Tac-Trainer: A Visual Analytics System for IoT-based Racket Sports Training

Jiachen Wang, Ji Ma, Kangping Hu, Zheng Zhou, Hui Zhang, Xiao Xie, and Yingcai Wu

Fig. 1. The devices we used in this work. A is a customized ball with six markers. B is a ball machine assisting the multi-ball training. C is a high-speed camera for speed and spin annotation. D displays the mounting method of IoT devices on a trainee's body. The axis directions are displayed in yellow. The direction of Z axis in right arm, right wrist, and left wrist is perpendicular to the figure, from inside to outside. E is the circuit board of the devices in D.

Abstract— Conventional racket sports training highly relies on coaches' knowledge and experience, leading to biases in the guidance. To solve this problem, smart wearable devices based on Internet of Things technology (IoT) have been extensively investigated to support data-driven training. Considerable studies introduced methods to extract valuable information from the sensor data collected by IoT devices. However, the information cannot provide actionable insights for coaches due to the large data volume and high data dimensions. We proposed an IoT + VA framework, Tac-Trainer, to integrate the sensor data, the information, and coaches' knowledge to facilitate racket sports training. Tac-Trainer consists of four components: device configuration, data interpretation, training optimization, and result visualization. These components collect trainees' kinematic data through IoT devices, transform the data into attributes and indicators, generate training suggestions, and provide an interactive visualization interface for exploration, respectively. We further discuss new research opportunities and challenges inspired by our work from two perspectives, VA for IoT and IoT for VA.

Index Terms—IoT, racket sports, training, sensor data, visual analytics

1 Introduction

Racket sports, such as table tennis, badminton, tennis, etc., are technique-centric sports that require players to coordinate their bodies to use different strengths, speeds, or spins to hit the ball [77]. Therefore, during the training, coaches are most concerned with trainees' motions when they hit the ball. Conventionally, coaches monitor trainees' motions by eyes or through videos and give suggestions based on their experience and knowledge during the training. However, coaches can only observe trainees' motions from a macroscopic perspective, lacking microscopic details. For example, they can observe that a trainee swings the racket, but they cannot know the exact speed and angle of the racket. To solve this problem, smart wearable devices based on the Internet of Things (**IoT**) technology have been widely investigated to provide a data-driven method to alleviate the biases caused by coaches' knowledge and experience [10, 15, 77].

Current IoT devices collect trainees' training data by using inertial sensors, which are attached to the trainees' bodies to record the acceleration and velocity during their movements. Fig. 2 presents an example of the data collected by a 9-axis inertial sensor in 2 seconds. Considerable studies have introduced various methods to extract valuable information such as trainees' skill levels [9, 16], technical attributes of actions [53, 83], etc., from the sensor data. However, given the large data volume and various dimensions, it is still hard for users to obtain actionable insights for the training from the information. A visual analytics system that integrates the sensor data, the information, and coaches' knowledge for training analysis is urgently needed. Therefore, we propose an IoT + visual analytics (VA) framework, Tac-Trainer, to facilitate racket sports training.

We encountered two challenges when developing Tac-Trainer. The first is how to coordinate the sensor data and the information extracted by models for coaches. The sensor data contains the most comprehensive details, but it is unintelligible to coaches. The extracted information is straightforward, yet it loses some details during the processing of the models. Integrating the advantages of these two types of data for coaches is a challenge. The second is how to provide effective and efficient analysis for coaches. Coaches pay special attention to the timeliness of training feedback. They prefer to discover the trainees' problems as soon as possible to adjust their training plans in time. Once the problems are found late, both the coaches and trainees may forget the details of the problems. However, the large volume and the high dimension of the data are barriers to efficient analysis.

We collaborated with three experts: a professor and a postdoctoral researcher analyzing table tennis and a Ph.D. candidate majoring in tennis to develop Tac-Trainer. All experts were professional players, and the professor and the researcher are experienced coaches and have served as data analysts for the Chinese national table tennis team for years. To solve the first challenge, we conducted interviews with our experts and summarized their requirements and considerations about the required data for analysis. To solve the second challenge, we proposed an IoT+VA framework, Tac-Trainer, for training. Tac-Trainer contains four components (Fig. 3): device configuration, data interpretation, training

- J. Wang, J. Ma, K. Hu, X. Xie, and Y. Wu are with the State Key Lab of CAD&CG, Zhejiang University. X. Xie and Y. Wu are also with the Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies. E-mail: {wangjiachen, zjumaji, hukp, xxie, ycwu}@zju.edu.cn. Y. Wu and X. Xie are the co-corresponding authors.
- Z. Zhou and H.Zhang are with the Department of Sport Science, Zhejiang University. E-mail: {zheng.zhou, zhang_hui}@zju.edu.cn

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

optimization, and result visualization. Data configuration configures the IoT devices for data collection. Data interpretation transforms the sensor data collected by IoT devices into semantic training information, including the technical attributes and performance indicators of actions. Training optimization helps coaches assess trainees' actions based on the semantic training information and generates optimization suggestions for the poorly-performed actions. Result visualization visualizes the process and results of data interpretation and training optimization to facilitate data exploration and interpretation in real-time. When developing Tac-Trainer, we found several new research directions in IoT + VA. We discussed the opportunities and challenges in these directions. The contributions of this work are as follows:

- We combined IoT devices and visual analytics technology to solve a domain problem. Our design study can be applied to solve problems in other domains.
- We established Tac-Trainer, an IoT + VA framework for racket sports training.
- We implemented a proof-of-concept system for table tennis training based on Tac-Trainer and conducted two use cases.

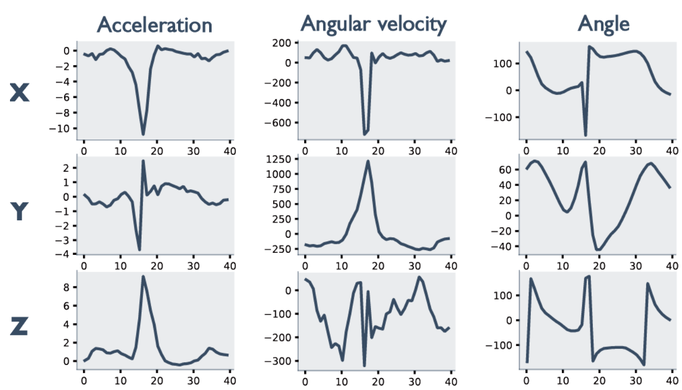

Fig. 2. The data collected by an IoT device fixed on a trainee's racket when he performed a stroke in 2 seconds.

2 RELATED WORK

In this section, we first review related works of IoT-based sports training. Then, since our data is similar to that of motion analysis and related to sports analysis, we further review the motion data visualization and sports data visualization.

2.1 IoT-based Sports Training

IoT technology has been widely applied to the training of diverse sports. For example, miPod [17] and miPod2 [15] are portable sensor systems that can be embedded into clothes and sports equipment to collect players' motion data. In addition, Xia et al. [83] proposed a wristband system to detect and recognize strokes in multiple racket sports, including badminton and table tennis. Moreover, similar systems are developed for particular sports including soccer [55], badminton [77], skateboarding [36], etc.. With the systems, researchers can collect the kinematic details of players' motions for analysis.

A popular problem in analyzing such data is action recognition. Lu et al. [55] used an SVM-based classifier to recognize the passing and shooting of soccer players. Besides, Blank et al. [16] also used SVM to classify different stroke types in table tennis. Similar analyses are also conducted in other sports, such as basketball [56] and golf [58]. Another frequently studied topic is skill assessment. Wang et al. [77] employed a PCA+SVM method to differentiate the skill of elite and amateurs. Ahmadi et al. [10] assessed the skill level of different players by plotting their data in a three-dimensional coordinate system. Other topics, such as jump frequency estimation in volleyball [40] and speed and spin estimation in table tennis [14] are also investigated by many researchers. However, the valuable information extracted by these studies cannot provide actionable insights for coaches due to the large data volume and various dimensions.

Currently, considerable commercial smart devices have been introduced for the training of sports, such as table tennis [1,3], badminton [5], tennis [2,4,6], etc. These devices are either mounted or embedded into the equipment to collect trainees' kinematic data. With the data, they present various statics, such as number of shots, ball impact location, speed of swing, etc., by using basic visualization charts. Although these products can provide straightforward training feedback for trainees, they cannot help discover valuable training insights since they only present the basic statistical results. The fine-grained training process is not available for further exploration and investigation. To solve these problems, we took racket sports as an example and propose an IoT + VA framework named Tac-Trainer. Tac-Trainer can integrate the sensor data and the information extracted by current studies for coaches to facilitate visual analytics of IoT-based training. We referred to the related studies on racket sports and implemented Tac-Trainer in the domain of table tennis.

2.2 Motion Data Visualization

Analysis of the motions of moving objects has been extensively studied [23, 33, 41, 46, 57, 78]. There are considerable approaches to the visualization of motion data. The survey by Bernard et al. [12] has provided a detailed review of related studies. It classified existing methods from three aspects that affect the design of visualizations, namely, the granularity of data, the scope of objects, and the concept to be analyzed. These methods solve various domain problems. Krekel et al. [47] combined 3D skeletons with 2D plots to visualize the motions of multiple joints of the upper extremity. FuryExplorer [79] visualizes horse motions based on three granularities, trajectories, poses, and markers on horses. Motion Browser [22] integrates and visualizes muscle signals, motion data, and videos to help physicians analyze brachial plexus injuries. Other studies such as MotionExplorer [13], MotionFlow [39], GestureAnalyzer [38] also provide efficient tools to investigate kinematic features of motions. However, methods for analyzing the motion data in sports training are rarely studied. Although our task to investigate the characteristics of players' motions is similar to existing methods, our goal is to help coaches obtain training feedback and adjust training plans efficiently, which existing methods cannot support.

2.3 Sports Data Visualization

Sports data visualization has been a popular topic in recent years. Several surveys [29, 66, 87] have provided a holistic view of the state-of-the-art methods for match analysis in various sports. For example, in soccer, researchers introduce efficient tools to investigate team tactics (e.g., formations [82], pass styles [85], and moving trajectories [11]), rankings [63, 65], game videos [71, 72], key events [20, 64], and match performances [70]. Besides, another popular sport, basketball, has also been widely studied. The spatial characteristics of players are often analyzed when investigating shooting abilities [31], defense abilities [30], movement patterns [73], and point prediction [21] in basketball matches. Losada et al. [54] introduced a comprehensive match analysis tool and Chen et al. [25] and Zhi et al. [89] developed storytelling tools for basketball. In other sports, such as baseball [60], rugby [37], badminton [28, 88], tennis [68, 69], table tennis [26, 49, 50, 75, 76, 81], ice hockey [67], snooker [61], etc., various tools are also proposed to facilitate technical and tactical analysis. Insights into players' performances, team strategies, tactic effects, etc., during matches, are revealed with these tools. Readers can refer to the surveys for details. However, these studies cannot directly solve our problems since they target the analysis of formal matches instead of the training process. Our data and requirements are different from theirs. Specifically, our data is collected by inertial sensors, whereas match data is usually collected by cameras with automatic (computer vision algorithms) or manual (video labeling) methods. Moreover, coaches are concerned about how to improve the quality of particular techniques or tactics when analyzing the training process. In contrast, in match analysis, analysts concentrate on how to employ different techniques and tactics to win a game.

In addition to match analysis, topics about personal visualization for sports activities [52, 86] have also been widely investigated. These studies explore the design space of activity tracking products, such as Apple watch [1] and Fitbit [2]. However, similar to the application of smart devices [1–6], the visualizations of the products only present basic statistics, such as heartbeat rate and calories. The design space and methods of these visualizations cannot be applied to our condition.

Wu et al. [80] combined virtual reality (VR) devices with a haptic feedback racket to provide intuitive cues for trainees in table tennis. In addition, Oagaz et al. [59] also introduced a VR system for table tennis training. The system integrates a VR helmet with a depth camera to create an immersive training environment. However, these systems are designed for helping beginners acquire new skills, which is different from our goals to improve professional players' performance. Besides, we target a real-world training instead of a virtual environment. Therefore, we referred to prior studies and proposed an IoT + VA framework, Tac-Trainer, for the IoT-based training in racket sports.

3 BACKGROUND

This section introduces related knowledge about racket sports and our interviews with experts.

3.1 Racket Sports

Racket sports (e.g., table tennis, badminton, tennis) are popular around the world, with hundreds of millions of participants [7]. In racket sports, two (or four for doubles) players use rackets to hit a ball to each other [51]. Each hit is called a stroke, which is the elementary observation unit in racket sports.

3.2 Expert Interview

We collaborated with three experts, namely, a professor (E1), a postdoctoral researcher (E2), and a Ph.D. candidate (E3). E1 and E2 have studied table tennis for more than 18 and 5 years, respectively. E3 has majored in tennis analysis for more than three years. All of them were professional players. Besides, E1 and E2 are experienced coaches and have provided data service for the Chinese national table tennis team for more than five years. We conducted one-on-one interviews with them. The interviews are summarized as follows.

Multi-ball training occupies the most time in training. We first asked our experts about the content of regular training. According to E1 and E2, the regular training often consists of physical training (*20%*), single-ball training (*35%*), and **multi-ball training (*45%*)**, which is similar to tennis. Physical training often contains some machine exercise. Single-ball training requires a trainee to play against one or two opponents. Multi-ball training requires a trainee to hit back the ball with particular techniques continuously. The ball will be served to the trainee one by one without a break. This part aims to improve players' mastery of particular techniques [32]. According to our experts, it occupies the most time during the training since mastering different techniques is the foundation of successful tactics and strategies. Therefore, we mainly focus on multi-ball training.

Speed, spin, and placement of the ball are the most important factors. Then, we collected the most important factors they care about during multi-ball training. According to E1 and E2, speed, spin, and placement of the ball play important roles when assessing a trainee's stroke. These three factors differentiate techniques in table tennis [84], leading to different effects of a stroke. Among these factors, spin is the most important since it can control the ball trajectories due to the Magnus effect [48]. Similarly, in tennis, E3 indicated that coaches often focus on the speed and the placement of the ball. They also pay attention to the spin when trainees practice special techniques, such as slice and drive. If trainees' strokes do not meet the requirements in these three factors, coaches will give suggestions on their motions.

Motions of upper extremities are critical. We further asked about coaches' suggestions of trainees' motions. E1, E2, and E3 all said that they paid more attention to the motions of trainees' upper extremities, especially the wrists and arms [35]. Although a player's footwork was also important, the decisive motions affecting the stroke quality were those of the upper extremities. For example, in table tennis, if a trainee raises his/her elbow too high when hitting the ball, the ball will be hit out of the table. Similarly, in tennis, a wrong motion of the wrist can not only affect the stroke performance but also hurt the wrist. Therefore, during the training, coaches often help trainees to correct their motions of the upper extremity.

Specific speed and spin and accurate suggestions are needed. After that, we collected two pain points during the multi-ball training. First, they mentioned the acquisition of the specific speed and spin of trainees' strokes. They told us that the placement is easy to evaluate since they can know whether the ball is hit to the right placement by their eyes. However, the speed and spin are difficult to judge. They can distinguish the high-speed strokes from the low-speed ones and the high-spin strokes from the low-spin ones, but they cannot tell the exact speed and spin. Second, they mentioned that their suggestions to trainees were based on their experience and knowledge, which did not always work. They can only observe a trainee's motions with their eyes, lacking the accurate quantitative details. For example, they observed that a trainee swung the racket too low, but they did not know the specific height and angle of the racket trajectory, which hindered them from giving accurate suggestions to the trainee.

Intuitive tools should be proposed. At the end of the interviews, we introduced several IoT-based training methods for them. We told them that these training methods could solve the problem of quantitatively assessing speed and spin. They presented much interest in the methods, but the complex data, algorithms, and models are difficult for them to apply to practical training. They hoped that an intuitive tool that coordinates all the data and methods for them to conduct efficient training analysis could be developed.

4 FRAMEWORK

This section introduces the design considerations and the details of Tac-Trainer.

4.1 Design Considerations

We summarized five design considerations for Tac-Trainer and iterated the considerations with our experts [8]. The details are as follows.

- **C1: Configuring appropriate devices.** Configurations such as parameters and mounting positions of devices should be carefully considered since these factors can influence the quality of the collected data. For example, if the mounting positions are wrongly chosen, the kinematic features of particular actions cannot be obviously reflected in the data.

- **C2: Extracting meaningful data.** Extracting the meaningful motion data from the raw sensor data is essential for further analysis. The raw sensor data records all details of players' motions. It inevitably contains some meaningless information, such as motions during warm-up. We need to remove these useless data by detecting and extracting the data of meaningful motions such as strokes in racket sports.

- **C3: Recognizing technical attributes.** The extracted data is still too abstract for coaches to understand. We should recognize the technical attributes of the actions based on the extracted data. For example, during multi-ball training of table tennis, the technique (e.g., topspin, push, etc.) and position (forehand, backhand, etc.) of each stroke should be recognized. With the technical attributes, coaches can understand the basic meaning of the sensor data.

- **C4: Estimating important indicators.** Only recognizing the technical actions is not enough for training analysis. According to the interviews, factors such as speed and spin of a stroke are important indicators for coaches to assess training quality. Quantitatively estimating these indicators can help coaches provide accurate suggestions for trainees, solving a pain point for coaches.

- **C5: Assessing training quality.** Multi-ball training requires trainees to hit the ball continuously with high frequency. Coaches cannot assess each stroke efficiently even they know the quantitative value of the speed and spin of the stroke. Therefore, we should provide methods to help assess each stroke efficiently so that coaches and trainees can obtain feedback in time.

[1] https://www.apple.com/watch/
[2] https://www.fitbit.com/global/hk/products

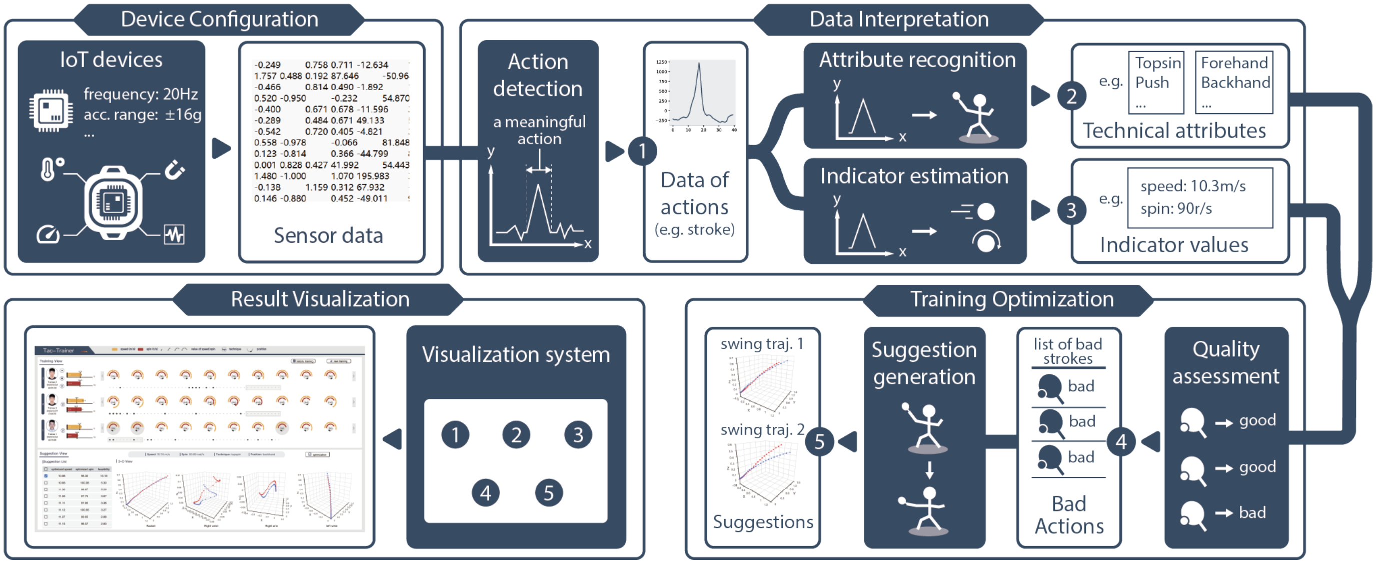

Fig. 3. The overview of Tac-Trainer. The whole framework contains four components: Device Configuration, Data Interpretation, Training Optimization, and Result Visualization. Device configuration configures IoT devices to collect the required data adequately. Data interpretation takes the sensor data as input. It detects and extracts the meaningful actions (1), recognizes the technical attributes (2), and estimates the values of the key indicators (3). With these results, trainin optimization first assesses the quality of each action (4) and then generates suggestions for the bad actions (5). Result visualization is responsible for visualizing all processes and results (1, 2, 3, 4, 5).

Size	Weight	Battery	Transmission frequency	Acceleration range	Angular velocity range	Angle range	Angular accuracy
51mm*36mm *15mm	20g	3.7V-260mAh	UDP: 1~200Hz	±16g	±2000°/S	X, Z: ±180° Y: ±90°	Static: 0.05° Dynamic: 0.1°

Fig. 4. The parameters of the device (WT901WIFI) we use. The table presents eight important parameters. Readers can visit the website: *https://www.wit-motion.com/iot-gyroscope/witmotion-wt901wifi-wireless.html* for more details.

- **C6: Generating data-driven suggestions.** According to the interviews, coaches often provide knowledge-driven suggestions for trainees. However, such suggestions do not always work and sometimes are biased. Therefore, we should provide data-driven suggestions for coaches. By integrating knowledge and data, coaches can provide more effective suggestions for trainees.

- **C7: Designing intuitive visualizations.** Visualization is an efficient and effective way to bridge the gap between complex data and users. Visualization of the results of **C2**-**C6** can facilitate coaches' interpretation and application of IoT-based training methods. However, an issue needs to be carefully solved: the visualizations should be intuitive for coaches. Basic charts and metaphor-based design are good choices for this issue.

4.2 Framework

We proposed Tac-Trainer based on the seven considerations. The framework contains four components, data interpretation, training optimization, and result visualization, as Fig. 3 shows.

Device Configuration configures the IoT devices for data collection. It mainly considers the sensors (e.g., gyroscope, magnetic sensor), the parameters (e.g., frequency, size, precision, etc.), the number, and the mounting position (e.g., wrist, arm, leg, etc.) (**C1**).

Data Interpretation takes the raw sensor data as input and processes the data in three steps. First, it cleans the data by detecting and extracting the data of meaningful actions such as strokes in table tennis (**C2**). The cleaned action data (Fig. 3(1)) is fed to attribute recognition (**C3**) and indicator estimation (**C4**). These two steps output the critical technical attributes (Fig. 3(2)) and indicators (Fig. 3(3)). For example, in table tennis, these two parts output attributes such as stroke technique and indicators such as stroke speed.

Training Optimization contains two steps: quality assessment (**C5**) and suggestion generation (**C6**). First, it assesses the quality of each stroke based on the technical attributes and indicator values and detects poorly-performed strokes (Fig. 3(4)).Then, it generates optimization suggestions (Fig. 3(5)) for the poorly-performed strokes. Coaches can refer to these suggestions to improve trainees' performances.

Result Visualization is responsible for visualizing (**C7**) the results of data interpretation and training optimization (Fig. 3(1, 2, 3, 4, 5)). Moreover, it also provides interactions to help coaches explore and integrate their knowledge into the analysis process.

5 Implementation

This section illustrates the implementation of Tac-Trainer in table tennis. Device configuration, data interpretation, and training optimization are introduced first. Then, we present the system of result visualization.

5.1 Device Configuration

We used existing IoT devices, *WT901WIFI* [3] developed by *WitMotion* [4] (Fig. 1E) to collect players' motion data during their training. Each device is portable enough (size: $51mm*36mm*15mm$, weight: $20g$) so that the influence of devices on players' performance is decreased as much as possible. Each device contains a 9-axis gyroscope collecting the three-dimensional acceleration, angular velocity, and angular of a moving object as shown in Fig. 2. It can transmit the collected data through WiFi at up to $200Hz$. During our implementation, we set the frequency to $20Hz$ since $20Hz$ was enough to collect trainees' detailed kinematic features. Besides, with a built-in battery, it can continuously work for two hours. The detailed parameters are shown in Fig. 4.

[3]https://www.wit-motion.com/iot-gyroscope/witmotion-wt901wifi-wireless.html

[4]https://www.wit-motion.com/

During the training, we used **four** such devices and fixed them on the trainee's **right arm**, **right wrist**, **left wrist**, and **racket handle**, respectively. The axis directions are shown in Fig. 1D and remain consistent in this work. We fixed devices on trainees' upper extremities given the interview with our experts. Moreover, we chose the four points by trial and error. In the beginning, we thought one device on the racket was enough. However, when training the data processing models, we found using data of multiple devices can improve the model performances. Therefore, we finally decided to use four sensors.

Table 1. The performance of the stroke detection model.

Precision	Recall	Accuracy	F_1 Score
0.998	0.998	0.995	0.998

Table 2. The accuracy of different models in attribute recognition.

		LSTM	RF	DF21	XGBoost	LightGBM
Position	Acc. (%)	**100**	**100**	**100**	99.67	**100**
	Time. (s)	53.8	**2.4**	18.8	6.9	8.2
Technique	Acc. (%)	99.67	**100**	**100**	99.33	**100**
	Time. (s)	54.7	**1.8**	19.3	2.8	3.6

5.2 Data Interpretation

In table tennis, data interpretation detects and extracts valid strokes, recognizes stroke attributes, and estimates stroke speed and stroke spin based on the raw sensor data. To estimate the stroke speed and spin, we used additional auxiliary devices for data annotation.

5.2.1 Auxiliary Devices

The auxiliary devices for data annotation include multiple customized balls (Fig. 1A), a ball machine (Fig. 1B), and a high-speed camera (Fig. 1C) as follows. These devices were not used in the final system.

- **Customized ball.** We drew six markers on each ball (i.e., top, bottom, left, right, back, and front) as Fig. 1A presents. The markers can assist the high-speed camera (Fig. 1C) in measuring the speed and spin of the ball. The measured data worked as ground truth to train the model for estimating speed and spin.
- **Ball machine.** We used a ball machine to conduct multi-ball training Fig. 1B. The ball machine can serve the ball with various speeds, spins, and placements. The speed can be set up to $5m/s$. The spin can be set up to $50r/s$ (both backspin or topspin). The placement can cover all of the half table. We used a remote controller to control the type of strokes it served.
- **High-speed camera.** We used a high-speed camera to measure the ground truth of the speed and spin of the ball. According to Blank et al. [14], the frame rate of the high-speed camera should be higher than $500Hz$. Otherwise, the ball would be blurry, hindering the measurement. Therefore, we selected a camera whose frame rate was $1000Hz$ and resolution was $640*480$ pixels. The captured video was black and white.

5.2.2 Stroke Detection and Extraction

We employed a simplified version of the method of Blank et al. [16] to detect the valid strokes. Readers can refer to the paper for more details. The raw sensor data consists of multi-dimensional discrete signals (Fig. 2). Therefore, according to Blank et al. [16], detecting strokes equals detecting the peaks P_i in signals. We used acceleration signals of the racket to detect peaks since explicit peaks existed in all axes of the acceleration signals (Fig. 2). The detection process contains two steps: energy computation and peak detection. First, we calculated the energy of the acceleration signal at t as follows,

$$E(t) = acc_x(t)^2 + acc_y(t)^2 + acc_z(t)^2 \quad (1)$$

where $acc_x(t), acc_y(t), acc_z(t)$ represent the signal value of X-axis, Y-axis, and Z-axis at t. After this step, the peaks were highlighted and more obvious than before. Second, we employed a peak detection algorithm to find the peaks in the signals. Here, we used the *signal.find_peaks* function in the python package, *SciPy* [5]. We did not use a high-pass Butterworth filter to amplify the peaks further [19] as Blank et al. [16] did because the peaks were explicit enough to be detected after the first step. Therefore we removed the filter to save time. We used 845 labeled strokes to test the performance of the stroke detection method. During the test, the method performed well, with only two strokes missed and two strokes wrongly detected. The precision, recall, accuracy, and F_1 score of the test are presented in Table. 1.

With the detected peaks $P = \{P_1, ..., P_n\}$, we extracted the data of all strokes $S = \{S_1, ..., S_n\}$ by setting a range for each peak. Specifically, for the i^{th} peak P_i, the range was $[t_{P_i} - \delta t, t_{P_i} + \delta t]$, where t_{P_i} was the timestamp of P_i and δt was a customized duration. We set δt to $0.75s$ which was longer than that of Blank et al. [16] since we wanted to save enough data to estimate the speed and spin accurately. In this way, the signal of each axis of the i^{th} stroke S_i contains 30 sampling points ($20Hz*(0.75s+0.75s) = 30points$), and the sensor data of S_i can be depicted by a tensor V_i, where $V_i \in \mathbb{R}^{4(devices)\times 9(dimensions)\times 30(points)}$.

5.2.3 Attribute Recognition

We mainly recognized two kinds of technical attributes: stroke position and stroke technique in table tennis. Stroke position represents a player's position when he/she hits the ball. Stroke technique represents the technique a player uses to hit the ball. Here, we focused on the two most common stroke positions (i.e., forehand and backhand) and the three most common stroke techniques (i.e., topspin, push, and short) during training. To accurately recognize the attributes, we tried five state-of-the-art models: LSTM [34] (we added a full connected layer to an LSTM network), DF21 [91], random forest [62], XGBoost [24], and LightGBM [43]. The input was V_i, and the label was the corresponding stroke position Pos_i and stroke technique Tec_i of S_i as follows,

$$F_r(V_i) = (Pos_i, Tec_i) \quad (2)$$

where F_r was the recognition model. Stroke position and stroke technique rarely change during training. Therefore, it can be easily labeled for model training. We finally labeled 100 strokes for each technique and position. We used 10-fold cross-validation to evaluate the performances of different models. Table 2 presents the accuracy and training time of each model. Although the accuracies of all models were 100%, except XGBoost [24], the training time of random forest [62] outperformed others. Therefore, we finally chose random forest [62] as the model for attribute recognition.

5.2.4 Speed & Spin Estimation

We estimated the speed and spin of the ball since these are significant indicators for coaches, according to our experts. Previously, Blank et al. [14] created a well-established physical model to estimate these two indicators. However, this method is sensitive to physical coefficients. For example, if the friction coefficient and restitution coefficient of the racket rubber is not measured accurately, the error of the results can be large. These coefficients need to be measured manually by analysts, which inevitably introduces deviations to these coefficients. Therefore, we did not refer to the physical models. Instead, we constructed a regression model to estimate the speed and spin based on the sensor data. After training, the model can estimate the indicators without additional coefficients.

The input of the model was V_i and the output was the speed Spd_i and spin Spn_i of S_i as follow,

$$F_e(V_i) = (Spd_i, Spn_i) \quad (3)$$

where F_e was the estimation model. We labeled the speed and spin of each stroke based on the customized ball (Fig. 1A) and high-speed camera (Fig. 1C). The camera captured the detailed movement of markers on the ball. With the videos, we developed an annotation tool to

[5] https://scipy.org/

label the speed and spin. We finally labeled 100 strokes for topspin, push, and short, respectively. We labeled strokes of different techniques because there is a great difference in the speed and spin among different techniques, as Table. 3 shows. We tried the same five models as in attribute recognition. We used mean absolute percentage error to evaluate the performance of each model. The detailed evaluation results are presented in Table. 4. We found that random forest [62] has the smallest error in both speed estimation (8.6%) and spin estimation (7.21%), outperforming other models. Therefore, we finally chose random forest [62] as the regression model. We interviewed our experts about the acceptance of the error. The experts said that they could accept such error that is less than 10% since what they wanted to do is not to tell $10.1m/s$ from $10.2m/s$, but to tell $10m/s$ from $11m/s$.

Table 3. The statistics of speed and spin of different techniques.

		Topspin	Push	Short	All
Speed	Avg. (m/s)	11.13	6.19	3.83	7.05
	Std. (m/s)	0.98	1.52	0.38	3.22
Spin	Avg. (r/s)	106.64	50.80	31.01	62.82
	Std. (r/s)	6.44	10.77	3.20	32.89

Table 4. The estimation error of speed and spin.

		LSTM	RF	DF21	XGBoost	LightGBM
Speed	Avg. (%)	9.37	**8.60**	8.80	9.69	9.20
	Std. (%)	1.64	**1.50**	1.80	1.85	1.93
Spin	Avg. (%)	7.74	**7.21**	7.66	7.69	7.91
	Std. (%)	1.34	1.42	1.33	1.83	**1.21**

5.3 Training Optimization

Training optimization helps coaches assess the quality of each stroke and generate optimization suggestions.

5.3.1 Quality Assessment

Coaches' knowledge is important when assessing different types of strokes. However, coaches' knowledge is too complex to be precisely quantified for automatic assessment. Therefore, we provided an interactive method for coaches to integrate the aforementioned information and their knowledge to find the poorly-performed strokes as follows,

$$F_a(F_r(V), F_e(V), K) = pS \tag{4}$$

where F_a was our assessment method, $V = \{V_1, ..., V_n\}$ was the set of the sensor data of all strokes S, K was coaches knowledge, and $pS = \{pS_1, ..., pS_m\}$ was the set of all poorly-performed strokes. The details of the assessment method are introduced in Section 5.4.2.

5.3.2 Suggestion Generation

We referred to the concept of counterfactual to generate suggestions for training optimization because counterfactual can provide human-friendly explanations for machine learning models [74]. It can generate data instances that satisfy a desirable model prediction. We applied the method of Cheng et al. [27] to suggestion generation. The input was the sensor data of a poorly-performed stroke pS_q ($pS_q \in pS$). We used pV_q ($pV_q \in \mathbb{R}^{4(devices) \times 9(dimensions) \times 30(points)}$) to denote the sensor data of pS_q. During the generation, the method would try to optimize the elements of pV_q to make pS_q be assessed as a good stroke by F_a. The fewer elements revised, the better an optimization was. The final output was a set $oV_q = \{oV_q^1, ..., oV_q^r\}$ that contained the top r best optimizations the method has tried. The function was as follows,

$$F_s(pV_q) = oV_q \tag{5}$$

where F_s was the counterfactual function. We used oV_q^s ($oV_q^s \in \{oV_q^1, ..., oV_q^r\}$) to denote an optimization. oV_q^s is an optimized result of pV_q, with only several elements different from pV_q. It is in the same form of pV_q and works as a suggestion to tell how to adjust the motions to improve stroke qualities. To make oV_q^s intuitive for coaches, we visualized oV_q^s (Fig. 6J) to help coaches understand the suggestions. We introduced the visualization design in Section 5.4.3. Moreover, we calculated a feasibility value for each suggestion. Specifically, we used the reciprocal of the distance between a pV_q and oV_q^s to measure the feasibility of oV_q^s as follows,

$$F_f(oV_q^s) = \frac{1}{Dist(pV_q, oV_q^s)} \tag{6}$$

where $Dist()$ was the function that calculated the Euclidean distance of two tensors. In this way, if oV_q^s is similar to pV_q, $F_f(oV_q^s)$ will be large, which naturally means the feasibility of oV_q^s is high.

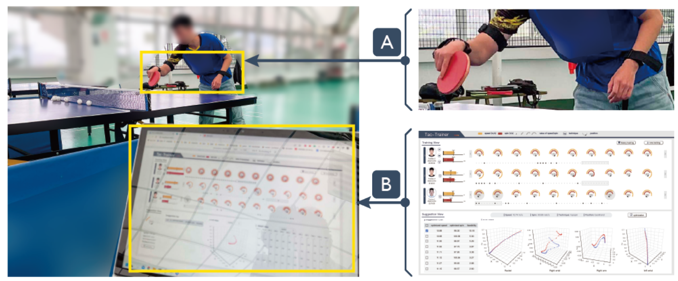

Fig. 5. The usage scenario of Tac-Trainer in case 1. During the training, a trainee needs to wear the devices (A), and the system visualizes the data from the devices in real-time (B)

5.4 Result Visualization

Result visualization visualizes all the processes and results of data interpretation and training optimization. The visualization system can help coaches efficiently monitor and adjust the training process.

5.4.1 System Overview

The whole system contains two views, a training view and a suggestion view (Fig. 6). The training view visualizes the recognized technical attributes and the estimated speed and spin of each stroke. Moreover, coaches can interactively assess the quality of each stroke in this view. If the coach finds a poorly-performed stroke, he/she can select it for optimization. After selection, the suggestion view visualizes multiple optimization suggestions for the stroke. Coaches can evaluate the feasibility of each suggestion and choose the best one. We used *React.js* to develop the frontend and Python to develop the backend.

During the training, a trainee needs to wear the IoT devices as required (Fig. 5A) and a coach needs to connect the system with the devices through Wi-Fi. After switching on the devices, the system will immediately receive the data from the devices. With the data, the system will automatically recognize the attributes and estimate the indicators. The results of the attributes and indicators will be simultaneously visualized in the system (Fig. 5B). In this way, the coach can monitor the trainee's performance in real-time and provide feedback for the trainee as soon as possible.

5.4.2 Training View

The training view visualizes the detected strokes and their technical attributes and indicators in the form of a flow. Each flow presents all strokes within a multi-ball training (Fig. 6C). We use a glyph to encode each stroke in a flow. The detailed encoding is as follows.

- **Speed & spin:** We used the metaphor of a dashboard to encode the speed and spin of a stroke (Fig. 7) since both indicators are related to the concept of velocity. The arc length encodes the value of the speed/spin. We used yellow to represent speed and red to represent spin. This color encoding is unified in the system.

Fig. 6. The interface of Tac-Trainer for table tennis. The system contains two views, a training view, and a suggestion view. The training view visualizes the strokes in a training session through a customized flow (C) consisting of a metadata panel (A), a control panel (B), and a stroke flow (F). The suggestion view provides a list of optimization suggestions (J) for a poorly-performed stroke (G). Each suggestion can be explored in a 3-D coordinate (K). The interface presents the details of Case 1. E is the first training session and D is the second. G is the stroke chosen for optimization. L is the optimization suggestion chosen by the coach.

- **Stroke technique:** There are fourteen stroke techniques in table tennis (e.g., topspin, push, etc.). According to Wang et al. [75], using identity channels to encode this information is difficult since it contains too many categories. Therefore, we referred to their works and encoded techniques by their abbreviation (Fig. 7B).
- **Stroke position:** There are four stroke positions in table tennis, namely, forehand, backhand, pivot, and back turn. They represent players' poses when hitting the ball and players' relative positions to the table. Therefore, we used position channels to encode them. As Fig. 7C shows, we used four arcs to encode the four stroke positions. The highlighted arc represents the stroke position of a particular stroke, which is similar to Wang et al. (Fig. 7).

The whole flow consists of three parts: a metadata panel (Fig. 6A), a control panel (Fig. 6B), and a stroke flow (Fig.6F).

The metadata panel (Fig. 6A) displays the basic information of a training session. It includes the training ID and the avatar of the trainee. The control panel (Fig. 6B) controls the start and end of a training session. Moreover, it uses bar charts to present the average and error bar to present the standard error of the speed and spin within a training session. With this information, coaches can easily assess the quality of strokes. They can set a threshold on the bar chart to find the poorly-performed strokes efficiently. The stroke flow displays the strokes within a training session (Fig. 6F). Before the training, the flow is empty. Once the IoT devices are switched on, and trainees start to hit the ball, the number of glyphs in the flow will increase progressively (Fig.6F). Due to the limited space of the stroke flow, we cannot display all strokes within a training session at the same time. Therefore, we placed a row of points at the bottom of the flow as an overview (Fig.6E). Each point corresponds to the glyph of a stroke (Fig.6G).

Interaction: We provide two kinds of interactions as follows.

- **Creating a new training session:** Coaches need to click the "new training" button at the upper right corner of the training view

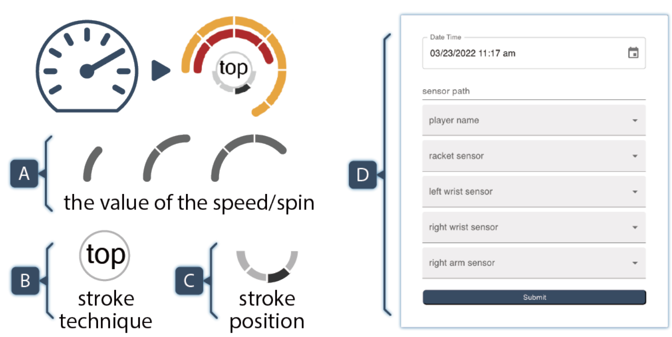

Fig. 7. The encoding of the glyph for a stroke and the form of creating a new training session. A, B, C presents the encoding of stroke speed/spin, stroke technique, and stroke position, respectively. E is the form coaches need to fill in before creating a new training session.

to create a blank flow for a new training session (Fig. 6I). Then, they need to finish a form as Fig. 7D shows. They need to fill the metadata of the training (i.e., trainee name, training time, and data path) and bind the sensors with the trainee's body and the racket. With the form completed, a new blank flow will be created in the training view. Then, coaches can click the start button (upper left in Fig. 6B) to enable the flow to visualize new strokes in real-time. If a training session is finished, coaches can click the stop button (middle left in Fig. 6B), and the flow will stop visualizing new strokes. In addition, coaches can also load the history training for analysis by clicking the "history training" button (Fig. 6H).

- **Assessing strokes:** Coaches can drag the slider on the bar chart to set the threshold for quality assessment. After dragging the slider to an ideal position, they can click the assessment button (lower left in Fig. 6B), and the glyphs of the strokes whose speed/spin are lower than the threshold will be highlighted in the flow (Fig. 6G). Besides, the corresponding points will also be highlighted in black. Coaches can easily find poorly-performed strokes.

5.4.3 Suggestion View

The suggestion view contains two sub-views: a suggestion list (Fig. 6J) and a 3-D view (Fig. 6K). The attributes and indicators of the selected poorly-performed stroke are presented at the top of the suggestion view as reference (Fig. 6M). Coaches can click the "Optimization" button to generate suggestions for the stroke (Fig. 6N). The suggestion list displays the top eight optimization suggestions $oV_q = \{oV_q^1, ..., oV_q^8\}$ (Fig. 6J). Each row presents the expected speed and spin of each suggestion oV_q^s, which is assessed by F_a. Moreover, the feasibility (Eq. 6) of each suggestion is also provided. Coaches can choose the suggestion they want according to these three indicators. Once coaches select a suggestion, the trajectories reconstructed based on the suggestion will be displayed in the 3-D view (Fig. 6K).

The 3-D view visualizes the trajectories of all four sensors within a suggestion. Since each suggestion oV_q^s represents the optimized sensor data of a poorly-performed strokes, we can reconstruct the trajectories based on oV_q^s. We referred to the widely-used basic dead reckoning [45, 90] for reconstruction. The basic idea of dead reckoning is to accumulate the displacement happening in each time interval where the velocity and acceleration are assumed to be constant. Therefore, we used the acceleration $(acc_x(t), acc_y(t), acc_z(t))$ data of four sensors in oV_q^s with two assumptions for reconstruction. First, we found the point t_0 ($t_0 \in [0, 30)$) where the acceleration was the smallest (close to zero) in the data. We set this point as the beginning of a trajectory because we assumed that at this point, trainees were holding their ready actions and all sensors were static ($v_{t_0} = 0$). Second, we assumed that the acceleration a_i between two sampling points, t_i and t_{i+1}, was constant since there was only $0.05s$ between t_i and t_{i+1}. With these assumptions, we computed the velocity v_{t_i} of a sensor at t_i as follows,

$$v_{t_i} = v_{t_0} + \sum_{j=t_0}^{t_{i-1}} a_j * \Delta t, (\Delta t = 0.05s) \tag{7}$$

where Δt was the duration between two sampling points. With the velocity, we can compute the displacement D_i between t_i and t_{i+1} with the acceleration data as follows,

$$D_i = \frac{(v_{t_i} + v_{t_{i+1}}) * \Delta t}{2}, (\Delta t = 0.05s) \tag{8}$$

Finally, we accumulated all displacements of the sensor to reconstruct its trajectory as follows,

$$T = \sum_{i=t_0}^{t_n} D_i (t_n = 30) \tag{9}$$

In this way, we reconstructed all four trajectories of a suggestion in blue in the 3-D view. Besides, we also reconstructed the trajectory of the poorly-performed stroke in red for comparison. The origins of the trajectories are placed at the same point. In this way, coaches can rotate the plot to efficiently compare the trajectories.

6 EVALUATION

This section illustrates two use cases we conducted with a coach and two trainees, T1 (male, 24) and T2 (male, 21), from the university team. The coach has served the university team for more than five years. He has taught considerable players of the university team. Both trainees are right-handed players who use shakehand rackets with pimple-in rubber on both sides.

6.1 Case 1: Training of Topspin

T1 wanted to improve his offensive techniques during matches. He chose topspin for training since topspin is a frequently-used offensive technique that requires extreme speed and spin. The coach adjusted the setting of the ball machine to make it serve the ball to the forehand of T1 with long placement. Then the coach created a new training session, and T1 started to use topspin to hit the ball (Fig. 5). During this training, the coach examined the speed and spin of each stroke displayed in the training view. He commented that this was quite convenient since the training process were accurately quantified, recorded, and visualized. He could efficiently and comprehensively analyze T1's performance. According to the training flow, the coach found that the speed and spin of T1's stroke fluctuated. He observed T1's motions and guessed that T1 swung the racket too high, which led to the low speed and low spin of the ball. With the guess, the coach stopped the training and used the system for analysis. He moved the slider of speed to $13m/s$ and found that most of the strokes are not qualified in speed as Fig. 6E shows. To validate his guess, he chose a poorly performed stroke (Fig. 6G) and clicked the "optimization" button. In the suggestion list, the coach did not choose the one with the highest feasibility since the expected speed was not qualified. After weighing the pros and cons, he chose the third one and paid attention to the "racket" plot (Fig. 6L). In the plot, the blue trajectory was lower than the red, which means the system suggested that T1 should lower the racket when hitting the ball. This result validated the coach's guess. The coach said that this function provided a proof-of-principle for his guess, which could save much time for trial and error since they often needed to rely on their knowledge and experience when giving suggestions.

The coach told T1 about the suggestion and started a second training. In the beginning, T1 performed well, and the speed and spin met the coach's standard. However, after dozens of strokes, the stroke quality started to drop as Fig. 6D shows. The coach explained that at the beginning, T1 improved the strokes by lowering the racket. However, after a while, T1 may feel tired since topspin is a technique consuming a lot of physical strength, and forget the optimized motion. Therefore, the stroke quality dropped in the second half of the training. We asked T1 about the explanation. T1 said that he indeed felt tired in the second half, and his motion distorted.

6.2 Case 2: Training of Short

T2 wanted to enhance his control technique, short. Short is a frequently-used control technique after an opponent's serve. It requires high spin for effective control. However, compared with the spin of topspin, the spin of short is much lower, as shown in Table 3. The coach set the ball machine to make it serve the ball to the backhand of T1 with short placement. Moreover, the ball was served with a backspin to simulate the condition of the opponent's serve. With the help of the training flow, the coach found that the spins of the strokes were not qualified (Fig. 8)A. The spins of all strokes were lower than $40r/s$. He randomly chose a poorly-performed stroke (Fig. 8C) for optimization.

The coach checked the suggestion list and chose a suggestion that could improve the spin to $45r/s$. He further explored the four trajectories of this suggestion. During his exploration, he said that a significant point of this system was that it could not only validate and refine his experience and knowledge through data-driven methods, but also provide new perspectives of training optimization for him, which was much efficient and reliable than only relying on their imagination and speculation. The optimized trajectory of the right wrist made sense for him (Fig. 8D). He explained that T2 should move his right wrist diagonally down to extend the time of friction between the ball and the rubber as suggested by the blue trajectory (Fig. 8D). The original (red) trajectory indicated that T2's right wrist was moving almost horizontally, which would bounce the ball earlier, leaving little time for friction. The lack of friction was an important reason for the low spin of the stroke. Then, the coach told T2 the optimization suggestion, and T2 started a new training session. This time, T2 moved his right wrist diagonally down to extend the time of friction. As Fig. 8B shows, the spin of T2's short was improved, much higher than the first training.

6.3 Feedback

The coach thought highly of our training system. He thought Tac-Trainer created a new training mode for him. This mode solves two limitations of the conventional training mode. First, conventionally, he needed to assess trainees' strokes based on his eyes and short-term

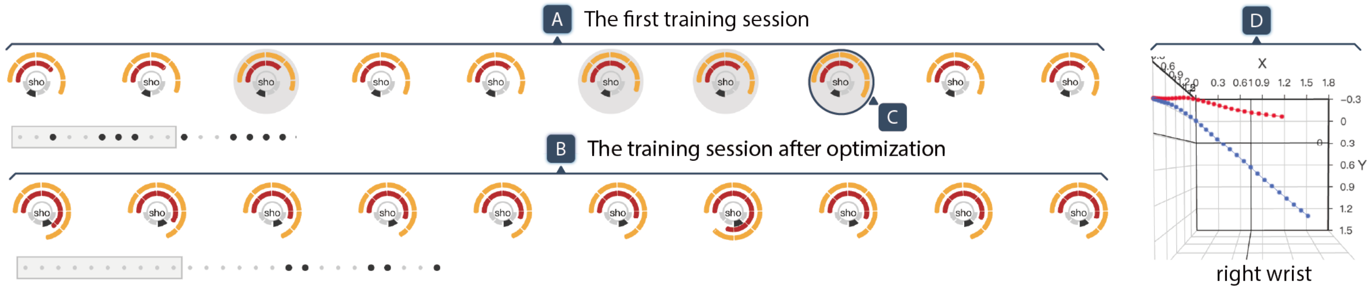

Fig. 8. The details of case 2. A is the training flow of T2's first training session. B is the training flow after T2 received the optimization suggestion. C is the stroke the coach selected for optimization. D is the optimization suggestion of the stroke for the right wrist.

memory, which was not reliable and scalable enough. He could not access and remember the exact speed and spin of each stroke and the detailed motions of trainees, which led to the missing of some poor performance patterns of trainees. With the system, all technical attributes and indicators could be quantified, recorded, and visualized for interactive exploration. The performance patterns of trainees could be comprehensively and accurately analyzed.

Second, when giving suggestions to trainees, he could only rely on his experience and knowledge, which did not always work during the training. Once new training problems occur, he needed to adjust the suggestion through trial and error. In our system, the generated suggestions could provide a data-driven proof-of-principle, validating and refining his suggestions and inspiring new suggestions. Although considerable other data-driven methods have been introduced to solve this limitation, these methods were not friendly to him because the learning curve of the underlying mathematical models was high. He said that, unlike these methods, Tac-Trainer provided an interactive visualization interface that got him into the loop of data-driven training.

7 DISCUSSION

In this section, we first discuss the research values and directions of IoT + VA from two aspects, VA for IoT and IoT for VA. Then, we discuss the limitations and future work.

7.1 IoT + VA as a New Research Direction

VA for IoT: Visual analytics can help spread the usage of IoT technology, especially for users who have no knowledge of processing the sensor data. Current smart products, such as smartphones and smartwatches, have provided considerable benefits for users. However, they prevent the users from accessing the IoT data, which blocks various creative and interesting ideas raised by the users. For example, as stated by our experts, although they knew that IoT devices could collect detailed kinematic features, they did not apply it to practical training since they had no ideas about how to use the data. In such a condition, a visual analytics system designed for IoT devices can provide opportunities for users to understand and coordinate the IoT data to conduct various analysis tasks, facilitate the popularization of IoT technology. Challenges will lie in designing efficient visual analytics methods for various types of IoT data. Besides, visual analytics can also facilitate the development of IoT technology, which is useful for developers. For example, when designing an IoT system, developers need to consider many factors (e.g., sensor precision, the data transmission frequency, etc.), which is a non-trivial task. In such a condition, a visual analytics system connecting to the IoT system can alleviate the difficulty by supporting visual debugging. Challenges lie in how to coordinate the interaction between an IoT system and a visual analytics system.

IoT for VA: With the development of IoT technology, existing IoT devices have excellent computing power. This feature can help improve the design of visual analytics systems in two aspects. First, the challenge of data quality and uncertainty [44] can be alleviated. These challenges are often solved by techniques of data provenance [18] and data wrangling [42]. IoT provides a new perspective of improving data quality from the source. For example, anomaly detection algorithms can be embedded into IoT devices. Once a sensor is abnormal, the device will send a warning to the visual analytics system and activate the backup sensor. Challenges will lie in designing algorithms for IoT devices to efficiently detect and handle data anomalies. Second, the scalability of visual analytics systems can be further improved. The powerful computing ability of existing IoT devices enables researchers to develop distributed visual analytics systems, alleviating the burden of large data volume and complexity. For example, some complex data processing components, such as simulation models, pattern recognition models, etc., can be divided and shared with IoT devices to improve the computing efficiency of systems. Challenges will lie in designing structures of distributed visual analytics system.

7.2 Limitations & Future Work

One limitation of our work lies in the generalizability of the data interpretation models. We tried to train general models. However, the accuracies were not ideal due to the great differences in kinematic features between various players. This hinders the efficient extension of more trainees. Therefore, in the future, we plan to improve the generalizability of the models. A possible direction is expanding the pool of trainees in our system. First, we need to identify the different playing styles of trainees based on their kinematic features. Then, for each style, we customized a series of robust data interpretation models. Finally, when a new trainee appears, we only need to classify the trainee based on the styles and choose the model of this style for him/her.

In addition, we are going to extend Tac-Trainer to other sports, including racket sports such as tennis and badminton and team sports such as soccer and basketball. Tac-Trainer can be easily implemented for other racket sports. For team sports, we need to collaborate with related experts to identify corresponding requirements, such as the most important factors, the assessment standard, etc. For example, in soccer, the devices may be fixed on trainees' feet. Such changes will inevitably change the design of some implementation details in Tac-Trainer.

8 CONCLUSION

This work aims to address an important problem of applying IoT technology to racket sports training. Recognizing visual analytics as the key component, we fully explore different pathways of coordinating IoT devices and visual analytics and propose an IoT + VA framework, Tac-Trainer, based on our practice. Differed from regular visual analytics frameworks, Tac-Trainer traces back to the data source and provides guidance for the IoT device configuration, sensor data processing, data inference, and IoT data visualization. To evaluate the framework, we implement a proof-of-concept system for table tennis training and conduct two case studies on improving trainees' stroke techniques. From this work, we identify the mutually reinforcing relation between IoT and visual analytics, i.e., visual analytics can popularize the application of IoT while IoT can alleviate issues such as data quality and scalability for visual analytics. We hope that this work can facilitate training in the sports domain and inspire future studies of **IoT4VA** and **VA4IoT**.

ACKNOWLEDGMENTS

The work was supported by NSFC (62072400) and the Collaborative Innovation Center of Artificial Intelligence by MOE and Zhejiang Provincial Government (ZJU).

REFERENCES

[1] ChoreiKing 2. http://www.gogotak.co.kr/bbs/content.php?co_id=CHOREIKING2. Accessed at June 22, 2022.

[2] Courtmatics. http://www.courtmatics.com/. Accessed at June 22, 2022.

[3] Janova Smart Racket. https://www.kickstarter.com/projects/janova/janova-smart-racket. Accessed at June 22, 2022.

[4] Qlipp Plus. https://www.qlipp.com/de/plus/. Accessed at June 22, 2022.

[5] Smart Xiaoyu 2.0. http://www.coollang-global.com/welcome/badminton. Accessed at June 22, 2022.

[6] Zepp. http://www.zepplabs.com/en-us/tennis/. Accessed at June 22, 2022.

[7] Is volleyball the world's most popular sport? https://www.topendsports.com/world/lists/popular-sport/final.htm, 2022. Accessed: 2022-02-09.

[8] Perspectives of visualization onboarding and guidance in VA. *Visual Informatics*, 6(1):68–83, 2022.

[9] A. Ahmadi, E. Mitchell, C. Richter, F. Destelle, M. Gowing, N. E. O'Connor, and K. Moran. Toward automatic activity classification and movement assessment during a sports training session. *IEEE Internet of Things Journal*, 2(1):23–32, 2014.

[10] A. Ahmadi, D. Rowlands, and D. A. James. Towards a wearable device for skill assessment and skill acquisition of a tennis player during the first serve. *Sports Technology*, 2(3-4):129–136, 2009.

[11] G. Andrienko, N. Andrienko, G. Anzer, P. Bauer, G. Budziak, G. Fuchs, D. Hecker, H. Weber, and S. Wrobel. Constructing Spaces and Times for Tactical Analysis in Football. *IEEE Transactions on Visualization and Computer Graphics*, (4):2280–2297, 2019.

[12] J. Bernard, A. Vögele, R. Klein, and D. W. Fellner. Approaches and Challenges in the Visual-interactive Comparison of Human Motion Data. In *Proceedings of International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pp. 217–224, 2017.

[13] J. Bernard, N. Wilhelm, B. Krüger, T. May, T. Schreck, and J. Kohlhammer. Motionexplorer: Exploratory Search in human motion capture data based on hierarchical aggregation. *IEEE transactions on visualization and computer graphics*, 19(12):2257–2266, 2013.

[14] P. Blank, B. H. Groh, and B. M. Eskofier. Ball speed and spin estimation in table tennis using a racket-mounted inertial sensor. In *Proceedings of ACM International Symposium on Wearable Computers*, pp. 2–9, 2017.

[15] P. Blank, S. Hofmann, M. Kulessa, and B. M. Eskofier. MiPod 2: A New Hardware Platform for Embedded Real-Time Processing in Sports and Fitness Applications. In *Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pp. 881–884, 2016.

[16] P. Blank, J. Hoßbach, D. Schuldhaus, and B. M. Eskofier. Sensor-based stroke detection and stroke type classification in table tennis. In *Proceedings of ACM International Symposium on Wearable Computers*, pp. 93–100, 2015.

[17] P. Blank, P. Kugler, and B. M. Eskofier. mipod-a wearable sports and fitness sensor. In *Proceedings of Symposium der Sektion Sportinformatik der Deutschen Vereinigung für Sportwissenschaft*, pp. 78–79, 2015.

[18] P. Buneman, S. Khanna, and T. Wang-Chiew. Why and where: A characterization of data provenance. In *International conference on database theory*, pp. 316–330. Springer, 2001.

[19] S. Butterworth. On the theory of filter amplifiers. *Wireless Engineer*, 7(6):536–541, 1930.

[20] A. Cao, X. Xie, J. Lan, H. Lu, X. Hou, J. Wang, H. Zhang, D. Liu, and Y. Wu. MIG-Viewer: Visual analytics of soccer player migration. *Visual Informatics*, 5(3):102–113, 2021.

[21] D. Cervone, A. D'Amour, L. Bornn, and K. Goldsberry. POINTWISE: Predicting Points and Valuing Decisions in Real Time with NBA Optical Tracking Data. In *Proceedings of MIT Sloan Sports Analytics Conference*, pp. 3–3, 2014.

[22] G. Y.-Y. Chan, L. G. Nonato, A. Chu, P. Raghavan, V. Aluru, and C. T. Silva. Motion Browser: Visualizing and Understanding Complex Upper Limb Movement Unde Obstetrical Brachial Plexus Injuries. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):981–990, 2020.

[23] L. Chen, S. Peng, and X. Zhou. Towards efficient and photorealistic 3D human reconstruction: a brief survey. *Visual Informatics*, 5(4):11–19, 2021.

[24] T. Chen and C. Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

[25] W. Chen, T. Lao, J. Xia, X. Huang, B. Zhu, W. Hu, and H. Guan. GameFlow: Narrative Visualization of NBA Basketball Games. *IEEE Transactions on Multimedia*, 18(11):2247–2256, 2016.

[26] Z. Chen, S. Ye, X. Chu, H. Xia, H. Zhang, H. Qu, and Y. Wu. Augmenting Sports Videos with VisCommentator. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):824–834, 2021.

[27] F. Cheng, Y. Ming, and H. Qu. DECE: Decision Explorer with Counterfactual Explanations for Machine Learning Models. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1438–1447, 2020.

[28] X. Chu, X. Xie, S. Ye, H. Lu, H. Xiao, Z. Yuan, Z. Chen, H. Zhang, and Y. Wu. TIVEE: Visual Exploration and Explanation of Badminton Tactics in Immersive Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):118–128, 2021.

[29] M. Du and X. Yuan. A survey of competitive sports data visualization and visual analysis. *Journal of Visualization*, (1):47–67, 2020.

[30] A. Franks, A. Miller, L. Bornn, and K. Goldsberry. Counterpoints: Advanced Defensive Metrics for NBA Basketball. In *Proceedings of MIT Sloan Sports Analytics Conference*, 2015.

[31] K. Goldsberry. Courtvision: New Visual and Spatial Analytics for the NBA. In *Proceedings of MIT Sloan Sports Analytics Conference*, pp. 12–15, 2012.

[32] Y. Gu, C. Yu, S. Shao, and J. S. Baker. Effects of table tennis multi-ball training on dynamic posture control. *PeerJ*, 6:e6262, 2019.

[33] H. Guo, S. Zou, Y. Xu, H. Yang, J. Wang, H. Zhang, and W. Chen. DanceVis: toward better understanding of online cheer and dance training. *Journal of Visualization*, 25(1):159–174, 2022.

[34] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[35] Y. Iino and T. Kojima. Kinetics of the upper limb during table tennis topspin forehands in advanced and intermediate players. *Sports Biomechanics*, 10(4):361–377, 2011.

[36] K. Ishida. IoT application in sports to support skill acquisition and improvement. In *Proceedings of IEEE Conference on Service-Oriented Computing and Applications*, pp. 184–189, 2019.

[37] Y. Ishikawa and I. Fujishiro. Tidegrapher: Visual Analytics of Tactical Situations for Rugby Matches. *Visual Informatics*, 2(1):60–70, 2018.

[38] S. Jang, N. Elmqvist, and K. Ramani. Gestureanalyzer: visual analytics for pattern analysis of mid-air hand gestures. In *Proceedings of ACM symposium on Spatial user interaction*, pp. 30–39, 2014.

[39] S. Jang, N. Elmqvist, and K. Ramani. Motionflow: Visual abstraction and aggregation of sequential patterns in human motion tracking data. *IEEE transactions on visualization and computer graphics*, 22(1):21–30, 2015.

[40] J. M. Jarning, K.-M. Mok, B. H. Hansen, and R. Bahr. Application of a tri-axial accelerometer to estimate jump frequency in volleyball. *Sports biomechanics*, 14(1):95–105, 2015.

[41] A. Jiang, M. A. Nacenta, and J. Ye. VisuaLizations As Intermediate Representations (VLAIR): An approach for applying deep learning-based computer vision to non-image-based data. *Visual Informatics*, 2022.

[42] S. Kandel, J. Heer, C. Plaisant, J. Kennedy, F. Van Ham, N. H. Riche, C. Weaver, B. Lee, D. Brodbeck, and P. Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, 2011.

[43] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems 30*, pp. 3146–3154. 2017.

[44] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler. *Visual Analytics: Scope and Challenges*, pp. 76–90. 2008.

[45] I. Klein and R. Diamant. Dead reckoning for trajectory estimation of underwater drifters under water currents. *Journal of Marine Science and Engineering*, 8(3):205–205, 2020.

[46] K. Klein, S. Jaeger, J. Melzheimer, B. Wachter, H. Hofer, A. Baltabayev, and F. Schreiber. Visual analytics of sensor movement data for cheetah behaviour analysis. *Journal of Visualization*, 24(4):807–825, 2021.

[47] P. R. Krekel, E. R. Valstar, J. De Groot, F. H. Post, R. G. Nelissen, and C. P. Botha. Visual analysis of multi-joint kinematic data. In *Proceedings of Computer Graphics Forum*, vol. 29, pp. 1123–1132, 2010.

[48] S. Kusubori, K. Yoshida, and H. Sekiya. The functions of spin on shot trajectory in table tennis. In *Proceedings of ISBS*, 2012.

[49] J. Lan, J. Wang, X. Shu, Z. Zhou, H. Zhang, and Y. Wu. RallyComparator: visual comparison of the multivariate and spatial stroke sequence in table tennis rally. *Journal of Visualization*, 25(1):143–158, 2022.

[50] J. Lan, Z. Zhou, J. Wang, H. Zhang, X. Xie, and Y. Wu. SimuExplorer: Visual Exploration of Game Simulation in Table Tennis. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021.

[51] A. Lees. Science and the major racket sports: a review. *Journal of sports sciences*, 21(9):707–732, 2003.

[52] I. Li, A. K. Dey, and J. Forlizzi. Using Context to Reveal Factors that Affect Physical Activity. *ACM Transactions on Computer-Human Interaction*, 19(1):1–21, 2012.

[53] R. Liu, Z. Wang, X. Shi, H. Zhao, S. Qiu, J. Li, and N. Yang. Table tennis stroke recognition based on body sensor network. In *Proceedings of International Conference on Internet and Distributed Computing Systems*, pp. 1–10, 2019.

[54] A. G. Losada, R. Therón, and A. Benito. BKViz: A Basketball Visual Analysis Tool. *IEEE Computer Graphics and Applications*, 36(6):58–68, 2016.

[55] S. Lu, X. Zhang, J. Wang, Y. Wang, M. Fan, and Y. Zhou. An IoT-Based Motion Tracking System for Next-Generation Foot-Related Sports Training and Talent Selection. *Journal of Healthcare Engineering*, 2021, 2021.

[56] R. Ma, D. Yan, H. Peng, T. Yang, X. Sha, Y. Zhao, and L. Liu. Basketball movements recognition using a wrist wearable inertial measurement unit. In *Proceedings of IEEE International Conference on Micro/Nano Sensors for AI, Healthcare, and Robotics*, pp. 73–76, 2018.

[57] H. Mansoor, W. Gerych, A. Alajaji, L. Buquicchio, K. Chandrasekaran, E. Agu, and E. Rundensteiner. ARGUS: Interactive visual analysis of disruptions in smartphone-detected Bio-Behavioral Rhythms. *Visual Informatics*, 5(3):39–53, 2021.

[58] T. Mitsui, S. Tang, and S. Obana. Support system for improving golf swing by using wearable sensors. In *Proceedings of Eighth International Conference on Mobile Computing and Ubiquitous Networking*, pp. 100–101, 2015.

[59] H. Oagaz, B. Schoun, and M.-H. Choi. Performance Improvement and Skill Transfer in Table Tennis Through Training in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021.

[60] J. P. Ono, C. A. Dietrich, and C. T. Silva. Baseball Timeline: Summarizing Baseball Plays Into a Static Visualization. *Computer Graphics Forum*, 37(3):491–501, 2018.

[61] M. L. Parry, P. A. Legg, D. H. S. Chung, I. W. Griffiths, and M. Chen. Hierarchical Event Selection for Video Storyboards with a Case Study on Snooker Video Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1747–1756, 2011.

[62] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[63] C. Perin, J. Boy, and F. Vernier. Using Gap Charts to Visualize the Temporal Evolution of Ranks and Scores. *IEEE Computer Graphics and Applications*, 36(5):38–49, 2016.

[64] C. Perin, R. Vuillemot, and J.-D. Fekete. SoccerStories: A Kick-off for Visual Soccer Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2506–2515, 2013.

[65] C. Perin, R. Vuillemot, and J.-D. Fekete. A table!: improving temporal navigation in soccer ranking tables. In *Proceedings of CHI Conference on Human Factors in Computing Systems*, pp. 887–896, 2014.

[66] C. Perin, R. Vuillemot, C. D. Stolper, J. T. Stasko, J. Wood, and S. Carpendale. State of the Art of Sports Data Visualization. *Computer Graphics Forum*, 37(3):663–686, 2018.

[67] H. Pileggi, C. D. Stolper, J. M. Boyle, and J. T. Stasko. SnapShot: Visualization to Propel Ice Hockey Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2819–2828, 2012.

[68] T. Polk, D. Jäckle, J. Häußler, and J. Yang. CourtTime: Generating Actionable Insights into Tennis Matches Using Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):397–406, 2020.

[69] T. Polk, J. Yang, Y. Hu, and Y. Zhao. TenniVis: Visualization for Tennis Match Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2339–2348, 2014.

[70] M. Ryoo, N. Kim, and K. Park. Visual analysis of soccer players and a team. *Multimedia Tools and Applications*, 77(12):15603–15623, 2018.

[71] M. Stein, H. Janetzko, T. Breitkreutz, D. Seebacher, T. Schreck, M. Grossniklaus, I. D. Couzin, and D. A. Keim. Director's Cut: Analysis and Annotation of Soccer Matches. *IEEE Computer Graphics and Applications*, 36(5):50–60, 2016.

[72] M. Stein, H. Janetzko, A. Lamprecht, T. Breitkreutz, P. Zimmermann, B. Goldlücke, T. Schreck, G. L. Andrienko, M. Grossniklaus, and D. A. Keim. Bring It to the Pitch: Combining Video and Movement Data to Enhance Team Sport Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):13–22, 2018.

[73] R. Therón and L. Casares. Visual Analysis of Time-Motion in Basketball Games. In *Proceedings of Smart Graphics*, pp. 196–207, 2010.

[74] S. Wachter, B. Mittelstadt, and C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31:841–841, 2017.

[75] J. Wang, J. Wu, A. Cao, Z. Zhou, H. Zhang, and Y. Wu. Tac-miner: Visual Tactic Mining for Multiple Table Tennis Matches. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):2770–2782, 2021.

[76] J. Wang, K. Zhao, D. Deng, A. Cao, X. Xie, Z. Zhou, H. Zhang, and Y. Wu. Tac-simur: Tactic-based Simulative Visual Analytics of Table Tennis. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):407–417, 2020.

[77] Y. Wang, M. Chen, X. Wang, R. H. M. Chan, and W. J. Li. IoT for Next-Generation Racket Sports Training. *IEEE Internet of Things Journal*, 5(6):4558–4566, 2018.

[78] D. Wei, C. Li, H. Shao, Z. Tan, Z. Lin, X. Dong, and X. Yuan. Sensoraware: visual analysis of both static and mobile sensor information. *Journal of Visualization*, 24(3):597–613, 2021.

[79] N. Wilhelm, A. Vögele, R. Zsoldos, T. Licka, B. Krüger, and J. Bernard. FuryExplorer: visual-interactive exploration of horse motion capture data. In *Proceedings of Visualization and Data Analysis 2015*, vol. 9397, pp. 148–162, 2015.

[80] E. Wu, M. Piekenbrock, T. Nakumura, and H. Koike. SPinPong - Virtual Reality Table Tennis Skill Acquisition using Visual, Haptic and Temporal Cues. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2566–2576, 2021.

[81] Y. Wu, J. Lan, X. Shu, C. Ji, K. Zhao, J. Wang, and H. Zhang. iTTVis: Interactive Visualization of Table Tennis Data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):709–718, 2018.

[82] Y. Wu, X. Xie, J. Wang, D. Deng, H. Liang, H. Zhang, S. Cheng, and W. Chen. ForVizor: Visualizing Spatio-Temporal Team Formations in Soccer. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):65–75, 2019.

[83] K. Xia, H. Wang, M. Xu, Z. Li, S. He, and Y. Tang. Racquet sports recognition using a hybrid clustering model learned from integrated wearable sensor. *Sensors*, 20(6):1638–1638, 2020.

[84] W. Xie, K. C. Teh, and Z. F. Qin. Speed and spin of 40mm table tennis ball and the effects on elite players. In *Proceedings of ISBS-Conference Proceedings Archive*, pp. 623–626, 2002.

[85] X. Xie, J. Wang, H. Liang, D. Deng, S. Cheng, H. Zhang, W. Chen, and Y. Wu. Passvizor: Toward Better Understanding of the Dynamics of Soccer Passes. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1322–1331, 2021.

[86] R. Yang, E. Shin, M. W. Newman, and M. S. Ackerman. When Fitness Trackers Don't 'Fit': End-User Difficulties in the Assessment of Personal Tracking Device Accuracy. In *Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 623–634, 2015.

[87] L. Yao, A. Bezerianos, R. Vuillemot, and P. Isenberg. Visualization in Motion: A Research Agenda and Two Evaluations. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–16, 2022.

[88] S. Ye, Z. Chen, X. Chu, Y. Wang, S. Fu, L. Shen, K. Zhou, and Y. Wu. Shuttlespace: Exploring and Analyzing Movement Trajectory in Immersive Visualization. *IEEE Transactions on Visualization and Computer Graphics*, pp. 860–869, 2020.

[89] Q. Zhi, S. Lin, P. T. Sukumar, and R. A. Metoyer. Gameviews: Understanding and Supporting Data-driven Sports Storytelling. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 269:1–269:13, 2019.

[90] Q.-L. Zhou, Y. Zhang, Y.-H. Qu, and C.-A. Rabbath. Dead reckoning and Kalman filter design for trajectory tracking of a quadrotor UAV. In *Proceedings of IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*, pp. 119–124, 2010.

[91] Z.-H. Zhou and J. Feng. Deep Forest: Towards an alternative to deep neural networks. In *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 3553–3559, 2017.