An Observant System to Lower Financial Barriers in Tennis using Motion-Tracking Hardware, Mobile Application, and Artificial Intelligence

Xiaotiao Wang[1], Tyler Boulom[2]

[1]Portola High School, 1001 Cadence, Irvine, CA 92618
[2]Woodbury University, 7500 N Glenoaks Blvd, Burbank, CA 91504

Abstract

Tennis is a globally growing sport with many people flocking to it in search of enjoyment or competition, but countless players are ultimately blocked by a financial barrier. A system is proposed to address this issue by providing an alternative to expensive coaching. The project, named My Swing Tracker, is comprised of a hardware piece to capture swing motion with an application to display calculated data and connect users to an AI coach that will give feedback based on their data. The hardware was coded in Python and uses a microcontroller, accelerometer, and battery. The application was made using the Flutter Software Development Kit and coded in Dart. When the hardware piece was first tested, it detected extraneous movement such as backswing, so later iterations implemented a calibration system that made the detection ignore certain axes on the 3D plane. The idea of AI coaching without human interaction was tested, and it was validated that the AI system would be accurate and that there would be no decrease in player intensity in training. MySwingTracker can help players of all levels because it focuses on the core of tennis in the swing motion, which is essential for learning tactics and new situational strokes

Keywords

Tennis, Accessibility, Hardware/Electronics, AI, App development

1. Introduction

The high costs of modern tennis lessons serve as a barrier for many aspiring players, especially those from a middle- or lower-income background. Tennis has always been perceived as a sport for the elite and wealthy [1]. However, as sport becomes more globalized and popular, the high costs associated with learning remain. Thus, it shapes not just who gets to play the sport, but who persists in the sport [2]. Coaching lessons are sought out by players because they are imperative to player improvement [3]. However, currently, in many places, like my local area of Southern California, there are private lessons that frequently cost nearly $100 [4]. Many players who look for a quick fix end up overspending on an entire $100 lesson that lasts an hour. This high cost is a problem because it discourages people from playing tennis, a sport that not only provides health benefits but also enjoyment for millions around the world [5][6]. Therefore, because sports participation is linked to improved well-being, economic barriers limiting access to tennis may disproportionately restrict these benefits from lower-income groups [7]. Research shows that

David C. Wyld et al. (Eds): CSITEC, NLPI, CMCA, NECOM, DMML, SP, ADCO – 2026
pp. 179-190, 2026. CS & IT - CSCP 2026 DOI: 10.5121/csit.2026.160815

socioeconomic status is a significant factor in both continuing tennis participation and achieving success in the sport, due mainly to the financial demands associated with coaching and competitive development [8]. Furthermore, at the competitive junior level, where many like me fit, families spend $5,000 to $25,000 a year for one child to learn tennis from high-quality trainers [9]. This trend of participation being contingent on families' ability to pay for coaching, equipment, travel, and competition fees is seen across modern youth sport, creating a systematic advantage for high-income families and reinforcing socioeconomic disparities in sport access and persistence [10]. This is a huge chunk of the average American household income of $80,000 [11]. Even adults who consider tennis only a hobby must still spend thousands a year. For many families and individuals, the price tag on pursuing tennis is too much for their income and an obstruction to their passion.

The first methodology, discussed in "Towards a Sustainable Financial Model for Professional Tennis Players," focuses on restructuring the professional tennis system to financially support lower-ranked players. While this approach addresses systemic inequality, it requires cooperation from numerous tournaments and organizations, making it unrealistic. MySwingTracker instead reduces individual coaching costs without requiring institutional change. The second methodology, proposed by Andrew Hawling in "Electronic Tennis Officiating: Low Cost, Accurate and Reliable Solutions," introduces a hardware-based officiating system to lower tournament expenses. However, it does not benefit developing players and is vulnerable to interference. MySwingTracker improves on this by supporting player development at all skill levels. The third methodology, from Jacob G. Emery's "Helping Youth Athletes Thrive," presents a motivational app to address mental stagnation in youth players. While effective for motivation, it does not improve technical skills. MySwingTracker addresses this gap by enabling measurable skill improvement that sustains motivation.

This study proposes a system that addresses this obstacle through the integration of hardware for real-time calculations, a mobile application for data visualization, and an AI assistant. In this study, a tennis training assistant is implemented within a hardware device, consisting mainly of a microcontroller and an accelerometer. The hardware is a compact device attached to the throat of the racket, and its function is to collect the user's racket movement to calculate data about the user's swing. There will also be an application to pair with the hardware that displays the data and gives feedback through an AI assistant powered by OpenAI. A key advantage of the hardware over coaches is that it is quickly accessible to users whenever they are in need of training. The intended impact of this system is to broaden access to analytical tennis training by using low-cost hardware to replace coaches, particularly in high-cost regions where access to coaching is financially restrictive. In doing so, the project will lower financial barriers to tennis, attract more athletes, and provide opportunities for players of all backgrounds. Although other technological tennis aides like SwingVision track your gameplay too, it doesn't go in depth about the quality of your shots and mainly tracks the ball's data (Source). My solution gives feedback on the specific stroke and swing of the player, making it more personalized and beneficial for those who want to know specific things to improve.

The fundamental proposal idea of MySwingTracker, replacing human and physical coaches, was first tested in effectiveness through the analysis of an experiment conducted by sports science professor Bulent Kilit. The research on this experiment revealed that while youth tennis players enjoy practicing more when there is a coach present, these players are actually prone to put in more effort when coaches are absent. This finding greatly supported the solo-coaching model of MySwingTracker, as it showed that it could be an effective alternative to tennis coaches. The second experiment was aimed at testing the accuracy of the AI feedback, and it was done so through two sets of testing on a beginner player's spin metric: one for before any feedback, and one for after reading the AI feedback. The AI gave specific instructions to help the beginner, and the experiment showed a drastic increase in the tested spin metric. This experiment determined that the AI was accurate in its feedback and could effectively help players improve.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. DESIGNING A STABLE RACKET-MOUNTED DEVICE CASING

One obstacle faced was the designing and implementation of a computer-aided design casingto attach my device onto the tennis racket. A lot of brainstorming went into this, as the placement of the case needed to not affect the user's gameplay, keep the racket in balance, and be in a stable spot. I eventually came to resolve this issue by creating a special case to fit on the throat part of the racket between strings and the handle. The casing was made to be trapezoidal to fit and be in a position where it can be binded easily.

2.2. CALCULATING SWING STATISTICS FROM LIMITED SENSOR DATA

The most crucial component of this project is the coding of the device, especially the formulas for calculating the statistics of a swing. The challenge starts with how the device is given limited data from a small accelerometer, which are the position and the rotational status of the device. To receive more data and calculate the exerted force of the user, the device first asks the user to input their racket's mass. The microcontroller receives the data and calculates the acceleration, which is then used to compute the force component. The changing positions tracked by the accelerometer in consideration of the time elapsed between positions yields the speed of the swing. The spin factor is able to be taken directly from the gyro.

2.3. INTEGRATING OPEN AI FOR AI PERFORMANCE FEEDBACK

The end functionality of the project is to give feedback on a user's performance, which requires an AI assistant. The app uses OpenAI, which has to be called on and inserted into the application. This required choosing the correct OpenAI model, and getting its API keys securely inserted into the application.

3. SOLUTION

The program contains three crucial components: the calibration of the device, the calculated data sent from the device, and the application where data is stored and displayed. The hardware is a compact device attached to the throat of the racket, and its function is to collect necessary data about the racket movement as the user swings. The device is coded in Python and is made up of a microcontroller to process information, an accelerometer to gather raw information, and a battery to make the device portable. A calibration system is in place to ensure accurate detection and collection of data. The system prompts the user to repeatedly swing in the direction of the stroke they want to practice. The hardware then keeps track of that direction to ensure it detects only the accurate data. The device then feeds the application the data that it calculates. The dataset is transferred to the application through a Bluetooth UART Server, which allows one device to transmit data to another wirelessly. The program will also include an essential application that is free to download on the app store. The application is created in Flutter and it uses Firebase to authenticate users and store the user's data. Upon opening the app, it will prompt the user to log in. Otherwise, if the user doesn't have an account, they can sign up for one. Once the user is logged into the account, they are able to view their past sessions and their performance data of power, speed, and spin during those sessions. The user will also have the choice to start a new session, which starts by searching and connecting the hardware to the app through bluetooth. In the user's session history, there is an option to get feedback from an AI assistant, which tells the user their areas of improvement and provides specific drills.

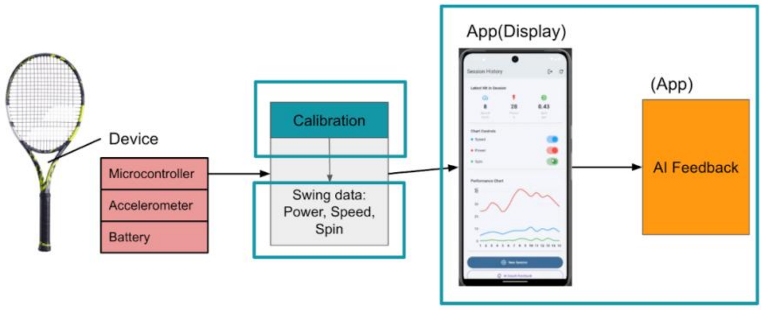

FIGURE 1.OVERVIEW OF THE SOLUTION

The calibration system is a crucial component of the device, as it drastically increases the quality of the feedback given to the user. It essentially excludes all other motions, such as the takeback of the racket, and only includes the swing. Therefore, the calibration is what makes the data accurate.

Figure 2. UART Communication Interface Used For Device–Application Data Transmission

```python
def calibrate():
    uart.write("Calibration mode: Swing repeatedly in your hitting direction...\n".encode())
    print("Calibration mode: Swing repeatedly in your hitting direction...")
    print("Watching for consistent directional spikes...\n")

    DURATION = 7  # seconds
    ACCEL_THRESHOLD = 6.0
    POSITIVE_STREAKS = [0, 0, 0]
    NEGATIVE_STREAKS = [0, 0, 0]

    start = time.monotonic()
    while time.monotonic() - start < DURATION:
        lin_accel = sensor.linear_acceleration
        if lin_accel is None:
            continue
        for i in range(3):
            if lin_accel[i] > ACCEL_THRESHOLD:
                POSITIVE_STREAKS[i] += 1
            elif lin_accel[i] < -ACCEL_THRESHOLD:
                NEGATIVE_STREAKS[i] += 1
        print(f"X: {lin_accel[0]:.1f}, Y: {lin_accel[1]:.1f}, Z: {lin_accel[2]:.1f}")
        time.sleep(0.1)

    max_pos = max(POSITIVE_STREAKS)
    max_neg = max(NEGATIVE_STREAKS)
    if max_pos >= max_neg:
        dominant_index = POSITIVE_STREAKS.index(max_pos)
        sign = 1
    else:
        dominant_index = NEGATIVE_STREAKS.index(max_neg)
        sign = -1

    axis_names = ['X', 'Y', 'Z']
    result = f"\nCalibration complete.\nDominant axis: {axis_names[dominant_index]}, Direction: {'+' if sign == 1 else '-'}\n"
    print(result)
    uart.write(result.encode())
    return dominant_index, sign
```

Figure 3. Calibration algorithm implemented for swing-axis detection

The code sample shows the calibration function, which is later called when the user requests it through the shown UART server. The function first initializes the variables: the duration of the calibration; the acceleration threshold to consider a swing; and the positive and negative streaks to keep track of which direction the user swings the most. A while loop, running based on the duration, repeatedly collects the acceleration vector of the swing, and counts the amount of negative and positive movement of each direction(X, Y, Z). It collects the statistics of each swing through the accelerometer's 'linear_acceleration' function, and checks every direction with the threshold to determine adding to that direction or not.. After the set duration, the function ends the loops and moves on to solve for the dominant X, Y, and Z axis and whether it is in the positive or negative direction. It creates two variables, max_pos and max_neg, that find the max value of their respective positive or negative direction. Finally, it compares the maxes and returns the dominant index and the positive or negative sign.

The Bluetooth connection is very crucial, as it is essentially what connects the device and the application. Through Bluetooth, a UART server between the device and the application is established. This is what allows the device to send over the collected data and display it for the application to analyze and display for the user.

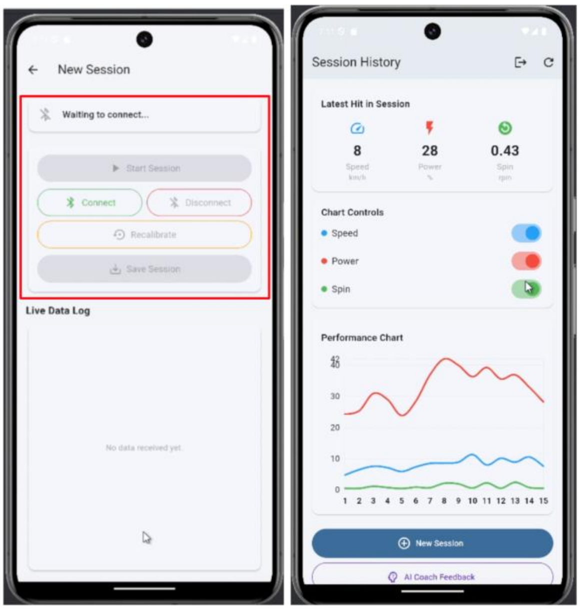

Figure 4. User interface displaying session history and performance metrics

```
FlutterBluePlus.startScan(timeout: const Duration(seconds: 10));
FlutterBluePlus.scanResults.listen((results) {
  if (_isConnected || !_isConnecting) return;
  for (ScanResult r in results) {
    if (r.device.platformName == "TennisTracker") {
      FlutterBluePlus.stopScan();
      _connectToDevice(r.device);
      break;
    }
  }
});
```

```
Future<void> _setupNotifications(BluetoothCharacteristic characteristic) async {
  await characteristic.setNotifyValue(true);
  final StringBuffer buffer = StringBuffer();
  characteristic.lastValueStream.listen((value) {
    buffer.write(utf8.decode(value, allowMalformed: true));
    while (buffer.toString().contains('-------------\n')) {
      final fullText = buffer.toString();
      final endIndex = fullText.indexOf('-------------\n') + '-------------\n'.length;
      final swingBlock = fullText.substring(0, endIndex);
      _parseAndStoreSwingData(swingBlock);
      setState(() {
        _dataLog.insert(0, swingBlock);
      });
      buffer.clear();
      buffer.write(fullText.substring(endIndex));
    }
  });
}

void _parseAndStoreSwingData(String block) {
  final lines = block.split('\n');
  final Map<String, dynamic> hitData = {};
  for (var line in lines) {
    if (line.contains("Power Index:")) {
      hitData['power'] = double.tryParse(line.split(':')[1].trim()) ?? 0.0;
    } else if (line.contains("Speed Index:")) {
      hitData['speed'] = double.tryParse(line.split(':')[1].trim()) ?? 0.0;
    } else if (line.contains("Spin Index:")) {
      hitData['spin'] = double.tryParse(line.split(':')[1].trim()) ?? 0.0;
    }
  }
  if (hitData.containsKey('power') && hitData.containsKey('speed') && hitData.containsKey('spin')) {
    _sessionHits.add(hitData);
    print("Stored hit: $hitData");
  }
}
```

Figure 5. Bluetooth data processing routine for swing metrics extraction

In the first code block sample, the application is prompted to start scanning for a Bluetooth device. It keeps scanning until the condition that the device “TennisTracker” is connected. This section corresponds with the first user interface with buttons that the user can press to connect, disconnect, and recalibrate. In the next segment of the code, the application retrieves the data sent to the UART buffer, and converts it to a string so that more methods can be called to manipulate the presentation of the data. This is achieved through a while loop that is set to run for as long as a preset indicator for data exists in the buffer. After processing the data into string form, the application begins to separate the data into power, speed, and spin by identifying the occurrences of the respective words in the string. Finally the swing data stored in variable ‘hitData’ is added to the live hits of that session.

The third component that significantly contributes to the functionality of this system is the Firebase server that the application uses. Firebase allows users to create accounts on the application, where they can then store their past sessions and reflect on those performances. This component allows users to track their progress over time and see the trend of their improvement.

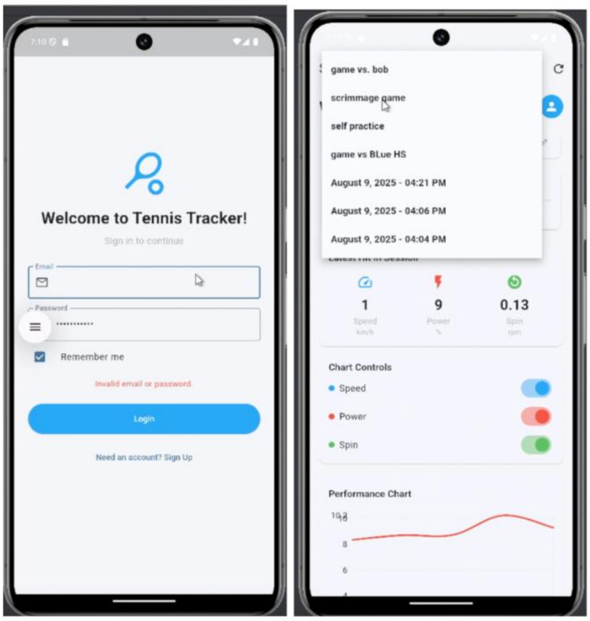

Figure 6. Mobile application interface for user session management and performance visualization

```
Future<String?> signInWithEmailAndPassword(
    String email, String password) async {
  try {
    await _auth.signInWithEmailAndPassword(
        email: email, password: password);
    return null; // Success
  } on FirebaseAuthException catch (e) {
    // Return a user-friendly error message.
    if (e.code == 'user-not-found' || e.code == 'wrong-password' || e.code == 'invalid-credential') {
      return 'Invalid email or password.';
    }
    return 'An error occurred. Please try again.';
  }
}
```

```
Future<void> createUserProfileInDB(
    {required String userId,
      required String name,
      required String email}) async {
  try {
    DatabaseReference userRef = _database.ref('sessions/$userId');
    await userRef.set({
      'placeholder_session': {
        'user_id': userId,
        'name': name,
        'email': email,
        'timestamp': DateTime.now().toIso8601String(),
        'session_id': 'placeholder_session',
        'hits': []
      }
    });
  } catch (e) {
    print("Error creating user profile in DB: $e");
  }
}
```

```
Future<void> _saveSessionToDatabase() async {
  final User? user = _auth.currentUser;
  if (user == null) {
    ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text("Error: You must be logged in to save.")));
    return;
  }
  if (_sessionHits.isEmpty) {
    ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text("No hits to save.")));
    return;
  }
  setState(() { _status = "Saving session..."; });
  try {
    final userRef = _database.ref('sessions/${user.uid}');

    // --- FIXED: Fetch existing user data to get the name reliably ---
    String userName = user.displayName ?? "No Name"; // Fallback
    final snapshot = await userRef.get();
    if (snapshot.exists && snapshot.value != null) {
      final userData = Map<String, dynamic>.from(snapshot.value as Map);
      if (userData.isNotEmpty) {
        // Get the name from the first available session record
        final firstSessionKey = userData.keys.first;
        userName = userData[firstSessionKey]['name'] ?? userName;
      }
    }
    // --- End of fix ---

    final sessionId = 'session_${DateTime.now().millisecondsSinceEpoch}';
    final email = user.email ?? "no-email@example.com";

    final Map<String, dynamic> sessionData = {
      'user_id': user.uid,
      'name': userName, // Use the reliably fetched name
      'email': email,
      'session_id': sessionId,
      'timestamp': DateTime.now().toIso8601String(),
      'hits': _sessionHits,
    };

    await userRef.child(sessionId).set(sessionData);
    await userRef.child('placeholder_session').remove();
```

Figure 7. Firebase authentication and session storage implementation

In the first code snapshot, there is a function that takes in the user's email and password, where it then checks with the Firebase database to determine whether the credentials entered are valid or not. The next section provided is the function that creates a new user profile during sign up in the Firebase database, as shown by all the placeholder information for a new user. Lastly, the final section of code runs when the user finishes a tennis session, and is saved for future review. Before it saves the session, it first checks if the user is logged in and whether any hits were recorded. If either condition is not met, the app sends an error message. Otherwise, if the session is valid, the code begins to prepare the data to be stored. It then creates a session ID for that session, and finally gets the user's information and saves all of it to a unique session data map.

4. EXPERIMENT

4.1. EXPERIMENT 1

A notable change in this project is that there is no human interaction during the coaching of the user. The replacement of the human coach with an AI has downsides that come with its benefits.
In the design to reveal the value of having an actual person for a coach, this paper will be referencing the research from "Effect of Coach Encouragement on the Psychophysiological and Performance Responses of Young Tennis Players" [12]. The article analyzes the presence of coaches and their in-person support during drills on the "psychophysiological" performance of tennis players. The tested group is comprised of twenty-five male teens and were examined on

their performance on a set of on-court tennis training drills, such as Star, Suicide, Box, and Big X. The performances of the subjects were then recorded with support from a coach, and then without.

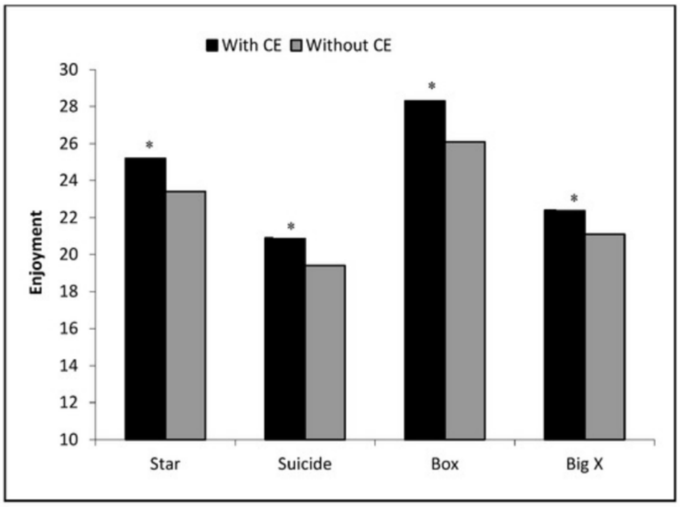

Figure 8. Figure of Enjoyment of with CE and without CE

Variables	Variables	With CE	Without CE	Difference	d	ES Magnitude
Star	HR (beats·min^{-1})	176.3 ± 9.4 *	173.9 ± 10.0	2.4	0.25	Small
	%HR$_{max}$ (%)	87.4 ± 4.1 *	86.2 ± 4.9	1.2	0.27	Small
	RPE	6.9 ± 0.6 *	7.1 ± 0.6	−0.2	0.33	Small
	PACES	25.2 ± 4.4 *	23.4 ± 4.9	1.8	0.39	Small
	Total Distance (m)	28.8 ± 3.4 *	26.9 ± 3.8	1.9	0.53	Small
	Number of Shots	8.8 ± 1.2 *	8.0 ± 1.5	0.8	0.59	Small
Suicide	HR (beats·min^{-1})	178.4 ± 8.5 $^{\Omega}$	176.2 ± 8.8	2.2	0.25	Small
	%HR$_{max}$ (%)	88.5 ± 4.2 $^{\Omega}$	87.3 ± 4.3	1.2	0.28	Small
	RPE	7.6 ± 0.5 $^{\Omega}$	7.9 ± 0.6	−0.3	0.54	Small
	PACES	20.9 ± 6.3 $^{\Omega}$	19.4 ± 6.8	1.5	0.23	Small
	Total Distance (m)	46.8 ± 2.8 $^{\Omega}$	43.1 ± 2.6	3.7	1.37	Large
	Number of Shots	2.9 ± 0.1 $^{\Omega}$	2.7 ± 0.1	0.2	1.99	Large
Box	HR (beats·min^{-1})	173.9 ± 9.6 #	171.8 ± 10.1	2.1	0.21	Small
	%HR$_{max}$ (%)	86.2 ± 4.7 #	85.2 ± 5.0	1.0	0.21	Small
	RPE	6.4 ± 0.5 #	6.8 ± 0.5	−0.4	0.80	Moderate
	PACES	28.3 ± 3.7 #	26.1 ± 4.0	2.2	0.57	Small
	Total Distance (m)	42.5 ± 5.1 #	40.8 ± 5.7	1.7	0.31	Small
	Number of Shots	9.7 ± 1.7 #	8.9 ± 2.0	1.8	0.97	Moderate
Big X	HR (beats·min^{-1})	174.2 ± 9.5 ¥	172.4 ± 10.0	1.8	0.18	Trivial
	%HR$_{max}$ (%)	86.4 ± 4.8 ¥	85.5 ± 5.1	0.9	0.18	Trivial
	RPE	7.5 ± 0.4 ¥	7.7 ± 0.5	−0.2	0.44	Small
	PACES	22.4 ± 5.1 ¥	21.1 ± 4.9	1.3	0.26	Small
	Total Distance (m)	42.6 ± 3.1 ¥	40.9 ± 3.3	1.7	0.54	Small
	Number of Shots	3.0 ± 0.3 ¥	2.9 ± 0.3	0.1	0.54	Small

HR = heart rate; %HR$_{max}$ = percentage of maximum heart rate; RPE = rating of perceived exertion; PACES = Physical Activity Enjoyment Scale. Values are given as mean ± SD. * Significant differences from Star without CE, $p < 0.05$. $^{\Omega}$ Significant differences from Suicide without CE, $p < 0.05$. # Significant differences from Box without CE, $p < 0.05$. ¥ Significant differences from Big X without CE, $p < 0.05$.

Table 1. Psychophysiological and performance responses of young tennis players for with CE and without CE

The graph displays the enjoyment level, collected from the group of young tennis players, in each drill with and without coach engagement (CE in the graphs). It is clear that with coach engagement, the tennis players were more likely to enjoy each of the four drills. The increased enjoyment coach engagement brings implies that the tennis players will be likely willing to do further training, which means more improvement among the players. In the second graph from the research, the physical performances of the group are displayed numerically. In every drill, it is shown that the heart rate and percentage of maximum heart rate are higher in the test cases with coach engagement than those without. This proves that having a coach can improve a player's intensity, as the human coach can continuously encourage the players to train harder. However, a contrasting piece of data from the graph is that the rating of perceived exertion is lower in every drill when there is a coach present. This suggests that players are likely to try harder when by themselves, which supports the effectiveness of an AI coach.

4.2. Experiment 2

Another potential flaw of an AI coach is the accuracy of the feedback on improvements for the player. The accuracy is crucial to the program, as it is required so that the users can steadily improve.

To test the accuracy of the AI's feedback, a control data set is first set up by horizontally swinging the racket with the device on at a slow pace. This action will then be repeated for at least ten times to create a more accurate data set. Then, the AI will be prompted to give feedback on the control data and follow the directions for improvement. Next, another set of data will be collected, but this time, it will be according to the AI's coaching. From the two data sets, a comparison can then be made to test the accuracy of the AI.

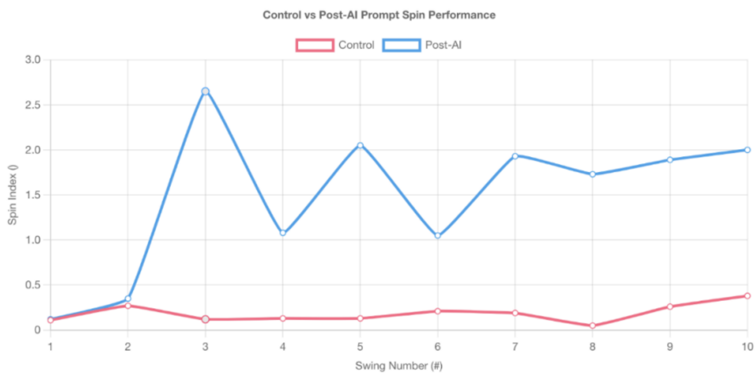

Figure 9. Control vs Post-AI Prompt Spin Performance

The collected data were analyzed by calculating descriptive statistics, including mean, median, minimum, and maximum values, to evaluate the effect of AI-guided feedback on spin performance.

The evaluation of the spin performance is based on the spin index, which is a number calculated by the device, and a higher number means better spin. The AI feedback was to engage the wrist more to achieve a better brushing motion. The average of the control data set was 0.185, while the average of the post-AI feedback data set was 1.485, showing a significant increase and that the entire data set shifted upward. The lowest control data was 0.05, and the highest was 0.38. In comparison, the lowest post-AI feedback data was 0.12, while the highest reached 2.65. This wide range does not suggest inaccuracy; it instead displays a slow yet existing improvement in the player's spin. These findings also suggest the biggest factor affecting the results was the improved wrist engagement encouraged by the AI feedback, which directly enhanced the brushing motion needed for higher spin.

5. RELATED WORK

A separate approach to the high costs of tennis lessons is documented in the paper "Towards a sustainable financial model for professional tennis players" [13]. This paper examines how many players, specifically aspiring professionals, are losing money as they participate in more competitive tournaments. The few solutions suggested by this paper can be summarized as reorganizing the professional tennis system so that it caters financially more to lower-ranked players. However, this solution requires the cooperation of the hundreds of tournaments and associations that make up the professional tennis system, which is highly improbable. MySwingTracker, on the other hand, can individually reduce coaching fees for players, which can hopefully allow players to be profitable and sustain their careers as tennis players for longer.
Another solution was proposed by Andrew Hawling in his research, "Electronic Tennis Officiating: Low Cost, Accurate and Reliable Solutions" [14]. Similar toMySwingTracker, it is a hardware device made for detection. Hawling's solution is a real-time electronic tennis officiating system designed to replace chair umpires and the currently used Hawk-Eye line-calling systems, so it aims to lower tournament fees by reducing the need to hire umpires and technicians. It is placed close to court lines to conduct line calling, but it risks extraneous movement, such as players running near the sensors, that could cause the system device to malfunction. Also, it does not help beginner players who do not even have the skill to enter tournaments. In comparison, MySwingTracker can help all tennis players as it addresses a universal issue in tennis: coaching costs.

Another proposal aimed to lower the expensive barrier to tennis was covered in Jacob G. Emery's paper "Helping Youth Athletes Thrive: An Exploratory Study of the Youth Tennis Market in New Zealand" [15]. This paper describes the problem that many youth players is that they get mentally unmotivated or stuck, but the existing solution of sports psychology is too expensive. Emery then presents a phone application that helps youth tennis players set goals for themselves to stay motivated. However, this motivational app does not get to the core of the problem of how tennis players get unmotivated because they do not have the resources to improve. MySwingTracker addresses this issue directly by helping players improve their foundational and technical skills so that they can continue to enjoy tennis by staying competitive.

6. CONCLUSIONS

A limitation of MySwingTracker is that it adds weight to the tennis racket, which may slightly alter swing feel and balance. For competitive players, even small changes in racket weight can affect timing and performance, potentially reducing long-term adoption. Additionally, the prototype was tested with a limited number of users, meaning results may not fully represent the broader tennis population. Battery life and sensor durability were also constraints, limiting extended on-court testing. If more time were available, the primary improvement would be reducing the device's size and weight. The location of the device can also be changed by making specialized case designs for different types of tennis rackets, as rackets mostly fall into either the category of head-heavy or handle-heavy. Further user testing across different skill levels would help refine data accuracy and usability. Software improvements, such as more intuitive data visualizations, would also enhance effectiveness. These changes would improve comfort, adaptation, and overall performance impact.

Overall, this project demonstrates the potential of wearable sports technology to enhance skill development in tennis. Although MySwingTracker is still a prototype, it highlights how thoughtfully designed technology can bridge the gap between training and performance. With further optimization in its hardware design, MySwingTracker could become a practical and valuable tool for players seeking to improve consistency, technique, and long-term performance outcomes.

REFERENCES

[1] Lake, Robert J."Social class, etiquette and behavioural restraint in British lawn tennis, 1870–1939." The International Journal of the History of Sport 28.6 (2011): 876-894.
[2] Thibaut, Erik, et al. "Time and money expenditure in sports participation: The role of income in consuming the most practiced sports activities in Flanders." Sport Management Review 20.5 (2017): 455-467.
[3] Reid, Machar, et al. "Skill acquisition in tennis: Research and current practice." Journal of science and medicine in sport 10.1 (2007): 1-10.
[4] Hamann, Donald L., and Robert S. Frost. "The effect of private lesson study on the practice habits and attitudes towards practicing of middle school and high school string students." Contributions to Music Education (2000): 71-93.
[5] Pluim, Babette M., et al. "Health benefits of tennis." British journal of sports medicine 41.11 (2007): 760-768.
[6] Bozyiğit, Elif. "Masters tennis participation in Turkiye: Women's masters." ITF Coaching & Sport Science Review 33.95 (2025): 46-50.
[7] Downward, Paul, and Simona Rasciute. "Does sport make you happy? An analysis of the well-being derived from sports participation." International review of applied economics 25.3 (2011): 331-348.
[8] Martin, Lorena. "Is socioeconomic status a contributing factor to tennis players' success." Journal of Medicine and Science in Tennis 20.3 (2015): 116-121.
[9] Johnson, Finn. "Unraveling the Net Worth: Exploring Financial Disparities and Diversity Deficits in Tennis."
[10] Tandon, Pooja S., et al. "Socioeconomic inequities in youth participation in physical activity and sports." International journal of environmental research and public health 18.13 (2021): 6946.
[11] Kollar, Melissa, and Zach Scherer. "Income in the United States: 2024." US Census Bureau Current Population Reports (2025): 3.
[12]Kilit, Bulent, et al. "Effect of coach encouragement on the psychophysiological and performance responses of young tennis players." International journal of environmental research and public health 16.18 (2019): 3467.
[13] Balliauw, Matteo, et al. Towards a sustainable financial model for professional tennis players. No. 2017006. 2017.
[14] Hawling, Andrew. Electronic tennis officiating: low cost, accurate and reliable solutions. Diss. Toronto Metropolitan University.
[15] Emery, Jacob G. Helping Youth Athletes Thrive: An Exploratory Study of the Youth Tennis Market in New Zealand. Diss. Open Access Te Herenga Waka-Victoria University of Wellington, 2019.