SpinMaster

**Intelligent Table Tennis Analysis**

### Project Proposal

## 

## HU logo High Resolution.png

## Supervisor

Sir Aamir Hussain

## Submitted by

Syed Abraash Ul Haq

2269-2021

Mubashir Naveed

2433-2021

**Department of Computer Science,**

Hamdard University, Karachi.

[01-07-2025]

# Introduction

In the dynamic world of table tennis, precision, strategy, and adaptability are crucial for success. Leveraging advanced computing techniques, this project aims to revolutionize how players analyze and improve their game. By utilizing computer vision (CV) and machine learning (ML), the system will meticulously track player movements, ball trajectories, and shot accuracies from game recordings.

This data-driven approach enables the identification of each player's unique style, strengths, and weaknesses. Inspired by the concept of a comprehensive guide akin to the Pokémon Pokédex, this innovative tool will provide detailed profiles and strategic insights for the top players globally, as well as personalized feedback for individual users.

The ultimate goal is to equip players with the knowledge and strategies needed to enhance their performance and outmaneuver their opponents, making table tennis training more intelligent and effective than ever before.

# Objective

Develop an AI system to analyze pre-recorded table tennis gameplay using computer vision. It tracks player and ball movements, providing feedback on strengths and weaknesses like reaction time and shot accuracy.

The system also identifies illegal moves based on official table tennis rules, helping players improve their skills and adhere to regulations.

# Problem Description

What:

This project aims to create an innovative table tennis analytics platform that harnesses the power of computer vision and machine learning. By analyzing player movements, shot accuracy, and game styles from recorded matches, the system will provide detailed insights into player performance.

It will also detect and address unjust playstyles to ensure fair competition and minimize fouls, ultimately revolutionizing how players and coaches approach training and strategy development.

Why:

Table tennis is a highly dynamic sport where even minor variations in player technique can have a significant impact on game outcomes. Traditional coaching methods often rely on subjective observations and manual analysis, which may overlook subtle patterns and opportunities for improvement. By leveraging cutting-edge technology, this project seeks to enhance player training and performance evaluation by providing objective, data-driven insights.

Additionally, the system's ability to identify unjust playstyles and minimize fouls aligns with the broader goal of promoting fair and ethical competition in the sport. Ultimately, this project addresses the need for more sophisticated tools in table tennis coaching and analysis, empowering players to reach their full potential while upholding the integrity of the game.

# Methodology

Our Table Tennis Analysis System undergoes a rigorous Verification and Validation (V&V) methodology. We review the code, test individual modules, and ensure seamless system integration.

Validation involves confirming that the system meets all requirements, functions correctly, and provides a positive user experience through thorough testing and feedback.

We also evaluate data accuracy and system performance. This comprehensive approach ensures a high-quality, reliable system for the table tennis community.

# Project Scope

This project will focus on the design and development of an advanced analytics system for table tennis using computer vision and machine learning techniques. However, certain aspects will be outside the scope of this project:

**1**. **Train AI Model:**

- Develop and validate an AI model for tracking and movements.

**2. Track Gameplay:**

- Implement sprints to monitor gameplay and identify moves in game.

**3. User Feedback:**

- Gather user feedback to refine the system and address technical challenges.

**4. User Satisfaction:**

- Focus on user satisfaction, accuracy, and sprint completion.

**5. Open Source:**

- Release the project for community collaboration and improvement.

**Real-World Potential:**

This project has significant potential in the real world, particularly in the realm of sports coaching and player development. By providing detailed insights into player performance and strategies, the system can help coaches and players make more informed decisions, identify areas for improvement, and optimize training regimens. Moreover, its ability to detect unjust playstyles and minimize fouls aligns with the broader goals of promoting fair play and integrity in sports. As technology continues to play a crucial role in sports analysis and training, this project stands to make a meaningful impact in the world of table tennis and beyond.

# Feasibility Study

Project Schedule:

Given the defined scope, meeting the project schedule is feasible, provided proper planning and allocation of resources. However, certain risks and resource requirements need to be considered:

Risks Involved:

**1. Data Quality:** The availability and quality of video recordings may vary, which could affect the accuracy of the analysis. To mitigate this risk, a robust preprocessing pipeline will be implemented to handle noise and inconsistencies in the data.

**2. Algorithm Performance:** The performance of computer vision and machine learning algorithms may not meet expectations initially, requiring iterative refinement and tuning. Regular testing and validation will help identify and address any issues promptly.

**3. User Adoption:** User acceptance and adoption of the system may vary, particularly if the interface is not intuitive or the insights provided are not perceived as valuable. Conducting user testing and incorporating feedback throughout the development process will be essential to address this risk.

**Resource Requirements:**

**1. Computing Resources:** High-performance computing resources will be required for training machine learning models and performing complex data analysis tasks. This may include access to GPUs or cloud computing services such as AWS or Google Cloud Platform.

**2. Data Storage:** Adequate storage infrastructure will be needed to store and manage large volumes of video data and analysis results. This could involve setting up a dedicated database or utilizing cloud-based storage solutions.

**3. Software Development Tools:** Development tools and frameworks such as Python, OpenCV, TensorFlow, and Scikit-learn will be essential for implementing the system's algorithms and user interface.

**4. Human Resources:** Skilled personnel with expertise in computer vision, machine learning, software development, and user experience design will be required to execute the project successfully. This may involve collaboration with domain experts in table tennis coaching and sports science.

By proactively identifying and addressing potential risks and ensuring adequate resource allocation, the project can stay on schedule and deliver the desired outcomes within the defined scope. Regular monitoring and adaptation to changing circumstances will be key to managing risks and optimizing resource utilization throughout the project lifecycle.

# Solution Application Areas

This project holds substantial real-world value, particularly in the sports industry and the domain of athletic performance analysis. The target domain for this solution could include:

1. Sports Coaching and Training:

- Coaches and trainers can use the analytics system to gain deep insights into player performance, identifying strengths, weaknesses, and areas for improvement.

- By understanding player game styles and strategies, coaches can tailor training regimens to enhance individual and team performance effectively.

2. Athlete Development Programs:

- National and regional athlete development programs can leverage the system to scout and identify talented players based on their performance metrics and playing styles.

- Athletes can use the system to track their progress over time, set goals, and optimize their training strategies for maximum effectiveness.

3. Sports Science Research:

- Researchers in sports science and biomechanics can use the system to study player movements, shot techniques, and game dynamics, contributing to a deeper understanding of athletic performance in table tennis.

- Insights gleaned from the system's analysis could inform the development of new training methodologies, equipment designs, and injury prevention strategies.

4. Broadcasting and Commentary:

- Broadcasters and commentators can enrich the viewing experience for audiences by incorporating real-time insights and analysis generated by the system during live matches.

- Viewers can gain a deeper understanding of the nuances of table tennis gameplay and appreciate the skill and strategy involved in professional matches.

Overall, the target domain stands to benefit significantly from this solution by improving player performance, enhancing coaching effectiveness, advancing sports science research, and enriching the spectator experience. By providing actionable insights and strategic recommendations based on data-driven analysis, the system can contribute to the overall growth and development of table tennis as a sport.

# Tools/Technology

Certainly! Here's a breakdown of the hardware, software tools, and technologies required for the project:

Hardware:

1. Computing Infrastructure:

- High-performance computing resources for training machine learning models and performing complex data analysis tasks. This may include:

- CPUs/GPUs with sufficient processing power.

- Sufficient RAM for data processing.

- Storage space for storing video recordings and analysis results.

Software Tools and Technologies:

1. Programming Languages:

- Python: Main programming language for implementing the system's algorithms and functionalities.

- JavaScript/HTML/CSS: For developing the user interface .

2. Libraries and Frameworks:

- OpenCV: For computer vision tasks such as player and ball tracking, shot detection, and image processing.

- TensorFlow or PyTorch: Deep learning frameworks for training and deploying machine learning models.

- Scikit-learn: For implementing machine learning algorithms such as SVM or Random Forests.

- Flask or Django: Web frameworks for building the user interface and backend services.

3. Database Management:

- SQL or NoSQL databases for storing player profiles, game data, and analysis results. Options include:

- PostgreSQL, MySQL for relational databases.

- Firebase for cloud-based databases.

4. Development Tools:

- Integrated Development Environment (IDE) such as PyCharm, VS Code, or Jupyter Notebook for coding and debugging.

- Git for version control to manage code changes and collaboration among team members.

5. Additional Tools:

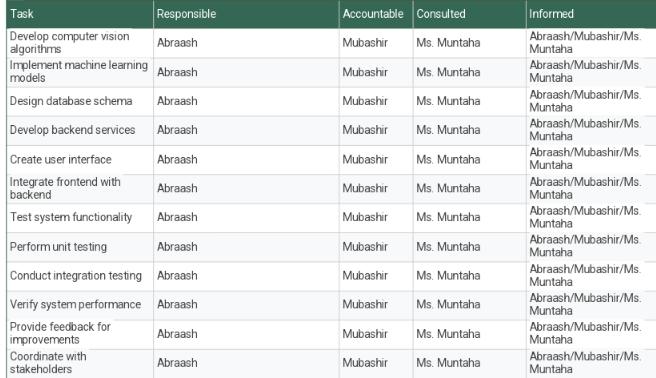
- Video editing software for preprocessing and editing recorded game footage if necessary.

- Communication and collaboration tools for project management and team coordination.

By leveraging these hardware and software tools and technologies, the project can effectively implement the advanced analytics system for table tennis, providing valuable insights and strategic recommendations for players, coaches, and sports enthusiasts.

# Responsibilities of the Team Members

Certainly! Here's a RACI matrix for the project, assigning roles and responsibilities to the project partners Abraash (Developer) and Mubashir (Quality Assurance):



Legend:

- R (Responsible): The person responsible for completing the task.

- A (Accountable): The person ultimately accountable for the task's success or failure.

- C (Consulted): The person or group to be consulted for input or expertise before the task can be completed.

- I (Informed): The person or group who needs to be informed of the task's progress or completion.

In this matrix:

- Abraash (Developer) is primarily responsible for developing the system components, including computer vision algorithms, machine learning models, backend services, and user interface.

- Mubashir (Quality Assurance) is primarily responsible for testing the system's functionality, performing unit and integration testing, verifying system performance, and providing feedback for improvements.

- Abraash ,Mubashir and Sir Aamir Hussain may collaborate and provide input or expertise as needed, especially during the development and testing phases. They also need to coordinate with stakeholders and keep them informed of the project's progress.

# 10. Planning

Here are the major tasks and sub-tasks to be accomplished for the project, along with their start and end dates and allocated resources:

1. Project Planning and Setup:

* Define project scope and objectives: Q1 (Start: Day 1, End: Day 3)
* Identify project partners and assign roles: Q1 (Start: Day 1, End: Day 3)
* Set up development environment and tools: Q1 (Start: Day 4, End: Day 5)
* Allocate computing resources: Q1 (Start: Day 4, End: Day 5)
* Gather initial requirements and specifications: Q1 (Start: Day 2, End: Day 7)

-Allocated resources: Abraash (Developer), Mubashir (QualityAssurance)

2. Data Collection and Preprocessing:

* Collect video recordings of table tennis matches: Q2 (Start: Day 6, End: Day 14)
* Preprocess video data: Q2 (Start: Day 10, End: Day 14)

-Allocated resources: Abraash (Developer)

3. Computer Vision Development:

* Develop player tracking algorithms: Q2 (Start: Day 15, End: Day 21)
* Implement ball tracking algorithms: Q2 (Start: Day 18, End: Day 24)
* Develop shot detection algorithms: Q2 to Q3 (Start: Day 22, End: Day 28)

- Allocate resources: Abraash (Developer)

4. Machine Learning Model Development:

* Collect and label training data: Q2 to Q3 (Start: Day 25, End: Day 35)
* Train machine learning models: Q3 (Start: Day 30, End: Day 42)
* Validate and fine-tune models: Q3 (Start: Day 40, End: Day 49 )

- Allocate resources: Abraash (Developer)

5. Backend Development:

* Design database schema: Q3 (Start: Day 35, End: Day 42)
* Develop backend services: Q3 (Start: Day 43, End: Day 56)

- Allocate resources: Abraash (Developer)

6. User Interface Development:

* Design user interface mockups: Q3 (Start: Day 42, End: Day 49)
* Develop frontend components: Q3 (Start: Day 50, End: Day 63)
* Integrate frontend with backend: Q3 to Q4 (Start: Day 60, End: Day 70)

- Allocate resources: Abraash (Developer)

7. Testing and Quality Assurance:

* Perform unit testing: Q4 (Start: Day 64, End: Day 70)
* Conduct integration testing: Q4 (Start: Day 68, End: Day 76)
* Verify system performance: Q4 (Start: Day 72, End: Day 80)
* Provide feedback for improvements: Q4 (Start: Day 76, End: Day 84)

- Allocate resources: Mubashir (Quality Assurance)

8. Documentation and Deployment:

* Prepare project documentation: Q4 (Start: Day 77, End: Day 84)
* Deploy system to production environment: Q4 (Start: Day 85, End: Day 90)

-Allocated resources: Abraash (Developer), Mubashir (Quality Assurance)

Here's the Gantt chart illustrating the project timeline:



This Gantt chart outlines the timeline for completing each major task/sub-task, including start and end dates and the allocated resources. It provides a visual representation of the project schedule, allowing for efficient project management and tracking of progress.

# 11. References

1. OSAI.ai. (2022). Table Tennis Stroke Dataset [Dataset]. <https://github.com/OSAI-ai/TTNet>
2. Google Research. (2023). MediaPipe: Cross-platform, customizable ML solutions for live and streaming media. <https://developers.google.com/mediapipe>
3. Renotte, N. (2021). Deep Learning and Object Detection with YOLO and Roboflow [Linkedin Profile]. <https://www.linkedin.com/in/nicholasrenotte>
4. Roboflow. (2023). Build, Train, and Deploy Computer Vision Models. [https://roboflow.com](https://roboflow.com/)
5. World Table Tennis (WTT). (2023). Official Player Rankings and Match Statistics. [https://worldtabletennis.com](https://worldtabletennis.com/)
6. Ultralytics. (2024). YOLOv8 - Real-Time Object Detection Framework [Computer software]. GitHub. <https://github.com/ultralytics/ultralytics>
7. Bradski, G. (2000). The OpenCV Library. Dr. Dobb’s Journal of Software Tools. [https://opencv.org](https://opencv.org/)
8. Grinberg, M. (2018). Flask Web Development: Developing Web Applications with Python. O'Reilly Media.
9. Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
10. McKinney, W. (2012). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. O'Reilly Media.
11. Van Rossum, G., & Drake, F. L. (2009). The Python Language Reference Manual. Network Theory Ltd.
12. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
13. Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array Programming with NumPy. Nature, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
14. Reback, J., McKinney, W., jbrockmendel, Van den Bossche, J., Augspurger, T., Cloud, P., ... & Hawkins, S. (2020). Pandas: Powerful Python data analysis toolkit [Computer software]. [https://pandas.pydata.org](https://pandas.pydata.org/)

### Tools & Libraries Used:

* **Flask** – Lightweight backend framework for handling routes, sessions, and form handling
* **OpenCV** – Video I/O, frame extraction, and image manipulation
* **MediaPipe** – Landmark-based pose detection (for elbow, shoulder, wrist tracking)
* **Pandas** – CSV parsing, aggregation, and analysis
* **NumPy** – Core numerical computing and angle operations
* **Matplotlib** (optional) – Can be used for future visualizations
* **YOLOv8 (Ultralytics)** – (Future use) for ball detection and shot-level event detection
* **TTNet** – Deep learning model for forehand/backhand/stroke classification
* **Roboflow** – Data labeling and YOLO training augmentation (future roadmap)