

Sport Performance Analyzer System

¹ Nitin Thakre, ²Nayan Nikure, ³Chaitali Lende, ⁴Achal Kamde, ⁵Shraddha Peshattiwar

¹Assistant Professor, ^{2,3,4,5}Students,

Department of Computer Science and Engineering

Govindrao Wanjari College of Engineering & Technology

Abstract—The Sport Performance Analyzer System is designed to monitor, analyze, and improve the performance of athletes by using data-driven techniques. The system collects performance-related data such as speed, endurance, accuracy, reaction time, and fitness parameters during training sessions or competitions. This data is processed and analyzed to provide meaningful insights into an athlete's strengths, weaknesses, and overall progress.

The system helps coaches and athletes make informed decisions by generating performance reports, visual statistics, and comparative analysis over time. It reduces dependency on manual observation, minimizes human error, and enables objective evaluation of athletic performance. The analyzer can be applied to various sports including running, football, cricket, basketball, and athletics.

By using modern technologies such as databases, analytics algorithms, and visualization tools, the Sport Performance Analyzer System improves training efficiency, enhances player performance, and supports injury prevention by tracking fatigue and workload. This system is highly beneficial for sports academies, schools, colleges, and professional training centers.

I. INTRODUCTION

In recent years, the field of sports has witnessed a technological revolution, where data analytics and artificial intelligence have become indispensable tools in assessing and improving athlete performance. Traditional coaching and performance monitoring techniques often rely on manual observation, experience-based judgment, and generalized performance metrics. While these methods have guided sports for decades, they are increasingly being complemented and in many cases, replaced by more accurate, objective, and automated systems. One such innovation is the **Sport Performance Analyzer** using **Position Mapping**, a tool designed to evaluate an athlete's movement and performance through advanced computer vision techniques implemented in **Python**.

Athletic performance is deeply rooted in the biomechanics of movement. Whether it's a sprinter's

form at the starting block, a footballer's agility on the field, or a gymnast's balance during a routine, small adjustments in body position can significantly impact outcomes. Recognizing this, the core goal of this project is to develop a software application that can analyze these movements in real time or through recorded videos, offering valuable insights into posture, coordination, and efficiency. Using **pose estimation algorithms**, the system detects and tracks body landmarks such as the head, shoulders, elbows, knees, and ankles enabling the creation of a **position map** that reflects the athlete's dynamic movements.

This position mapping is achieved using state-of-the-art libraries and frameworks, such as **Media Pipe** and **Open Pose**, which can detect human skeletal structures from image or video input. These detections are then processed using **Python's** data handling libraries like **NumPy** and **Pandas**, which calculate joint angles, velocities, and motion patterns. The output data is visualized using **Matplotlib**, **Plotly**, or similar plotting libraries, allowing coaches, physiotherapists, and athletes to analyze every detail of a physical movement.

A key feature of this system is its **ability to compare an athlete's current performance with a reference model** which could be a recording of an elite performer or a computer-generated ideal motion pattern. By overlaying and analyzing the two sets of data, the system can highlight inconsistencies, imbalances, or inefficiencies in movement. This comparative approach transforms training into a highly personalized experience, where each athlete receives feedback based on their unique biomechanical data rather than generalized averages.

II. LITERATURE REVIEW

Performance Analysis in Sport

Authors: Duarte Araújo, Micael S. Couceiro, Ludovic Seifert, Hugo Sarmento

Understanding successful sports performance requires studying how athletes adapt their movements and interact with others in changing competitive environments. This work explains the use of an ecological dynamics approach combined with artificial intelligence techniques to study sports performance and improve training design. The authors describe how modern AI tools are applied to both team-based sports such as football and individual sports like golf and climbing. The study emphasizes the importance of analyzing movement behavior and physical responses using advanced data-driven methods to gain deeper insight into athletic performance.

Performance Analysis in Sport

Authors: Miguel-Angel Gomez-Ruano, Sergio José Ibáñez, Anthony S. Leicht

Performance analysis has become an essential area within sports science, providing valuable support to coaches, players, analysts, and sports organizations across different levels of competition. The rapid growth of technology has led to the use of advanced performance measurement tools, including player tracking systems and video-based analysis techniques. These technologies enable accurate monitoring of individual and team activities during training and matches, helping improve evaluation accuracy and strategic decision-making in sports.

A System for Analysis of Indoor Team Sports Using Video Tracking and Wireless Sensor Networks

Authors: Per Wilhelm, Emad Monier, Patrick Thomas, Ulrich Ruckert

This study presents a technology-based sport performance analysis system developed specifically for indoor team sports. The system integrates high-quality video tracking with wireless sensors to record player movements and physiological parameters in real time. By merging visual and sensor-based data, the system offers a detailed and interactive performance evaluation platform. This combined approach enhances the understanding of player behavior, physical workload, and team coordination during training sessions and competitive events.

III. METHODOLOGY

The development of the Sport Performance Analyzer using Position Mapping in Python involves a multistage methodology that integrates computer vision, machine learning, and data analytics to

monitor, record, and evaluate an athlete's performance. This system aims to process video inputs, extract human pose landmarks, map positions over time, and analyze movements to provide meaningful feedback. The following describes each phase in detail:

1. Data Acquisition and Input Handling

The system begins with the acquisition of video data, which may either be real time from a webcam or prerecorded footage. Python's `OpenCV` library is utilized for capturing and reading frames from video input. This module allows the system to access and iterate through each frame individually, enabling frame by frame pose analysis.

The application interface, developed using `PyQt5`, provides a user friendly GUI where users can upload videos and view pose estimations in real time. It also provides options to save pose data, review sessions, and generate performance reports.

2. Pose Detection and Position Mapping

The core functionality of the analyzer is built using Media Pipe, a framework developed by Google for real time human pose detection. Media Pipe Pose uses machine learning models to detect and track 33 landmarks across the human body, including key joints such as elbows, shoulders, knees, hips, and ankles.

Each video frame is passed to the Media Pipe pipeline, which returns the 2D (or optionally 3D) coordinates of detected landmarks. These coordinates are mapped onto the frame, creating a visual skeleton that tracks human motion throughout the session.

To improve performance:

Frames are resized and normalized before processing. A confidence threshold is applied to discard frames with poor landmark detection.

Temporal smoothing techniques are used to stabilize landmark tracking across consecutive frames.

3. Data Processing and Feature Extraction

Once the pose landmarks are extracted, the next step involves calculating biomechanical metrics such as: Joint angles (e.g., knee, elbow, shoulder) Angular velocity and acceleration Displacement and trajectory of body parts Symmetry of movement (e.g., comparison between left and right limbs)

These features help in identifying movement patterns, detecting anomalies, and evaluating technique. The data is stored in structured formats like CSV or JSON, allowing for easy analysis and future reference.

4. Performance Analysis and Evaluation

The extracted data is compared against prerecorded benchmarks or ideal pose sequences. These benchmark datasets can be created from expert athlete recordings or defined manually.

The comparison involves:

- Overlaying user data against reference movements
- Computing deviation scores using Euclidean distance or cosine similarity between pose vectors
- Evaluating performance consistency over time
- Identifying movement inefficiencies or improper form

This module uses libraries like NumPy and SciPy to perform statistical evaluations and comparisons.

5. Visualization and Feedback Generation

The system visually represents the user's motion over time through:

- Animated skeleton overlays on original footage
 - Timeseries graphs of joint angles
 - Heatmaps of movement intensity
- These visual tools help users and coaches easily interpret performance and identify areas of improvement.
- Feedback is provided in natural language, highlighting:
- Specific actions that need correction
 - Exercises to improve weak points
 - Progress tracking compared to previous sessions

6. Export and Reporting

Finally, users can export their session data in the form of:

- Summary reports in PDF or HTML
 - Charts and visual comparisons
 - Frame by frame breakdowns with commentary
- These outputs are essential for long term tracking and sharing performance results with coaches or physiotherapists.

7. Optional Machine Learning Integration (Future Scope)

A machine learning module can be trained using labeled data to:

- Classify skill level (beginner, intermediate, expert)
- Predict injury risks based on movement patterns

Recommend personalized training plans

This would use models like Random Forest, SVM, or lightweight neural networks built using TensorFlow or scikit learn.

IV. RESULT

The development of the Sport Performance Analyzer using Position Mapping in Python involves a multistage methodology that integrates computer vision, machine learning, and data analytics to monitor, record, and evaluate an athlete's performance. This system aims to process video inputs, extract human pose landmarks, map positions over time, and analyze movements to provide meaningful feedback. The following describes each phase in detail:

1. Data Acquisition and Input Handling

The system begins with the acquisition of video data, which may either be real time from a webcam or prerecorded footage. Python's 'OpenCV' library is utilized for capturing and reading frames from video input. This module allows the system to access and iterate through each frame individually, enabling frame by frame pose analysis.

The application interface, developed using 'PyQt5', provides a user friendly GUI where users can upload videos and view pose estimations in real time. It also provides options to save pose data, review sessions, and generate performance reports.

2. Pose Detection and Position Mapping

The core functionality of the analyzer is built using Media Pipe, a framework developed by Google for real time human pose detection. Media Pipe Pose uses machine learning models to detect and track 33 landmarks across the human body, including key joints such as elbows, shoulders, knees, hips, and ankles.

Each video frame is passed to the Media Pipe pipeline, which returns the 2D (or optionally 3D) coordinates of detected landmarks. These coordinates are mapped onto the frame, creating a visual skeleton that tracks human motion throughout the session.

To improve performance:

Frames are resized and normalized before processing. A confidence threshold is applied to discard frames with poor landmark detection.

Temporal smoothing techniques are used to stabilize

landmark tracking across consecutive frames.

3. Data Processing and Feature Extraction

Once the pose landmarks are extracted, the next step involves calculating biomechanical metrics such as:
Joint angles (e.g., knee, elbow, shoulder)
Angular velocity and acceleration
Displacement and trajectory of body parts
Symmetry of movement (e.g., comparison between left and right limbs)

These features help in identifying movement patterns, detecting anomalies, and evaluating technique. The data is stored in structured formats like CSV or JSON, allowing for easy analysis and future reference.

4. Performance Analysis and Evaluation

The extracted data is compared against prerecorded benchmarks or ideal pose sequences. These benchmark datasets can be created from expert athlete recordings or defined manually.

The comparison involves:

Overlaying user data against reference movements
Computing deviation scores using Euclidean distance or cosine similarity between pose vectors
Evaluating performance consistency over time
Identifying movement inefficiencies or improper form

This module uses libraries like NumPy and SciPy to perform statistical evaluations and comparisons.

5. Visualization and Feedback Generation

The system visually represents the user's motion over time through:

Animated skeleton overlays on original footage
Timeseries graphs of joint angles
Heatmaps of movement intensity

These visual tools help users and coaches easily interpret performance and identify areas of improvement.

Feedback is provided in natural language, highlighting:

Specific actions that need correction
Exercises to improve weak points
Progress tracking compared to previous sessions

6. Export and Reporting

Finally, users can export their session data in the form of:

Summary reports in PDF or HTML
Charts and visual comparisons

Frame by frame breakdowns with commentary
These outputs are essential for long term tracking and sharing performance results with coaches or physiotherapists.

7. Optional Machine Learning Integration (Future Scope)

A machine learning module can be trained using labeled data to:

Classify skill level (beginner, intermediate, expert)
Predict injury risks based on movement patterns
Recommend personalized training plans

This would use models like Random Forest, SVM, or lightweight neural networks built using TensorFlow or scikit learn.

V. CONCLUSION

The **Sport Performance Analyzer using Position Mapping in Python** presents a significant step forward in leveraging technology to enhance athletic performance. By combining real-time video processing, pose estimation, and data analysis, this system provides athletes and coaches with a powerful tool for tracking and improving movement mechanics. The use of Python, along with libraries such as OpenCV, Media Pipe, NumPy, and PyQt5, allows for the creation of an efficient, cost-effective, and highly customizable application tailored for sports performance monitoring.

Throughout the development of this system, emphasis has been placed on accuracy, usability, and insightful feedback. The core functionality detecting human pose landmarks from video input enables the system to break down complex movements into measurable data. This data is then analyzed to identify joint angles, body positions, and overall movement flow, helping users to recognize errors, correct techniques, and improve consistency.

The visual feedback provided by the system through pose overlays and performance metrics offers a clear and intuitive way to understand athletic motion. This allows both athletes and trainers to assess form, track progress over time, and make informed decisions about training strategies. Moreover, by storing the analyzed data, the system supports long-term performance tracking and historical comparisons, further enriching its value as a training assistant.

The system's modularity also allows for future enhancements, such as integrating machine learning models for automatic classification, predictive injury analysis, or personalized training recommendations. These additions could transform it from a simple analyzer into an intelligent coaching assistant capable of adapting to various sports disciplines and skill levels.

VI. APPLICATIONS

1. Athlete Performance Evaluation

Used to analyze individual athlete performance based on speed, accuracy, stamina, and consistency.

2. Coaching and Training Support

Helps coaches design personalized training programs by identifying strengths and weaknesses of players.

3. Team Performance Analysis

Enables comparison and evaluation of overall team performance during matches and training sessions.

4. Talent Identification

Assists in identifying promising athletes based on performance data and historical analysis.

5. Injury Prevention and Fitness Monitoring

Monitors workload and physical activity to reduce injury risks and maintain player fitness.

VII. ADVANTAGES

1. Accurate Performance Evaluation

Provides precise and objective analysis of player and team performance using data driven methods.

2. Time Saving and Automated Analysis

Reduces manual effort by automating data processing, analysis, and report generation.

3. Improved Training Efficiency

Helps coaches design focused and personalized training programs based on performance insights.

4. Better Decision Making

Supports coaches and sports managers in making informed decisions regarding player selection and strategy.

5. Performance Comparison

Enables easy comparison of players and teams using historical and current performance data.

REFERENCES

[1] Lugaresi, C., et al. (2019). Media Pipe: A Framework for Building Perception Pipelines. Google AI Blog. [<https://google.github.io/mediapipe>](<https://goo>

[gle.github.io/mediapipe](https://google.github.io/mediapipe))
[2] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.
[3] Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9(3), 90–95. DOI: 10.1109/MCSE.2007.55
[4] Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
[5] van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace.
[6] Qt for Python Documentation. (2023). *PyQt5 Widgets and GUI Reference*. <https://doc.qt.io/qtforpython>
[7] Müller, P., & Gross, M. (2018). Pose Estimation for Real-Time Sports Analysis. *ACM Transactions on Graphics*, 37(4), 1-12.
[8] Zhou, X., et al. (2017). Towards 3D Human Pose Estimation in the Wild: A Weakly-supervised Approach. *IEEE International Conference on Computer Vision (ICCV)*.
[9] Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer Science & Business Media.
[10] Duarte, K., et al. (2019). Video-based Performance Assessment in Sports: A Review. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(10), 3402–3417.