hawk-Eye Innovations being one of the most well-known examples, but they rely on multiple high-speed cameras positioned around the court. While effective, these setups are costly and difficult to deploy outside professional environments. Other solutions, such as ReSpoVision ReSpo.Vision and SwingVision SwingVision, aim to make analytics more accessible, but they still require specialized hardware or controlled conditions.

Our goal is to build the foundation for similar analytics for Tennis without the need for expensive multi-camera systems. Instead, we focus on designing an accessible, single-camera pipeline that can reliably detect the key components of a tennis match: the ball, the players, and the court lines.

Developing such a system introduces several challenges. Court lines may be faint or partially occluded by players or other objects. The players themselves move unpredictably and at high speed, often covering or crossing important court regions. The tennis ball is particularly difficult to track—it is tiny, extremely fast, frequently motion-blurred, and occasionally disappears entirely between frames. Previous research has explored these problems individually: some works track fast-moving balls using heatmap-based methods, while others estimate court geometry by detecting key line intersections using CNN-based models.

In this project, we bring these ideas together into one unified system. We train a one-stage YOLO detector to identify 14 key points on the court, players, and the tennis ball, and we also further implement TrackNet Chen et al. (2020) for ball detections across frames. Our objective is to achieve near real-time performance while remaining robust on different court surfaces, lighting conditions, and camera perspectives. Although our current focus is on detection, this serves as a crucial first step toward a more complete analytics system capable of supporting more advanced applications.

2 Related Work

A variety of computer vision techniques already exist to analyze tennis matches, particularly concerning small object detection for the balls. There's a lot of ideas proposed addressing different components of the problem.

One of the most influential approaches for tennis ball tracking is TrackNet Chen et al. (2020). Unlike traditional detectors that try to predict exact coordinates or bounding boxes, TrackNet treats ball localization as a heatmap regression problem. Instead of asking the model to pinpoint a single pixel, it predicts a smooth Gaussian heatmap that shows where the ball is most likely to be. This makes the model much more forgiving when the ball is blurred, extremely small, or almost invisible—situations that happen constantly in real tennis footage.

Another thoughtful design choice in TrackNet is its use of three consecutive frames rather than a single still image. This gives the model a short window of motion to learn from, allowing it to recognize how the ball moves, how it blurs, and where it tends to reappear after momentary occlusions. At the core of TrackNet is a VGG-style encoder–decoder architecture. The encoder is derived from the VGG network, which is known for its simple yet powerful stack of 3×3 convolutions and deep feature extraction capability. These features make TrackNet a strong and reliable foundation for small-object tracking in tennis, particularly when working with broadcast videos where the ball covers only a few pixels.

YOLO Ge et al. (2024) is a widely adopted one-stage object detector and has become a standard choice for real-time sports analytics due to its efficiency and ability to predict bounding boxes, objectness scores, and class probabilities in a single forward pass. Its frame-wise design enables processing at high frame rates, making it well suited for fast-paced environments such as tennis. Recent YOLO variants have substantially improved accuracy through architectural enhancements.

Detecting the tennis court layout is crucial for meaningful analytics such as ball trajectory projection, bounce detection, and player positioning. Prior research often detects court lines or keypoint intersections using convolutional neural networks, sometimes treating each landmark as a separate class in a multi-class detection model. After keypoints are extracted, homography estimation is typically performed to map the broadcast image to a canonical overhead court template.

3 Dataset

We used a custom tennis dataset of about 16,000 images from Roboflow Roboflow Universe. The dataset covers a wide range of court types, including hard, clay, and grass courts, and includes many different match conditions. Each image is labeled with 16 object classes: 14 court keypoints, one tennis ball, and one player. The court keypoints correspond to specific line intersections, such as baseline corners and service-line intersections, and each is annotated with a small bounding box and a unique class ID from 1 to 14. The dataset also contains bounding boxes for the tennis ball and for players. Roboflow provides a dataset split of 15,818 training images, 1952 validation images, and 139 test images, with most frames containing one tennis ball and up to two players. In addition to the labeled image dataset, we also used a separate collection of ten inference videos. These videos were not part of the training process; instead, they served as an additional evaluation set to observe how the model performs on real match footage.

4 Methodology

We used the YOLO11s model because it offers strong accuracy while remaining fast enough for real-time tennis analysis. Our model was trained to detect all 16 classes - 14 court keypoints, one player, and one tennis ball in a single forward pass. We initialized it with pre-trained weights (COCO) and then fine-tuned it for our tennis dataset. Player detection was handled as a standard object class and it performed well due to the larger bounding boxes and distinct visual appearance. Court keypoint detection was also effective, since these points remain structurally consistent once the model learns the geometry of the tennis court.

However, detecting the tennis ball was considerably challenging because it is small, fast, and often blurred. To improve ball performance, we incorporated ideas from TrackNet, which predicts a

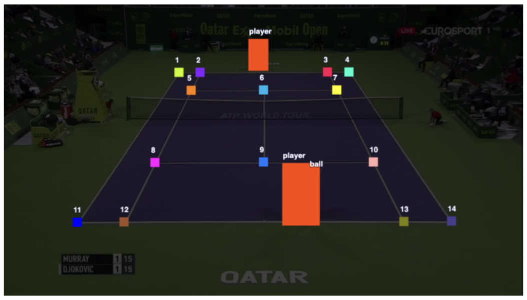

Figure 1: Sample annotated tennis frame -1

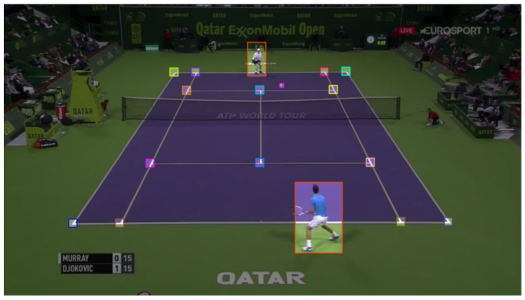

Figure 2: Sample annotated tennis frame -2

heatmap of the ball using multiple consecutive frames, which helps a lot with motion blur and partial occlusions. In our pipeline, we used a pre-trained TrackNet model to generate heatmaps on selected training video frames. These heatmaps helped verify the ball's true location and served as an additional supervisory signal during training and during inference was used alongside YOLO to make better decisions for difficult ball cases.

4.1 Training Process

Training was done for 50 epochs on an NVIDIA GPU (T4x2) using Kaggle. We used an input image resolution of 960 × 960 pixels to capture fine visual details such as line intersections and the small, fast-moving tennis ball. With the selected batch size, each epoch processed approximately 495 batches, covering around 8,000 images and allowing the full training set to be seen each epoch.

We applied standard YOLO augmentations to increase robustness, including horizontal flipping, scaling, and moderate color adjustments. These augmentations helped introduce natural variability without distorting the court geometry. The model was optimized using AdamW with YOLO's composite loss, which combines bounding box regression (IoU-based), object confidence loss and classification loss.

Key hyperparameters used during training were:

- **epochs:** 50
- **batch size:** 32
- **image size:** 960
- **optimizer:** AdamW (learning rate = 0.001)
- **device:** dual T4 GPUs (kaggle)
- **horizontal flip probability (fliplr):** 0.5
- **close-mosaic:** 10

5 Results

5.1 Detection Performance

All quantitative results are reported on a held-out validation set of 1,952 images that was not used during training. The final YOLO11s model achieved strong performance on the 1,952-image validation set. The overall mAP@50 reached 98.2%, while the stricter mAP@50-95 reached 88.3% across all classes, indicating both high detection accuracy and good localization quality.

5.1.1 Court Keypoints

All fourteen court keypoints were detected with near-perfect accuracy, with precision and recall values close to 0.99 and an average mAP@50 of approximately 99%. The mAP@50–95 values ranged from 80–90%, with keypoints closer to the camera detected most consistently. Keypoints near the net or partially occluded by players showed slightly lower scores. Overall, court keypoint detection remained stable across different lighting conditions, court surfaces, and camera angles.

5.1.2 Player Detection

Player detection was also highly reliable. The model achieved a mAP@50 of approximately 99.5% and a mAP@50-95 of roughly 93%. Qualitatively, players were consistently detected even during fast movements or partial occlusions, and the predicted bounding boxes aligned closely with their true positions in the frame.

5.1.3 Ball Detection

The tennis ball remained the most challenging object to detect. The model achieved a mAP@50 of 79.9% for the ball class, with precision near 0.81 and recall near 0.70. The lower mAP@50-95 value (approximately 39%) reflects the inherent difficulty of accurately localizing a small, fast-moving, and frequently motion-blurred object. Incorporating TrackNet-based temporal supervision improved robustness in difficult cases, although single-frame detection still resulted in occasional missed or imprecise predictions.

5.1.4 Overall System Behavior

The combined detection of court keypoints, players, and the ball remained robust across different court types, lighting conditions, and camera viewpoints. The system relied heavily on the geometric consistency of the fourteen court keypoints, enabling reliable reconstruction of the court even when some lines were faint or partially occluded by players.

5.2 Failure Case Analysis

Despite strong overall performance, several failure cases were observed. Ball detection errors most commonly occur during high-speed serves and smashes, where the ball appears severely motion-blurred or disappears entirely between frames. Occlusions caused by players, the net, or low contrast against the background can also lead to missed or imprecise ball predictions.

Court keypoint detection may degrade when lines are heavily worn, overexposed by lighting, or partially occluded by players; however, geometric consistency across the fourteen keypoints generally prevents significant reconstruction errors. Player detection failures are rare but can occur when players overlap substantially or move near court boundaries.

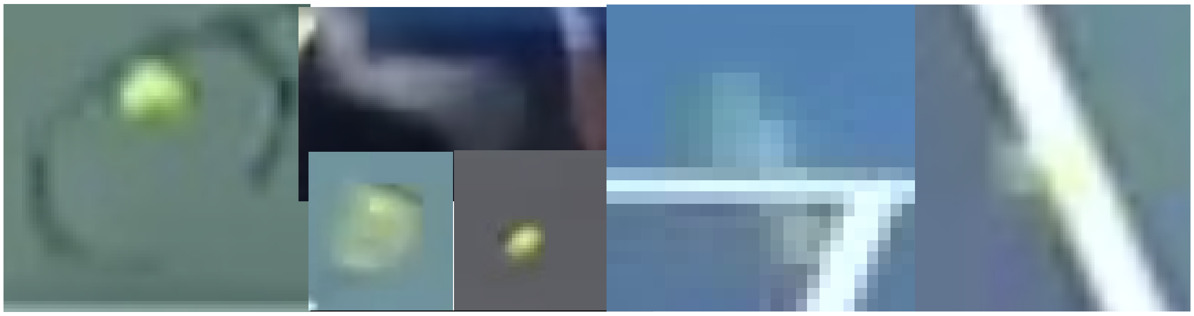

Figure 3: Representative failure cases for ball detection

Table 1: YOLO11s Detection Performance on the Validation Set

Class	**Precision**	**Recall**	**mAP@50**	**mAP@50-95**
All Classes	0.984	0.979	**0.982**	**0.883**
Keypoint 1	**0.998**	**0.998**	0.994	0.920
Keypoint 2	**0.999**	**0.999**	0.994	0.934
Keypoint 3	**0.999**	0.998	0.994	0.930
Keypoint 4	0.998	0.998	**0.995**	0.912
Keypoint 5	0.998	**0.999**	**0.995**	0.942
Keypoint 6	0.998	**0.999**	**0.995**	**0.956**
Keypoint 7	0.998	**0.999**	**0.995**	0.928
Keypoint 8	0.998	0.998	**0.995**	0.922
Keypoint 9	0.997	0.998	**0.995**	0.930
Keypoint 10	0.997	**0.999**	0.994	0.922
Keypoint 11	0.996	0.995	0.994	0.871
Keypoint 12	0.996	0.996	0.995	0.901
Keypoint 13	0.994	0.996	0.995	0.879
Keypoint 14	0.996	0.993	0.994	0.854
Ball	0.809	0.700	0.799	0.394
Player	0.977	**0.996**	**0.994**	0.932

6 Conclusion

We built a tennis vision system that detects court lines, players, and the ball from single-camera broadcast footage using a YOLO-based model supported by TrackNet-style tracking. The model performs extremely well on court keypoints and player detection, and while ball detection is harder, simple tracking helps follow most rallies reliably. Our results show that even with just one standard camera, it's possible to produce Hawk-Eye-like insights such as ball trajectories, player positions, and shot depth.

Despite these challenges, the system demonstrates that reliable tennis analytics can be achieved using a single-camera setup under realistic conditions.

7 Limitations and Future Work

7.1 Limitations

While the proposed system demonstrates strong performance under standard broadcast conditions, several limitations remain. First, the approach assumes a relatively stable broadcast camera viewpoint; performance degrades for unusual camera angles, extreme zoom levels, or videos where the full court is not visible. Second, although temporal interpolation improves robustness, prolonged ball occlusions (such as when the ball is hidden behind a player) can still lead to tracking errors. Finally, the current system focuses on detection and tracking and does not attempt to classify shot types or model player-specific response times.

7.2 Future Work

These limitations suggest several promising directions for future research. Incorporating shot classification models could enable recognition of forehands, backhands, serves, and volleys, providing deeper insight into playing styles and match dynamics, and integrating full-body player pose estimation would allow holistic analysis of strokes and movement efficiency.

References

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:1907.03698*, 2020.

Yuyin Ge, Shuhua Xu, Mingyu Zhao, Chengliang Xu, Ying Chen, Yuandong Tian, and Kaiming He. Simmim 2: A simple model for large-scale masked image modeling. *arXiv preprint arXiv:2410.17725*, 2024.

Hawk-Eye Innovations. Official website. https://www.hawkeyeinnovations.com/. Accessed: 2025-12-10.

ReSpo.Vision. Official website. https://respo.vision/. Accessed: 2025-12-10.

Roboflow Universe. Tennis all dataset. https://universe.roboflow.com/tennis-wkg7w/tennis-all. Accessed: 2025-12-10.

SwingVision. Official website. https://swing.vision/. Accessed: 2025-12-10.