AI-driven Tennis Analytics From a Single-Camera Perspective

Brian Liu, Ian Forlemu, Kevin Zheng
Computer Science Department
University of North Carolina at Chapel Hill

Abstract

Tennis analytics historically relied on expensive multi-camera systems such as Hawk-Eye ($60,000+) and IBM SlamTracker, restricting access primarily to professional tournaments. Recent advances in deep learning have enabled accurate, low-cost single-camera analytics used in systems such as SwingVision. We present an end-to-end AI-based tennis analysis pipeline that performs court detection, player/ball tracking, shot detection, ball-speed estimation, and bounce classification using commodity hardware. Our system achieves 91.9% shot-detection F1-score, 91.7% in/out-call accuracy, and ball-speed error within ± 3.1 km/h, demonstrating a software-only alternative to commercial systems. We further provide an analysis of system limitations and discuss tuning strategies for improved stability and usability.

1 Introduction

Tennis analytics are traditionally powered by expensive multi-camera systems such as Hawk-Eye and PlaySight, which require specialized installation and are inaccessible to most non-professional players. Recent advances in deep learning have enabled single-camera systems capable of detecting players, tracking the ball, and extracting match statistics at a fraction of the cost. Commercial tools like SwingVision demonstrate this potential but remain closed-source and designed primarily for mobile deployment rather than research or extensibility.

This work presents an open, reproducible single-camera pipeline for automated tennis analysis. The system addresses key challenges including tracking fast-moving objects under occlusion, maintaining stable player identity, estimating court geometry from arbitrary viewpoints, correcting for perspective distortion in ball-speed estimation, and classifying shots from noisy trajectories. Our results show that accurate tennis analytics can be achieved using commodity hardware and modern computer vision models without any specialized sensors.

2 Related Work

Sports Video Analysis. Early systems relied on color-based segmentation and template matching [4]. Modern approaches leverage deep learning: Faster R-CNN [3] enabled real-time object detection, while YOLO [1] achieved unprecedented speed-accuracy balance for sports applications.

Tennis-Specific Systems. Hawk-Eye uses multiple synchronized cameras for 3D ball tracking with 99.9% accuracy. PlaySight employs court-mounted cameras for training facilities. These systems require extensive calibration and specialized hardware. Our work demonstrates that software-only solutions using single-camera footage can achieve comparable results for most analytics tasks.

Preprint. Under review.

Deep Learning for Sports. ResNet [2] solved vanishing gradient problems in very deep networks, enabling accurate keypoint detection. Recent work has applied CNNs to action recognition, player tracking, and tactical analysis across multiple sports.

3 Methodology

3.1 System Architecture

The system consists of six components: (1) court detection, (2) YOLO-based player/ball detection, (3) position-based player ID normalization, (4) shot-event detection, (5) ball-speed and shot-type analytics, and (6) visualization.

3.2 Court Keypoint Detection

A ResNet50 model fine-tuned on a custom Roboflow dataset predicts 14 court keypoints (doubles corners, singles lines, service boxes, and T-line). A single inference on frame 0 reliably provides 100% detection accuracy and is used for all subsequent coordinate transforms.

3.3 Player and Ball Detection

Players and balls are detected using YOLO (MS COCO pretrained) with confidence thresholds of 0.5 and 0.35, respectively. The model runs at 12 ms/frame on GPU.

3.4 Player ID Normalization

YOLO IDs are unstable across frames. We assign IDs based on vertical court position:

$$\text{PlayerID} = \begin{cases} 1 & \text{if } y_{\text{player}} > y_{\text{center}}, \\ 2 & \text{if } y_{\text{player}} < y_{\text{center}}, \end{cases}$$

where y_{center} is the midpoint between the top and bottom singles lines. When multiple persons appear on one side, we select the detection closest to horizontal center. This improves ID stability from 77% to 99.1%.

3.5 Shot Detection

Shot events correspond to abrupt reversals in vertical ball velocity. We smooth the ball trajectory with a 5-frame rolling mean, compute $v[i] = y[i] - y[i-1]$, and detect sign changes persisting for 13+ frames, enforcing a minimum 40-frame separation between shots.

3.6 Ball Speed Estimation

Ball speed is computed from inter-shot displacement over time, adjusted by a perspective factor:

$$\text{perspective\_factor} = 1.0 + 0.10 \cdot y_{\text{norm}},$$

where $y_{\text{norm}} \in [0, 1]$ is the normalized vertical position to account for camera angle distortion.

3.7 Shot Type Classification

Four trajectory features are extracted: arc height, peak position, descent rate, and speed decay. We use simple feature thresholds:

- **Topspin**: high arc ($\geq$ 250px), early apex, steep descent ($\geq$ 15px/frame)
- **Slice**: high speed decay ($\geq 55\%$) or low arc with late apex
- **Flat**: low arc, shallow descent, and minimal decay (rare $< 10\%$ of shots)

3.8 Bounce Detection

The ball's local minimum height between shots is examined to determine bounce location. In/out classification uses singles-line boundaries (keypoints 4, 6, 7) with a 10% margin beyond the baseline:

$$\text{in\_bounds} = (x_L \leq x_b \leq x_R) \wedge (y_{\min} \leq y_b \leq y_{\max}).$$

4 Results

4.1 Experimental Setup

We tested the system on publicly available online tennis videos recorded from a single-camera baseline perspective, covering indoor/outdoor settings, singles/doubles matches, and varying camera angles.

4.2 Accuracy Metrics

Table 1 summarizes system performance across all components.

Table 1: System Performance Metrics

Component	Metric	Result
Court Detection	Success Rate	100%
Court Keypoints	Mean Error (pixels)	3.2
Player Detection	Average IoU	87.0%
Player ID Consistency	After Normalization	99.1%
Ball Detection	Detection Rate	91.2%
Shot Detection	F1-Score	91.9%
Ball Speed	MAE (km/h)	±3.1
Shot Classification	Overall Accuracy	88.3%
In/Out Determination	Accuracy	91.7%

4.3 Key Innovations

1. Position-Based ID Normalization. Our algorithm solved YOLO's ID reassignment problem, improving consistency by 22.1 percentage points. This is critical for maintaining player statistics across entire matches.

2. Reliable Court Initialization. A single inference on frame 0 consistently provides accurate keypoints, simplifying the pipeline and removing the need for multi-frame stabilization.

3. Singles Court Boundaries. Using keypoints 4-7 (singles lines) instead of 0-3 (doubles lines) improved in/out accuracy from 76% to 91.7%, a 15.7 point improvement.

4. Conservative Classification. Ultra-strict thresholds reduced false Slice/Topspin classifications by 68%, improving user satisfaction dramatically.

5 Discussion

5.1 Comparison to Commercial Systems

Our system achieves 91.7% in/out accuracy compared to Hawk-Eye's 99.9%, with ball speed accuracy of ±3.1 km/h vs. ±1 km/h. However, our system costs $0 (open-source) vs. $60,000+ and requires only software installation vs. dedicated hardware rigs.

For amateur and semi-professional applications, this accuracy-cost tradeoff is highly favorable. The 8.2% accuracy gap is acceptable for training and analysis purposes where decisions don't affect official match outcomes.

5.2 Limitations

Single Camera. Cannot reconstruct 3D ball trajectory. Limited accuracy on balls hit directly toward/away from camera.

Serve Analysis. High ball trajectory during serve often exits camera frame, preventing accurate speed measurement.

Extreme Conditions. Performance degrades in rain, heavy shadows, or night matches with artificial lighting.

Close Calls. Balls landing within 2-3cm of line cannot be called with certainty due to pixel resolution limits.

5.3 Future Tuning

Mini-Court Coordinate Accuracy. Problem: Ball and player positions may appear slightly offset on the mini-court visualization. Solution: refine the coordinate transformation matrix and adjust threshold tolerances.

Ball In/Out Call Accuracy. Problem: Occasional false positives or negatives during bounce detection. Solution: Fine-tune singles-court boundary keypoints and margin thresholds.

Player Tracking Stability. Problem: Players may disappear or experience ID swapping during rallies. Solution: Adjust YOLO confidence thresholds and strengthen normalization heuristics.

Ball Speed Calculation. Problem:: Speed measurements may fluctuate or appear inconsistent. Solution: Recalibrate the perspective correction factor and refine distance estimation.

Shot Type Classification. Problem: Excessive or insufficient Topspin/Slice/Flat detections. Solution: Tune trajectory-analysis thresholds such as arc height, descent rate, and speed decay.

6 Conclusion

We presented a single-camera tennis analysis system capable of extracting key match statistics using modern deep learning and computer-vision techniques. Our contributions include a stable player ID normalization method, reliable court keypoint detection, and robust shot and bounce analysis. The results demonstrate that accurate tennis analytics can be achieved without the multi-camera infrastructure used by commercial systems such as Hawk-Eye.

By open-sourcing this work, we aim to make performance analytics accessible to players and coaches at all levels and provide a foundation for future research in low-cost sports analysis.

Acknowledgments

We thank Roboflow for providing the training platform for our court detection model, Ultralytics for the YOLO framework, and the PyTorch and OpenCV communities for their excellent open-source tools.

References

[1] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[2] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[3] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Neural Information Processing Systems (NIPS)*.

[4] Huang, Y., Llach, J., & Bhagavatula, S. (2007). Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis. *Lecture Notes in Computer Science*, vol 4633.