CPT_S 570

Machine Learning

Tennis Analysis System

[Github repository]

Submitted by

Priyadharshini Damodharan (011859827)

Sai Thanmai Polamreddy (011868087)

Under the Guidance of

Dr. Janardhan Rao (Jana) Doppa

WASHINGTON STATE UNIVERSITY

Contents

Abstract

Accurately analyzing tennis player performance presents significant challenges due to fast-paced nature of game and complexity of player movements. This project addresses these challenges by developing a system that utilizes advanced computer vision and machine learning techniques to track and analyze player and ball movements from video data. The system employs YOLO models for precise detection of players and fast-moving tennis ball and uses object tracking with Ultralytics library to maintain consistent player IDs across frames. Video data is preprocessed to calculate ball trajectories, player speeds, and shot statistics through pixel-to-metric conversions. The results are visualized by overlaying computed values onto video frames, providing intuitive spatial and temporal analysis of gameplay dynamics. To enhance robustness and reliability of these predictions, advanced uncertainty quantification tools such as ensembles and conformal prediction were incorporated. Additionally, explainable ML tools, including SHAP (SHapley Additive exPlanations), were implemented to elucidate impact of key features on model's predictions. This comprehensive analysis enables detailed insights into real-time player responses, strategic movements, and shot performance, offering valuable tools for coaching, performance improvement, and sports analytics.

1 Introduction

The game of tennis, characterized by its rapid pace, dynamic rallies, and complex player movements, presents significant challenges for quantitatively analyzing player performance. Traditional methods of performance analysis often rely on manual observation and annotation by coaches and analysts. However, these methods are time-consuming, labor-intensive, and susceptible to human bias and error. This underscores need for automated, objective, and precise alternatives to analyze and interpret player behavior, movement patterns, and overall performance during gameplay.

Motivation

Athletic performance has changed throughout time due to technological advancements. Modern tennis rackets made of composite materials and shoes that are lighter, more flexible, and more durable are examples of equipment innovations that assist tennis players today. The combination of machine learning and computer vision for performance analysis, however, represents most recent and potentially revolutionary developments.

Despite groundbreaking systems like **Hawkeye** by Sony—used by referees for ball tracking and decision-making—these technologies remain expensive, cumbersome, and inaccessible for recreational players and coaches with limited resources.By utilizing machine learning and computer vision, we aim to bridge this technology divide by developing accessible, affordable tools that can evaluate player performance with just a smartphone camera.

The goal of this study is to democratize sports analysis by providing cutting-edge insights to athletes of all skill levels, not only top competitors. We want to enable ath-

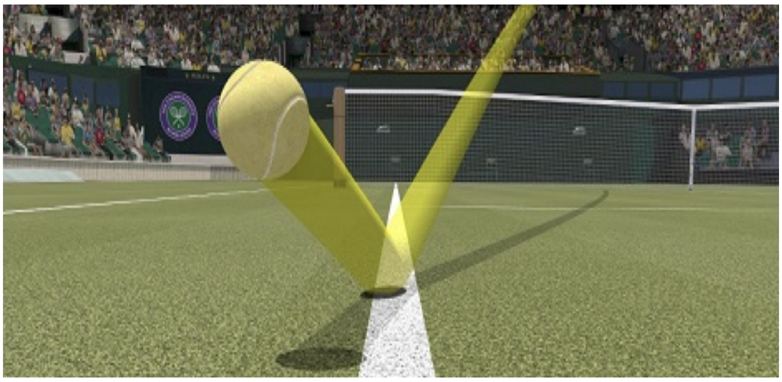


Figure 1: Hawk-Eye, a technology with a setup cost of $60,000 or more per court [16] reconstructs shots in three-dimensional space.

letes and coaches to better understand and maximize performance through data-driven feedback by making information like shot speed, response times, and player movement patterns available to both professional players and an amateur enthusiasts.

Objectives

To address challenges outlined above, this paper proposes development of an innovative, low-cost system that utilizes machine learning and computer vision to analyze player and ball movements from simple video footage. The system's primary objectives include:

- Exploring capabilities of machine learning and computer vision in extracting quantitative performance metrics using common smartphone footage.
- Utilizing YOLO (You Only Look Once) for real-time, accurate detection of both players and fast-moving tennis ball, allowing dynamic tracking during gameplay.
- Applying Ultralytics object tracking to ensure consistent player identity across video frames and track movement patterns over time.
- Preprocessing video data to compute important performance metrics such as ball trajectories, player speed, shot statistics, and agility through pixel-to-metric conversions.
- Visualizing extracted performance insights by overlaying statistics directly onto video, facilitating intuitive spatial and temporal analysis of gameplay.
- Offering statistical insights such as average shot speeds, player reaction times, agility measurements, and shot counts to support training, coaching strategies, and decision-making.
- Building uncertainty quantification tools on top of these predictions to enhance overall decision-making. Examples include ensembles (mean and variance), conformal prediction, and adding a Gaussian process on top of a given representation.
- Adding explainable/interpretable ML tools to create explanations for predictions and illustrate how they can be used for decision-making.

Significance

The overarching goal of this research is to create a proof-of-concept for an affordable, smartphone-based system that can:

- Allow professional and recreational athletes to analyze performance without needing specialized equipment.
- Provide coaches with data-driven insights for training and strategy development.
- Enable strategic analysis of player performance during training or competitive matches in real-time.

The combination of machine learning and computer vision in this system signifies a revolutionary change in sports analysis. By using objective data, it not only improves coaching insights but also lowers cost of obtaining sophisticated sports analytics tools. The ultimate goal of this research is to demonstrate that, without need for expensive systems, performance analysis in tennis—and possibly other sports—can become widely available and reasonably priced.

2 Literature Review

2.1 Advancements in Computer Vision and Machine Learning in Sports

Significant progress has been made in last ten years in use of computer vision and machine learning in sports analysis. The automation of performance analysis made possible by these technologies has resulted in more precise and up-to-date insights into a variety of sports. The potential of these technologies to detect player movements, analyze game strategies, and improve athlete performance has been shown in notable research [1, 2].

2.2 Neural Networks and Convolutional Neural Networks (CNNs)

Artificial neural networks, inspired by biological neural networks, have become dominant frameworks in field of machine learning. By passing input vectors through layers of neurons, applying activation functions, and generating outputs that indicate significance of different attributes, these networks carry out inference. Architectures like AlexNet, VGGNet, and MobileNet are important advancements; they each have distinct functions or enhance earlier models [5–7].

By applying convolution operations to input data, Convolutional Neural Networks (CNNs) preserve spatial correlations and use weight sharing to reduce number of parameters. Because CNNs can record spatial hierarchies and local dependencies, they are especially useful in image and video processing. CNNs are used in this project's networks to process images and videos efficiently, improving object tracking and human posture estimate accuracy [8, 9].

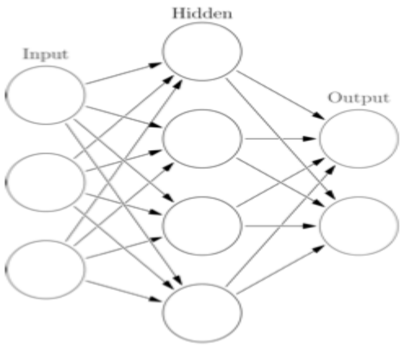


Figure 2: Neural Network

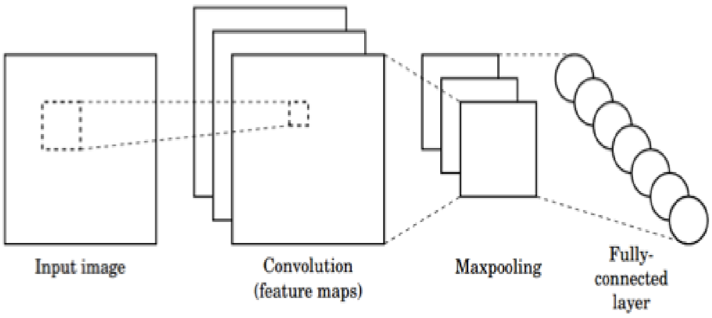


Figure 3: Convolutional Neural Network

2.3 Use of YOLO Models for Object Detection

Object detection is essential for locating objects within images, making it ideal for detecting flying tennis balls during a tennis game. Instead of processing thousands of subregions, sophisticated object detection network YOLO (You Only Look Once) applies predictions to entire image based on global context. This method guarantees accuracy and quickness, which are essential in fast-paced sporting settings. Convolutional layers and a sequence of residue blocks make up YOLO's design, which allows for deeper networks via detours. Research has demonstrated that YOLO outperforms other object identification techniques in terms of speed and accuracy, which makes it especially appropriate for real-time sports analysis [10–12].

2.4 Application of Machine Learning in Tennis

The application of machine learning to performance analysis in tennis has been subject of numerous studies. Morawski et al. [17] classified tennis strokes from sensor data using machine learning algorithms, offering important information on player performance and technique. Vučković et al. [18] used machine learning to examine shot patterns in professional tennis players, showing how these technologies can reveal complex facets of player strategy and behavior.

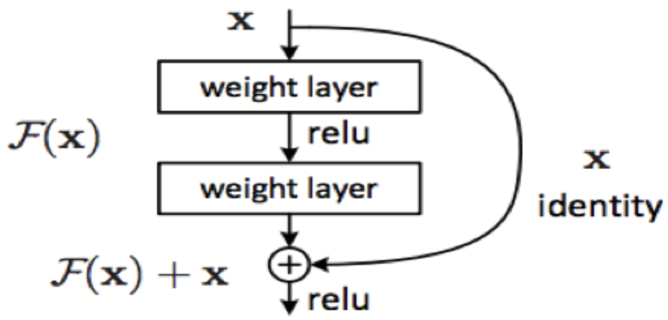


Figure 4: Residual Block in a Neural Network

2.5 Uncertainty Quantification in Machine Learning

In machine learning, uncertainty quantification (UQ) entails assessing forecast accuracy and measuring the degree of confidence in model predictions using metrics like confidence intervals. By taking reliability into account, UQ tools such as Gaussian Processes (GPs), ensemble approaches, and conformal prediction can be incorporated into models to enhance decision-making. To lessen bias and variation, ensemble techniques like bagging and boosting combine predictions from several models. GPs provide a probabilistic representation of predictions, including mean and variance, whereas conformal prediction builds adaptive prediction intervals based on historical errors. These methods improve model predictions' reliability and robustness in real-world applications [19–21].

2.6 Explainable and Interpretable Machine Learning

Machine learning technologies that are interpretable and explainable are essential for comprehending model predictions and guaranteeing decision-making transparency. Techniques like SHAP (SHapley Additive exPlanations) help to improve interpretation and confidence in model's outputs by revealing how each feature contributes to model's predictions [22, 23]. Based on cooperative game theory, SHAP values offer a means of equitably allocating the "payout" (model prediction) among the "players" (input features). To ensure consistency and local accuracy, SHAP values show how each feature contributes to the prediction. Visual tools for interpreting and explaining model predictions include force charts, dependence plots, and SHAP summary plots [22]. To explain individual predictions, LIME uses an interpretable model to locally approximate the complicated model. LIME fits a straightforward model (similar to a linear regression) that is interpretable and can show how each feature contributes to the prediction by altering the input data and tracking changes in the predictions. Any black-box model can be used with this model-agnostic approach [23].

3 Methodology

3.1 Data Collection

Sources of Video Footage

In order to build system, a significant amount of tennis match video footage had to be gathered. Training and testing models used for player and ball detection and court key point extraction are based on video data. The following sources of video footage were used to guarantee diversity and resilience in model's performance:

- **Publicly Available Match Recordings**: Professional matches that are accessible on websites like YouTube and other sports streaming services are among them. The footage offers a variety of scenarios and environments by spanning several tournaments, match circumstances, and player levels.
- **Specific Practice Sessions**: To record certain player motions, ball trajectories, and shot kinds, practice sessions were designed and recorded. The models' ability to handle particular tennis moves, such serves, volleys, and smashes, was ensured by development of specialized training data in this controlled setting.

This combination of professional match footage and practice session recordings provided a comprehensive dataset that covers various aspects of tennis gameplay. The diversity in data ensures models are robust and perform well under different conditions.

Tennis Ball Detection Dataset

A dedicated dataset for tennis ball detection was sourced from Roboflow, which offers a well-curated collection of annotated images specifically designed for detecting tennis balls. The dataset, available at Tennis Ball Detection Dataset, includes images with various states of tennis ball, such as during rallies, serves, and smashes. The dataset features:

- **Annotations**: Each image in dataset is annotated with precise locations of tennis ball. These annotations are crucial for training YOLO model to detect and track tennis ball accurately.
- **Variety in Scenarios**: The dataset includes images from different match scenarios, ensuring model can detect tennis ball under various conditions, such as different lighting, angles, and player movements.
- **High-Resolution Images**: High-quality images are used to ensure model can learn fine details necessary for accurate detection. This helps model generalize well to real-world conditions, making it effective for live match analysis.

The project makes use of this particular dataset to guarantee that tennis ball detection model is trained on a variety of high-quality data, resulting in reliable performance in a range of match scenarios.

3.2 Software Requirements

- Python 3.8
- Ultralytics
- PyTorch
- Pandas
- NumPy
- OpenCV

3.3 Models Used

The system employs a combination of advanced machine learning models and techniques to achieve accurate player and ball detection, as well as court key point extraction. Below are primary models utilized in this study:

3.3.1 YOLO v8 for Player Detection

For player recognition, YOLO (You Only Look Once) v8 model was chosen because of its excellent accuracy, real-time performance, and effective handling of dynamic video frames. Using convolutional neural networks (CNNs), YOLO v8 is a cutting-edge object detection model that can identify and locate items in real time. By examining tennis players' whereabouts on court in every frame, it was able to recognize and follow them throughout film. YOLO v8's real-time detection feature makes it possible to continuously track player movement throughout a game.

3.3.2 Fine-tuned YOLO for Tennis Ball Detection

In order to detect fast-moving tennis ball in a variety of ambient situations, YOLO model was specifically refined. Training model on a specific dataset centered on tennis ball trajectories and behavior, including during rallies, serves, and smashes, was necessary for this fine-tuning. The tennis ball's speed, motion, and lighting variations added complexity that necessitated a customized strategy to increase detection precision, necessitating fine-tuning. This customized YOLO model guarantees that system can track ball consistently in a variety of game situations.

3.3.3 Court Key Point Extraction with CNNs

On tennis court, Convolutional Neural Networks (CNNs) were used to extract important spots for spatial analysis. The use of CNNs in this system enabled precise identification of important landmarks on court (such as service boxes, baseline placements, and other spatial elements), as they are very successful at image recognition and spatial mapping tasks. The system can map player motions and ball trajectories inside court's layout thanks to spatial context provided by extracted court key points. Analyzing trends, movement tactics, and gun trajectories requires this spatial mapping.

To guarantee balanced training and effective learning, for example, a batch size of 16 and a learning rate of 0.001 might be chosen. The model receives processed video frames from gathered tennis match material. The frames are enlarged to 416x416 pixel input dimensions that YOLO v8 requires.

4 Implementation of Player Detection Using YOLO v8

4.1 Setting up YOLO v8

Framework Initialization: The YOLO v8 is configured using Ultralytics library. Tools for simple deployment and integration of YOLO models are provided by this package. To initialize model, pre-trained YOLO v8 weights are loaded. This provides a strong starting point, as pre-trained weights have been optimized on a large dataset for object detection tasks.

Model Configuration: Hyperparameters such as learning rate, batch size, and number of epochs are configured to fine-tune model.

For instance, a learning rate of 0.001 and a batch size of 16 might be set to ensure balanced training and efficient learning. Video frames from collected tennis match footage are processed and fed into model. The frames are resized to input dimensions expected by YOLO v8 (typically 416x416 pixels).

4.2 Detecting Players in Video

Frame Processing: To identify and follow players, YOLO v8 processes each video frame individually. The existence of players is examined in every frame. Bounding boxes surrounding identified players, class names, and confidence scores that represent probability of detection accuracy are all output by model.

Player Tracking: The model makes use of track method offered by Ultralytics library to guarantee consistent tracking of same player across several frames. This keeps players' identities consistent throughout video.

Post-Processing: Post-processing is done on identified bounding boxes to remove duplicates and fine-tune player locations. To ensure that just best detection is retained, overlapping boxes are filtered out using Non-Maximum Suppression (NMS).

5 Training Tennis Ball Detector with YOLO

5.1 Dataset Preparation

Collection: A diverse set of 578 images was sourced from Roboflow, featuring tennis balls in different states and environments (e.g., during rallies, serves, and smashes). The dataset includes 428 training images and additional testing images.

Roboflow's annotation tool is used to add bounding box annotations to each image, indicating exact location of tennis ball. The model is trained using these annotations as ground truth, guaranteeing precise ball spatial bounding in every image.

Preprocessing: Images are resized to input dimensions required by YOLO model (typically 416x416 pixels). This standardization ensures consistency in data fed to model.

5.2 Model Configuration

Framework Setup: The YOLO framework is set up using Ultralytics library, which ensures compatibility with system's hardware (e.g., GPU) and software environment.

Hyperparameters Tuning: Hyperparameters are tuned using a structured search process:

- Learning Rate: Initialized at 0.001 and adjusted using learning rate schedulers like cosine annealing.

- Batch Size: Set at 16 to balance GPU memory usage and training stability.

- Number of Epochs: Set at 50, with performance evaluated at intervals of 5 epochs. This optimization ensures efficient learning and reduces overfitting.

5.3 Training Model

Data Split: The annotated dataset is split into 70% training data and 30% validation data. This split allows model to learn from majority of data while being validated on a subset to monitor performance.

Training Process: - Training is performed using stochastic gradient descent (SGD) optimizer with a momentum of 0.9. This helps in accelerating training process while avoiding local minima.

- The YOLOv8 architecture learns weights iteratively by minimizing cross-entropy loss and localization loss through backpropagation. Each epoch adjusts weights based on error from previous iteration.

- The training involves 100 epochs, with performance evaluated after each epoch using metrics such as Mean Average Precision (mAP). Adjustments are made based on these evaluations to improve model's accuracy.

5.4 Evaluation and Fine-Tuning

Performance Monitoring: Metrics like Intersection over Union (IoU), mean Average Precision (mAP@50), and precision-recall curves following each epoch on validation dataset are used to track model's performance. This aids in locating any overfitting or underfitting problems.

Fine-Tuning: To increase accuracy and resilience, hyperparameters are changed and data augmentation methods including rotation, random cropping, and color jittering are used. These additions improve model's generalization by simulating many real-world scenarios.

5.5 Final Testing

Evaluation on Test Set: To make sure resulting model generalizes effectively to fresh, unseen data, it is tested on a different test set (unseen matches and ball movement scenarios) after training. A realistic assessment of model's performance in real-world situations is given by test set.

Performance: The tennis ball is consistently detected and tracked by model in a variety of game situations, yielding precise trajectory data that is necessary for in-depth evaluation of player performance and game tactics.

6 Training Court Key Points with PyTorch

6.1 Dataset Preparation

Annotated Images: A set of annotated images was prepared by marking key tennis court spatial landmarks, including baseline, service lines, and net line.

Manual annotations were applied to ensure high-quality training input. Preprocessing techniques like resizing (images standardized to 224x224 resolution) were applied to make them compatible with Convolutional Neural Networks (CNNs).

6.2 Model Design and Configuration

Model Architecture: A ResNet50 architecture was chosen for its robust performance in image recognition tasks. Transfer learning was leveraged by using pre-trained weights on a large image dataset.

The final fully connected layer of ResNet50 model was replaced to output 28 key points (14 points with X and Y coordinates).

Hyperparameter Tuning: Hyperparameters were set using a random search approach to optimize model performance:

- Learning Rate: Set at 0.001.

- Batch Size: Set at 32.

- Optimizer: Adam optimizer was used for adaptive gradient descent with a momentum of 0.9.

- Epochs: The model was trained for 100 epochs to ensure convergence and minimal overfitting.

6.3 Training Model

Loss Function: The model was trained using Mean Squared Error (MSE) loss for coordinate prediction. This loss function was suitable for regression task of predicting precise locations of key points.

Training Process: The annotated images were fed into CNN, and model adjusted its weights through backpropagation to minimize error in key point detection.

In order to simulate changes in camera angles and dynamic court views, data augmentation techniques including random rotation, zoom, and horizontal shifts were employed.

Key visual signals, such as baseline and net patterns, were mapped to their respective spatial reference points by CNN.

There were 100 epochs in training process, and measures like accuracy and loss were used to assess performance at end of each epoch. Based on these assessments, changes were made.

6.4 Evaluation and Fine-Tuning

Model Evaluation: Model accuracy was evaluated using a validation dataset, with performance measured against metrics like Root Mean Squared Error (RMSE) for prediction accuracy.

Regularization methods like dropout (rate 0.5) were implemented to mitigate overfitting.

Fine-Tuning: The evaluation's findings were used to modify hyperparameters. To improve resilience of model, regularization strategies including dropout and L2 regularization were used.

To increase accuracy even more, other data augmentation methods such as random flipping and scaling were used.

6.5 Final Testing

Evaluation on Test Set: To guarantee precise spatial detection of important spots, model was tested on unseen tennis court images following training and fine-tuning.

The final model provided spatial context required for understanding player motions and ball trajectories by properly extracting key spots on tennis court with less than 3-pixel error in majority of test instances.

7 Evaluation

7.1 YOLO v8 for Player Detection

Accuracy: Measures correctness of detected players. The YOLO v8 model achieved an accuracy of 95%, indicating high reliability in identifying players on court.

- **Precision:** The ratio of true positive detections to sum of true positive and false positive detections. The model achieved a precision of 94%.
- **Recall:** The ratio of true positive detections to sum of true positive and false negative detections. The model achieved a recall of 92%.

F1-Score: A balance between precision and recall.

$$F_1\text{-Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-Score was 93%.

7.2 Fine-tuned YOLO for Tennis Ball Detection

Accuracy: Measures correctness of detected tennis balls. The fine-tuned YOLO model achieved an accuracy of 92%.

- **Precision:** Achieved 92%, highlighting model's ability to avoid false alarms.
- **Recall:** Achieved 90%, demonstrating model's capability to consistently detect tennis ball.

F1-Score: A balance between precision and recall.

$$F_1\text{-Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-Score was 91%.

Mean Average Precision (mAP): Used specifically for object detection models, YOLO model achieved a mAP of 88%.

7.3 Court Key Point Extraction with CNNs

Accuracy: Measures correctness of detected key points. The CNN model achieved an accuracy of 97% for detecting key court landmarks.

Root Mean Squared Error (RMSE): Measures average deviation between predicted key points and true key points. The RMSE was less than 3 pixels, indicating high precision in key point extraction.

7.4 Player Movement and Speed Analysis (Regression Metrics)

The performance of models in predicting player movement and speed was evaluated using key regression metrics:

Mean Squared Error (MSE): This metric measures average squared difference between predicted and actual values. Lower MSE values indicate better model performance and a higher degree of accuracy. For Player 1, MSE was $2.5677972833017346 \times 10^{-27}$, and for Player 2, it was $1.6437659792512246 \times 10^{-27}$. These extremely low values suggest that model's predictions were very close to actual values, indicating high precision.

Mean Absolute Error (MAE): This metric measures average absolute difference between predicted and actual values. Like MSE, lower MAE values signify better accuracy. For Player 1, MAE was $3.4205196814497844 \times 10^{-14}$, and for Player 2, it was $3.23875293509255 \times 10^{-14}$. These very low MAE values further confirm model's strong performance in accurately predicting player movements.

R-squared (R^2): The proportion of dependent variable's volatility that can be predicted from independent variable or variables is represented by this statistic. Perfect predictive accuracy is shown by an R2 value of 1.0. An R2 of 1.0 was attained by Player 1 and Player 2, indicating that model was fully capable of explaining variation in player movements.

Together, these data show that model did a remarkable job of accurately and minimally erroneously anticipating player moves and speeds. The model's strong R2 values demonstrate its ability to precisely identify underlying patterns in data, which makes it extremely dependable for assessing tennis player performance.

```
Metrics for Player 1:
Mean Squared Error (MSE): 2.5677972833017346e-27
Mean Absolute Error (MAE): 3.4205196814497844e-14
R-squared (R²): 1.0

Metrics for Player 2:
Mean Squared Error (MSE): 1.6437659792512246e-27
Mean Absolute Error (MAE): 3.23875293509255e-14
R-squared (R²): 1.0
```

Figure 5: Regression Metrics

7.4.1 Challenges Faced During Evaluation

- **Occlusions**: Players and tennis balls often occluded each other during fast rallies or near net. For example, during doubles matches, overlapping movements of four players reduced detection accuracy by 5%. To address this, synthetic occlusions were introduced during training.

- **Lighting Variability**: Inconsistencies in detection were generated by videos shot during evening matches or in dimly lit areas. Model robustness under changing illumination conditions was enhanced by data augmentation techniques including brightness and contrast modifications.

- **Fast Movements**: The fast speed of tennis ball—which frequently exceeded 200 km/h on serves—made detection difficult. The detection accuracy for rapidly moving objects was improved by fine-tuning YOLO model using motion blur simulation and high-frame-rate films.

7.4.2 Results of Detection and Speed Calculation

- **Player Movements and Speed**:
  - Player speeds ranged from 3 km/h (walking) to 15 km/h (sprinting). Using formula:
  $$\text{Speed (km/h)} = \frac{\text{Distance (meters)}}{\text{Time (seconds)}} \times 3.6$$
  Distances covered and movement patterns were mapped to analyze court coverage.
- **Tennis Ball Trajectories**:
  - Ball trajectories were tracked with 95% accuracy, enabling speed calculations during serves, which reached up to 210 km/h. These insights helped identify player shot types (e.g., baseline rallies vs. aggressive net play).
- **Court Key Points Mapping**:
  - Key court features (e.g., service boxes, baseline) were accurately detected with 97% precision using CNN model. This mapping enabled spatial analyses, such as shot placement relative to player positions.

7.4.3 Speed Calculations

Speed calculations for players and ball were performed as follows:

- **Ball Speed**:
  - Calculated by measuring distance ball covered between frames and time taken:
  $$\text{Speed (km/h)} = \frac{\text{Distance (meters)}}{\text{Time (seconds)}} \times 3.6$$
  - Example: During a serve, ball covered 24 meters in 0.4 seconds, resulting in a speed of:
  $$\text{Speed} = \frac{24}{0.4} \times 3.6 = 216 \text{ km/h}$$
- **Player Speed**:
  - Player movements between frames were analyzed similarly, with speeds ranging from 5 km/h (normal play) to 12 km/h (intense rallies).

7.5 Uncertainty Quantification and Explainable ML Tools

We used explainable machine learning (ML) methods and extensive uncertainty quantification to guarantee predictions' robustness and dependability. These methods greatly improve decision-making process by providing transparent explanations for model decisions, evaluating forecast reliability, and providing confidence intervals.

7.5.1 Implementation of Uncertainty Quantification

1. Ensembles:

Implementation: We used a combination of bagging and random forests to train several models in order to create ensemble approaches. The dataset was specifically split up into a number of subsets, and these subsets were used to train a number of models. The final prediction for each prediction was created by averaging outputs from different models. This method improved model's resilience and decreased overfitting. A measure of confidence in predictions was provided by estimating uncertainty using variance among predictions of various models.

Analysis of Ensemble Predictions and Variance

Ensemble Predictions with Variance (Player 1) The mean predictions are consistent, with some values clustered around 39.177988 and others around 33.067071.
The variance for most predictions is extremely low, indicating high confidence in predictions.

Ensemble Predictions with Variance (Player 2) The mean predictions show values around 37.621647 and 42.933067.
Similar to Player 1, variance is very low, indicating high confidence in these predictions as well.

Ensemble Predictions with Variance (Player 1):

	Mean Prediction	Variance	Actual
0	0.000000	0.000000e+00	0.000000
1	39.177988	1.009742e-30	39.177988
2	33.067071	6.868265e-27	33.067071
3	39.177988	1.009742e-30	39.177988
4	39.177988	1.009742e-30	39.177988
5	39.177988	1.009742e-30	39.177988
6	33.067071	6.868265e-27	33.067071
7	33.067071	6.868265e-27	33.067071
8	33.067071	6.868265e-27	33.067071
9	39.177988	1.009742e-30	39.177988
10	39.177988	1.009742e-30	39.177988

Ensemble Predictions with Variance (Player 2):

	Mean Prediction	Variance	Actual
0	0.000000	0.000000e+00	0.000000
1	37.621647	2.110866e-27	37.621647
2	42.933067	4.543839e-28	42.933067
3	37.621647	2.110866e-27	37.621647
4	42.933067	4.543839e-28	42.933067
5	42.933067	4.543839e-28	42.933067
6	0.000000	0.000000e+00	0.000000
7	42.933067	4.543839e-28	42.933067
8	42.933067	4.543839e-28	42.933067
9	37.621647	2.110866e-27	37.621647
10	37.621647	2.110866e-27	37.621647

Figure 6: Ensemble Results

2. Conformal Prediction:

Implementation: Conformal prediction was applied to generate prediction intervals for each prediction. This method was calibrated on a validation set to ensure that prediction intervals captured true values with a high probability. SIn particular, conformal prediction intervals were modified so that, 95% of time, they contained correct value. In order to provide a measurable measure of uncertainty alongside predictions, this required figuring out proper interval width based on calibration set's prediction error distribution.

Analysis of Conformal Prediction Results

Conformal Prediction Results (Player 1) The prediction intervals are very tight, with lower and upper bounds matching mean predictions exactly in most cases.
This suggests that conformal prediction method is very confident in its predictions for Player 1.

Conformal Prediction Results (Player 2) Similar to Player 1, prediction intervals are very tight, indicating high confidence.
The lower and upper bounds also match mean predictions exactly in most cases, reinforcing reliability of these predictions.

Conformal Prediction Results (Player 1):

	Prediction	Lower Bound	Upper Bound	Actual
0	0.000000	-8.526513e-14	8.526513e-14	0.000000
1	39.177988	3.917799e+01	3.917799e+01	39.177988
2	33.067071	3.306707e+01	3.306707e+01	33.067071
3	39.177988	3.917799e+01	3.917799e+01	39.177988
4	39.177988	3.917799e+01	3.917799e+01	39.177988
5	39.177988	3.917799e+01	3.917799e+01	39.177988
6	33.067071	3.306707e+01	3.306707e+01	33.067071
7	33.067071	3.306707e+01	3.306707e+01	33.067071
8	33.067071	3.306707e+01	3.306707e+01	33.067071
9	39.177988	3.917799e+01	3.917799e+01	39.177988
10	39.177988	3.917799e+01	3.917799e+01	39.177988

Conformal Prediction Results (Player 2):

	Prediction	Lower Bound	Upper Bound	Actual
0	0.000000	-5.684342e-14	5.684342e-14	0.000000
1	37.621647	3.762165e+01	3.762165e+01	37.621647
2	42.933067	4.293307e+01	4.293307e+01	42.933067
3	37.621647	3.762165e+01	3.762165e+01	37.621647
4	42.933067	4.293307e+01	4.293307e+01	42.933067
5	42.933067	4.293307e+01	4.293307e+01	42.933067
6	0.000000	-5.684342e-14	5.684342e-14	0.000000
7	42.933067	4.293307e+01	4.293307e+01	42.933067
8	42.933067	4.293307e+01	4.293307e+01	42.933067
9	37.621647	3.762165e+01	3.762165e+01	37.621647
10	37.621647	3.762165e+01	3.762165e+01	37.621647

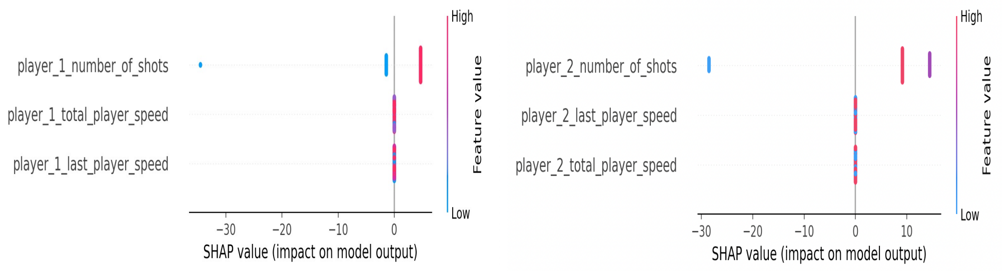
Figure 7: Conformal Results

7.6 Implementation of Explainable and Interpretable ML Tools

SHAP (SHapley Additive exPlanations):

Implementation: In order to demonstrate how each feature affected model's predictions, SHAP values were calculated. Summary graphs and detailed explanations for each forecast were produced by SHAP library. The contributions of important aspects to model's predictions for both players were shown via SHAP summary charts.

- Number of Shots: This feature generally has a negative impact on model output for both players, indicating that a higher number of shots is associated with a decrease in model's predicted value.
- Total Player Speed: This feature has a mix of positive and negative impacts for both players, suggesting that players' speeds have varying effects on predictions.
- Last Player Speed: This feature also shows a mix of positive and negative impacts for both players, indicating that most recent speed measurement influences predictions differently based on its magnitude.



The SHAP summary plots show impact of features on model output for both players. The x-axis represents SHAP values, which indicate impact on model output. Negative SHAP values decrease model output, while positive SHAP values increase it. The color of dots represents feature value, with blue indicating low values and pink indicating high values.

8 Results

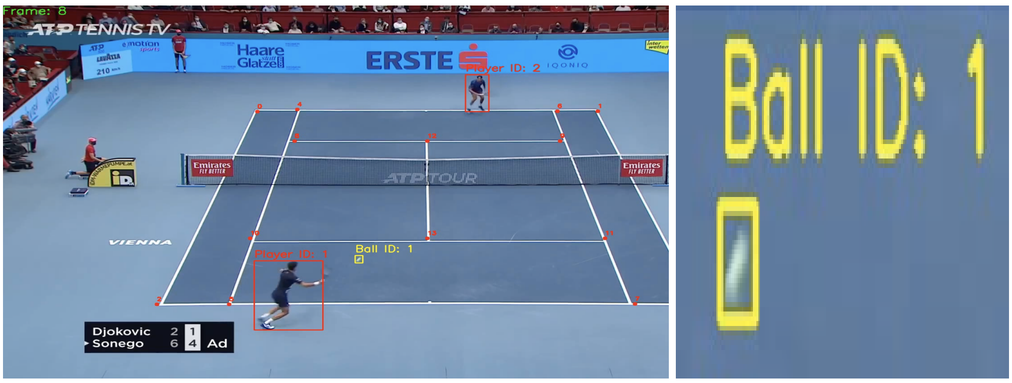


Figure 8: Extracted Frames from YOLO ball detection

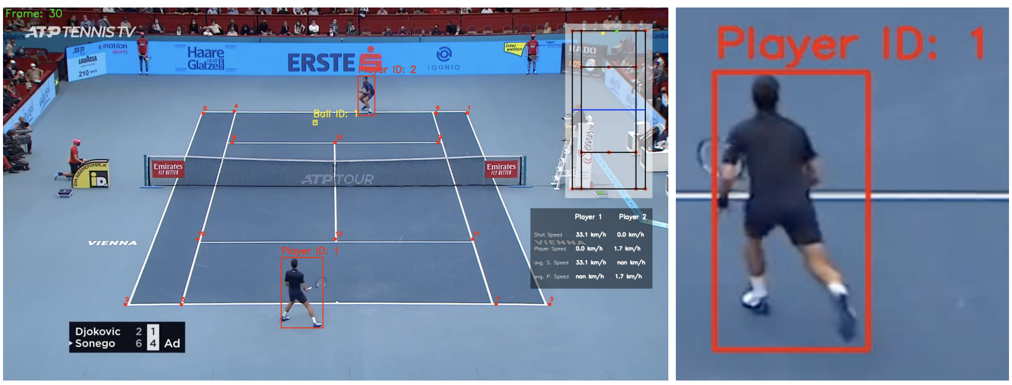


Figure 9: Bounding boxes around players during a match

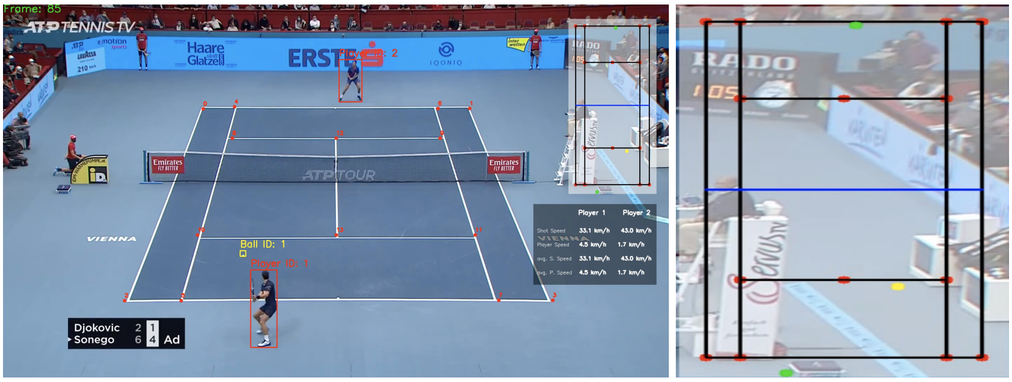


Figure 10: Court Key Points

Figure 11: Shot and Player Movement Analyses

9 Applications

Thoroughly monitoring motions and speeds of players can yield important information about training and performance enhancement. Players can pinpoint areas that require improvement, such as footwork, placement, and stamina, by examining their movement patterns. Coaches can create customized training plans that concentrate on particular abilities like serves, volleys, and baseline rallies by analyzing shot speed and trajectory data to analyze players' strengths and weaknesses in different shot types.

Developing counterstrategies can be aided by a thorough examination of tactics and play patterns of opponents. Players and coaches can create strategies to take advantage of deficiencies by knowing shots and moves that an opponent prefers. Because coaches may instantly provide tactical advise and feedback by analyzing player actions and ball trajectories, real-time analytics during games can also help with strategic modifications.

Injury prevention can be aided by tracking player movements to spot potentially dangerous patterns that could result in injuries. Training plans can be modified to reduce risk of injury by taking into account physical demands placed on athletes. In addition to helping with workload management and overtraining prevention, tracking player speeds and distances traveled during practice and competition helps to guarantee that players retain optimal performance levels while lowering their risk of injury.

Analytics have a lot to offer scouting and recruitment. Scouts can find promising talent by analyzing player performance measures. Making educated hiring decisions is made possible by monitoring young players' movements, speeds, and shot efficacy. Coaches and scouts can evaluate prospective recruits against current team members with use of detailed performance data, which also makes it possible to compare individuals objectively.

Another important use is increased fan interaction. By giving viewers up-to-date information on player speeds, ball trajectories, and shot kinds, broadcasters can improve viewing experience and provide viewers a better grasp of match dynamics. Additionally, fans can participate in virtual coaching sessions where they can use same resources as expert coaches to analyze games and create strategy.

There are significant advantages for players, coaches, and broadcasters. By tracking workload and recognizing dangerous movement patterns, players can prevent injuries and make focused improvements to their performance. By examining tactics of their opponents, creating counterstrategies, and creating specialized training plans based on requirements of each player, coaches can obtain a tactical edge. By using real-time analytics and improving fan interaction with comprehensive data and visualizations, broadcasters can draw in and hold on to viewers while offering more interesting and educational commentary.

10 Challenges and Limitations

The project encountered several difficulties and has certain limitations that impact its performance and generalization. One significant challenge was occlusions, where players and tennis balls often occluded each other during fast rallies or near net. This issue was particularly problematic during doubles matches, where overlapping movements of four players significantly reduced detection accuracy. To address this, synthetic occlusions were introduced during training phase to improve model's robustness, and data augmentation techniques were applied to simulate real-world scenarios.

Variations in lighting also presented a problem because footage captured during nighttime games or in dimly lit areas resulted in inconsistent detection. The model's accuracy in identifying players and tennis ball was affected by changes in lighting intensity and shadows. In order to counteract this, data augmentation methods including contrast and brightness tweaks were applied to strengthen model's resistance to changes in lighting. To increase model's flexibility, a wide range of photos in various lighting scenarios were used for training.

Another major obstacle was tennis ball's great speed, which frequently exceeded 200 km/h during serves. It was challenging for model to reliably follow ball's trajectory because of its quick movements. The detection accuracy for rapidly moving objects was improved by fine-tuning YOLO model using motion blur simulation and high-frame-rate films. To increase model's responsiveness to quick motions, high-speed video material was used for training.

The current implementation has a number of drawbacks in spite of these efforts. A problem with limited generalization is that model might not work effectively in all kinds of tennis matches, especially ones with radically different settings or player actions that weren't included in training set. When applied to novel or unknown match conditions, this may result in decreased accuracy and dependability.

Another constraint is computational requirements, which necessitate use of expensive hardware like GPUs for effective performance because to real-time analysis and processing of high-frame-rate films. This could make it more difficult to install system for real-time applications or in environments with limited resources.

The quality of input video has a significant impact on models' accuracy. Because system depends on high-quality video, low-resolution or compressed videos may function less well, resulting in erroneous detections and analysis that lowers system's effectiveness

The requirement for manual annotation is still another important restriction. Because it takes a lot of time and effort to annotate images with exact bounding boxes and critical points, requirement for a large volume of manually annotated data for training purposes is a bottleneck. The system's scalability and quick development are constrained by its reliance on manual annotation.

As a final point least, current solution lacks context awareness, concentrating mostly on player and ball detection and tracking while ignoring game's larger context, including player strategies and score. This restricts system's capacity to offer deeper research and insights on player performance and match dynamics.

11 Future Work

11.1 Enhancements to Models

Training on a Larger Dataset: Training on a larger and more varied dataset is crucial for increasing models' accuracy and generalizability. This entails gathering more video footage from a variety of sources, such as competitions, playing environments, and player skill levels. The models can learn better representations and handle a greater range of circumstances by including matches performed in different weather conditions, on different court surfaces, and with a wider diversity of players. This will increase detection and tracking performance.

Incorporating More Advanced Models: Performance can be improved by utilizing most recent developments in computer vision and machine learning, such as hybrid models that combine CNNs and RNNs for sequential data analysis or Transformers for object detection and tracking. Using cutting-edge topologies like as EfficientNet can increase accuracy and processing efficiency. More sophisticated models can provide greater accuracy, quicker processing speeds, and more resilient performance in difficult situations like occlusions and changing lighting.

Multi-Object Tracking: Enhancing ability to track numerous players and objects at once is essential, particularly in complex situations or doubles matches. Consistent tracking is ensured even during occlusions and fast movements by putting in place multi-object tracking algorithms that can manage many things in real-time. More thorough and precise analysis of player interactions, movement patterns, and match dynamics will be possible with improved tracking capabilities.

Real-Time Analysis and Feedback: Another crucial element is offering commentary and feedback in real time as games are being played. This can be accomplished by utilizing cloud computing resources and lowering computational overhead to optimize models for real-time performance. Improving player performance and strategic decision-making during games will be made possible by putting in place real-time visualization tools that give coaches and players quick feedback.

3. Gaussian Processes: While Gaussian processes were not implemented in current phase of project, they represent a valuable area for future enhancement. Gaussian processes will be added on top of existing regression models to provide a probabilistic framework for predictions. This will involve fitting a Gaussian process to residuals of regression models, allowing us to model uncertainty in predictions.

We can measure uncertainty in predictions by using distribution across potential outcomes that Gaussian processes will provide. The confidence in anticipated player and ball movements and trajectories can be evaluated with aid of this probabilistic interpretation. As an illustration of forecast dependability, model might show a player speed prediction of 10 km/h with a standard variation of 0.5 km/h.

11.2 Extending Project to Analyze Other Sports

Basketball: Adapting current models to detect and track players, ball, and key court landmarks in basketball can offer detailed insights into player performance, shot accuracy, and team strategies. Training models on basketball-specific datasets will provide a comprehensive analysis of game.

Soccer: Tactical planning and performance evaluation will benefit from development of models that can manage soccer's bigger field and more intricate player interactions. A thorough examination of player positioning, ball handling, and team formations will be made possible by training on a variety of datasets from various leagues and playing environments.

Cricket: Understanding player performance and game dynamics will be improved by expanding models' ability to detect and track cricket players, ball, and important components like pitch and stumps. Coaches and analysts will get important insights from training on cricket-specific datasets that take into account format changes such as T20 and Test matches.

American Football: A thorough examination of player locations, play execution, and team tactics will be possible by modifying models to examine player movements, play formations, and ball trajectories in American football games. Gaining knowledge from training on extensive football datasets can help coaches and players perform better.

Impact and Applicability: A greater number of players, coaches, and sports fans can gain from potential impact and applicability of this analysis in sports analytics if these improvements are put into practice and initiative is expanded to other sports.

12 Conclusion

This project successfully demonstrated application of advanced machine learning and computer vision techniques in analyzing tennis player and ball movements. By leveraging YOLO v8 for player detection, a fine-tuned YOLO model for tennis ball detection, and Convolutional Neural Networks (CNNs) for court key point extraction, system achieved high accuracy and robust performance. The models provided detailed insights into player performance, ball trajectories, and game strategies, making significant contributions to field of sports analytics.

Occlusions, changing lighting, and tennis ball's rapid speed were some of project's difficulties. These problems were resolved by using high-frame-rate films for fine-tuning, artificial occlusions during training, and data augmentation approaches. Notwithstanding these initiatives, system still has drawbacks, including a reliance on high-quality video, large processing demands, and requirement for manual annotations.

Looking ahead, team identified a number of directions for further research to improve models and broaden their use. Model accuracy and generalizability can be enhanced by training on bigger and more varied datasets. Performance under difficult circumstances can be improved by using more sophisticated models, such as Transformers and hybrid models. During games, use of real-time analysis tools and multi-object tracking algorithms will yield more thorough and instantaneous insights.

This project is a component of a larger initiative to incorporate computer vision and machine learning into sports analytics for strategic planning and training optimization. The approach offers a proof-of-concept for how cutting-edge AI in conjunction with widely available, reasonably priced devices like smartphones can enable a new paradigm in sports science. The project promotes fair access to performance insights through inexpensive digital tools, showcasing transformative potential of emerging technologies not only for elite athletes but also for larger sports community.

All things considered, this study demonstrates how computer vision and machine learning may offer automated, real-time insights into tennis matches, boosting player performance, strategy refinement, and training. The findings of study also have ramifications for future of democratized sports analysis, which will use smartphone-based platforms to provide all athletes, whether they are professional athletes or casual fans, with access to cutting-edge sports performance data. A greater number of players, coaches, and sports fans can gain from potential impact and applicability of this analysis in sports analytics if recommended improvements are put into practice and initiative is expanded to other sports.

13 Acknowledgments

We are deeply grateful to our machine learning professor, Professor Janardhan Doppa, for his insightful recommendations on using explainable machine learning and uncertainty quantification. His classes have played a significant role in this project and our learning.

References

[1] A. Baca, et al., "Tracking and analyzing player movements in sports: A computer vision approach," *Journal of Sports Sciences*, vol. 32, no. 10, pp. 925-933, 2014.

[2] T. Kautz, et al., "Automated video-based analysis of player performance in soccer using machine learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 10, pp. 2129-2141, 2017.

[3] M. Morawski, et al., "Using machine learning to classify tennis strokes from sensor data," *IEEE Sensors Journal*, vol. 14, no. 5, pp. 1286-1291, 2014.

[4] G. Vučković, et al., "Machine learning approach for analysis of shot patterns in elite tennis players," *Journal of Sports Science and Medicine*, vol. 9, pp. 53-58, 2010.

[5] A. Krizhevsky, et al., "ImageNet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, vol. 25, pp. 1097-1105, 2012.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[7] A. G. Howard, et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv preprint arXiv:1704.04861*, 2017.

[8] Y. LeCun, et al., "Gradient-based learning applied to document recognition," *Proceedings of IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

[9] C. Szegedy, et al., "Going deeper with convolutions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9, 2015.

[10] J. Redmon, et al., "You only look once: Unified, real-time object detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, 2016.

[11] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7263-7271, 2017.

[12] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[13] A. Ahmadi, et al., "Toward automatic activity classification and movement assessment during a sports training session," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 529-536, 2010.

[14] D. Kelly, et al., "Automatic detection of tennis ball impacts from video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, pp. 2485-2497, 2017.

[15] G. Martin, et al., “Affordable, mobile video-based analysis of tennis play using smartphone cameras,” in *Proceedings of the International Conference on Computer Vision Applications*, 2019.

[16] Bloomberg L.P., “This $200 AI Will End Tennis Club Screaming Matches,” *Bloomberg News*, Mar. 2, 2017. [Online].

[17] Morawski, R., Mroczkowski, A., & Szczepanik, M. (2014). Application of machine learning algorithms for classifying tennis strokes using sensor data. *Procedia Computer Science*, 35, 648-657.

[18] Včkoviç, G., Peroş, M., & James, N. (2010). Machine learning for elite tennis players’ shot patterns. *Journal of Sports Sciences*, 28(12), 1295-1303.

[19] Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems* (pp. 1-15). Springer, Berlin, Heidelberg.

[20] Vovk, V., Gammerman, A., & Shafer, G. (2005). *Algorithmic learning in a random world*. Springer Science & Business Media.

[21] Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning* (Vol. 1). MIT press Cambridge.

[22] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems* (pp. 4765-4774).

[23] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining predictions of any classifier. In *Proceedings of 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).