Visual Analytics of Multivariate Event Sequence Data in Racquet Sports

Jiang Wu* Ziyang Guo* Zuobin Wang* Qingyang Xu* Yingcai Wu*

The State Key Lab of CAD&CG
Zhejiang University

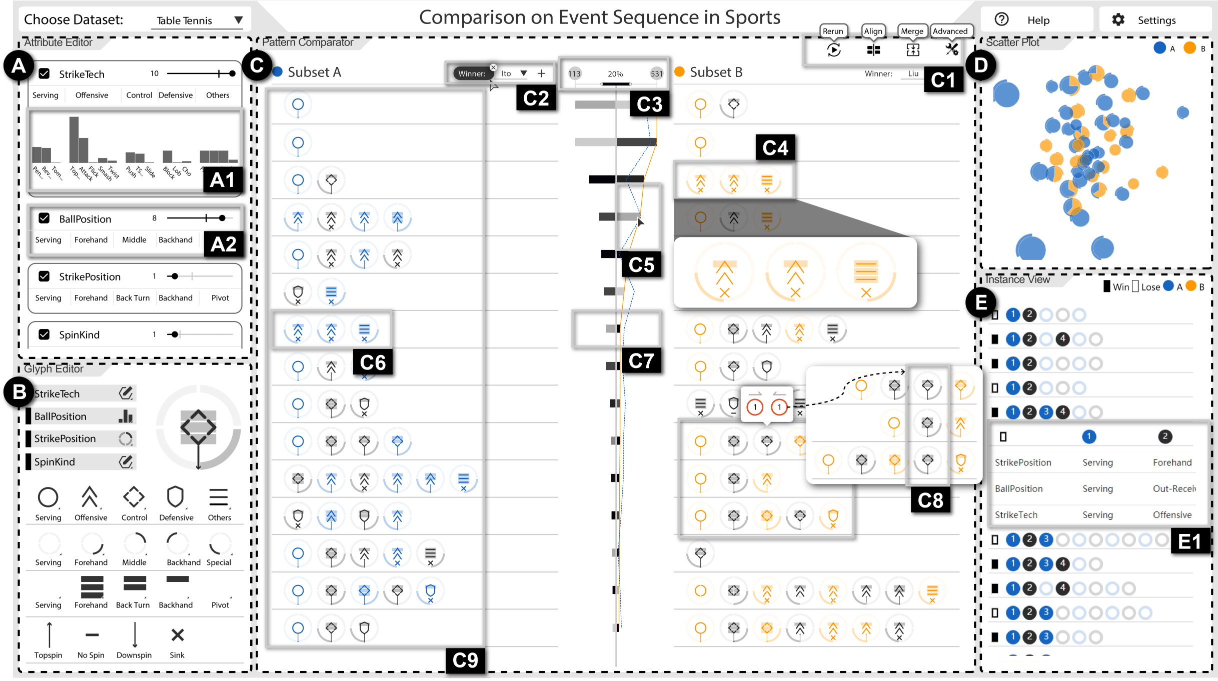

Figure 1: System Interface. (A) *Attribute Editor* is used to adjust the focus of the analysis and import domain knowledge. (B) *Glyph Editor* can fine-tune the glyph design, where a glyph is used to encode an event to show multiple attributes simultaneously. (C) *Pattern Comparator* provides a one-to-one comparison on patterns. (D) *Scatterplot* provides a coarse-level overview and filters patterns. (E) *Instance View* shows the detailed information of sequences.

Abstract

In this work, we propose a generic visual analytics framework to support tactic analysis based on data collected from racquet sports (such as tennis and badminton). The proposed approach models each rally in a game as a sequence of hits (i.e., events) until one athlete scores a point. Each hit can be described with a set of attributes, such as the positions of the ball and the techniques used to hit the ball (such as *drive* and *volley* in tennis). Thus, the mentioned sequence of hits can be viewed as a multivariate event sequence. By detecting and analyzing the multivariate subsequences that frequently occur in the rallies (namely, tactical patterns), athletes can gain insights into the playing styles adopted by their opponents, and therefore help them identify systematic weaknesses of the opponents and develop counter strategies in matches. To support such analysis effectively, we propose a steerable multivariate sequential pattern mining algorithm with adjustable weights over event attributes, such that the domain expert can obtain frequent tactical patterns according to the attributes specified by himself. We also propose a re-configurable glyph design to help users simultaneously analyze multiple attributes of the hits. The framework further supports comparative analysis of the tactical patterns, e.g., for different athletes or the same athlete playing under different conditions. By applying the framework on two datasets collected in tennis and badminton matches, we demonstrate that the system is generic and effective for tactic analysis in sports and can help identify signature techniques used by individual athletes. Finally, we discuss the strengths and limitations of the proposed approach based on the feedback from the domain experts.

Index Terms: Sports Analytics; Event Sequence; Multivariate Data; Sequential Pattern Mining; Comparative Analysis

1 Introduction

Sports analytics, especially sports data visualization, are increasingly important nowadays as athletes and coaches try to derive insights from the data collected in matches to gain competitive advantage. Among all sports, racquet sports (*e.g.*, tennis and badminton) have a similar rule that players need to hit the ball alternatively until one player scores a point. Each stroke (*i.e.*, one hit) can be considered as an event, which depends on the previous hits and prepares for the following hits. And each rally (*i.e.*, a series of alternating hits from one player's serving to one player's winning) can be considered as a sequence. For each stroke, its detailed characteristics (*e.g.*, which technique is used, in which area of the court the ball is hit, etc.) can be recorded. Based on the detailed characteristics of each stroke, tactical patterns (*i.e.*, the frequently used subsequences) can be extracted from hundreds of related rallies (*e.g.*, rallies played in a

*e-mail: {wujiang5521, ziyangguo27, zuobinwang01, qingyangxu17, ycwu}@zju.edu.cn, Yingcai Wu is the corresponding author.

single game or between the same opponents) to reveal the individual styles of the athletes, which can help them identify existing tactical patterns that lead to better chances of winning or develop effective counter strategies. Example tactical patterns in tennis include forcing the opponent to run side-to-side and consistent groundies. Among the numerous tactical patterns in tennis, domain experts want to know which ones a player prefers and which ones can help them score more points.

Given that such insights can have a profound impact on the outcome of the games, our goal is to develop a visual analytics framework that can help domain experts perform sequential tactical analysis on event sequence data in racquet sports. However, several features of racquet sports make developing such a framework a challenging task. First, to fully describe the tactical characteristics of a stroke, multiple attributes are needed. Taking tennis as an example, to fully describe a stroke, we need to know the placement of the ball, the position of the two players, the technique used (*e.g.*, smash and drive), and so on. A pattern mining algorithm that is capable of comprehensively considering multiple attributes and a visual encoding scheme that can simultaneously display multiple attributes are therefore essential. Furthermore, the algorithm should be steerable to emphasize the more important attributes in exploratory analysis. For example, when analyzing *Nadal* in tennis, who excels at nimble movements, the attribute encoding the players' positions should be assigned a higher weight.

Second, to enable effective tactical analysis in racquet sports, the system should support comparative analysis of the sequential patterns. Players usually change their strategies depending on various factors, including the opponent he/she is playing against, whether he/she serves in the rally, whether the rally is a tiebreaker, and so on. Comparative analysis of the patterns could reveal insights about how the athletes adapt their strategies to different scenarios, such that domain experts can know the player better and consider how to improve his/her playing strategies (or how to win against him/her). However, the rallies can differ in many ways with variations in multiple attributes. Thus, performing a cross-comparison is a challenging task.

To address these challenges, we introduce a visual analytics framework that enables multivariate pattern mining. To tackle the first challenge, domain experts can interactively define which attributes are more important to be analyzed. We also introduce a multivariate pattern mining algorithm to extract players' tactical patterns considering multiple attributes with weight. At the same time, a steerable glyph design that combines multiple symbols is implemented to display multiple attributes simultaneously. Based on the steerable definition of analysis requirements and the intuitive glyphs, domain experts can obtain a clear overview of tactical patterns used by players.

To address the second challenge, we design the visualization views to show multiple levels of detail [45]. Users can construct two subsets containing rallies with different conditions (*e.g.*, who serves and whether it is a tiebreaker). At the highest level, to help users understand how the patterns within these two subsets differ, we project the patterns into a 2D plane according to the editing distance between the patterns and visually encode the winning rate of the patterns. The individual patterns and the killer patterns (*i.e.*, the unusual but effective ones) will be clear at a glance. At the medium level, a pattern list shows the set of sequential patterns. We also provide a rich set of interactions for comparison, such as linked-highlighting, alignment, etc. At the lowest level, we show the details of each rally.

To evaluate our system, we conducted two example usage scenarios with domain experts on two different racquet sports data. One is tennis, which showcases the difference of an athlete's behavior between the rallies after deuces (key time) and other rallies. The second case is badminton. We compare the tactical characteristics of two athletes. We also held expert interviews and gathered feedback.

In summary, the main contributions of this work are as follows:

- A visual analytics framework that comprehensively considers and displays multiple attributes of events in racquet sports data.
- A steerable multivariate pattern mining algorithm based on the minimum description length (MDL) principle.
- A system for comparing sports event sequence data, which includes multi-scale visualization and intuitive interactions.
- Two example usage scenarios with sports event sequence datasets and expert interviews that reveal the usability of the analysis method and the effectiveness of the system.

2 Related Work

This work is mainly related to event sequence mining methods and visualizations of the mining results, visual comparison between event sequences, and visual analytics of sports data.

2.1 Event Sequence Mining and Visualization

The pattern mining algorithm is a common and useful method for event sequence analysis. Methods based on sequential pattern mining (SPM) attempt to find the subsequences with a frequency higher than a certain threshold [19,20]. Many visualizations build on top of SPM techniques and display the sequential patterns in the data, such as sunburst diagrams [46], flow diagrams [39], scatterplot [50], and tree-based visualizations [47]. However, in the sports domain, athletes prefer to use some unusual but effective tactical patterns, which will be ignored if the threshold is set incorrectly.

Recently, many works take machine learning as the core algorithm and help users identify the patterns with visualizations [54], such as a cluster of small units [6], scatterplot [23], and flow charts [24,25]. But the domain experts in sports prefer the results with great interpretability, which are usually not well supported by these algorithms.

The MDL principle is introduced to extract patterns from sequences [2,7,27] for interpretable pattern mining. For visualizations, Chen et al. [12] used the MDL principle to construct an overview for event sequences. AirVis [15] determined air pollution propagation patterns by MDL principle. MDL considers all sequences so that the unusual patterns will not be ignored. However, most existed MDL algorithm does not handle event sequence data with multiple attributes well. Bertens et al. [2] introduced an MDL-based method to handle the multivariate event sequences. However, the method can generate a cross-sequence pattern, *e.g.*, a pattern composed of the first event of two sequences. The cross-sequence patterns are useless in racquet sports analysis because different rallies are independent. Thus, we introduce our MDL-based algorithm to enable consideration of multiple attributes and effectivity in racquet sports.

Although few good pattern mining methods consider multiple variables comprehensively, previous studies had many visualizations for it. PatternFinder [18] and EventPad [8,9] allowed users to query for multivariate event sequences with a certain pattern. Timespan [32] showed how the multidimensional data changes over time. Liu et al. [31] used a tooltip to show multivariate events. The present study uses glyphs to display multiple attributes simultaneously.

2.2 Visual Comparison of Event Sequences

Following the guidance of the general visual comparison [21], a few visualizations have been designed to compare two event sequence datasets. MatrixWave [55] compared adjacent steps in clickstream data. Coco [33–35] provided a cohort comparison on event sequences that were divided into two cohorts. Borland et al. [5] designed tree-based and icicle-plot-based visualizations to compare the baseline and the user-specified focus cohort. In addition, Session viewer [29] and EventAction [17] included features for comparative analysis. However, for multivariate event sequence data, effectively comparing two datasets is difficult. This study presents a comparative visualization to explore the multivariate sports event sequence data with multiple levels of detail.

2.3 Visual Analytics of Sports Data

Visual analysis methods are developing rapidly and are widely used in various sports [41]. In racquet sports, such as tennis, Tennivis [43] visualized statistical data, such as score and rally information. CourtTime [42] facilitated pattern discovery with a novel visual metaphor. In badminton, Ren et al. [44] explored the effectiveness of different offensive techniques with a heatmap. ShuttleSpace [53] explored shuttle trajectories with a VR system. In table tennis, iTTVis [49] used a matrix to reveal the relationship among multiple attributes within strokes. Tac-Simur [48] simulated and visualized the results of some tactical changes. Aside from racquet sports, many visualization studies for other sports exist, such as soccer [40, 51, 52], basketball [10, 11, 22], baseball [16, 28, 38], rugby [13, 26], and ESports [1]. These studies utilized innovative visual designs and interactions to support domain-specific analysis tasks, such as showing the trajectory of the ball [42] and simulating the result of changing playing styles [48]. We use the event sequence to model the sports data and an MDL-based mining algorithm to obtain players' tactical patterns, which brings a new perspective to visual analytics in sports data. Inspired by the effectiveness of glyphs to encoding multivariate data [13, 43, 49], we design a glyph-based visualization. To enable our system in different racquet sports, we propose a steerable glyph design to configure tailored glyphs.

3 Background

Over the past three years, we have been closely working with two teams of experts in sports science from two universities, respectively. Each team comprises a professor and several PhD students dedicated to sports data analysis. All the team members used to be athletes. Both teams have been working for one of the top national teams in racquet sports (i.e., tennis and badminton, respectively) for more than five years. We have designed and developed various tools and systems to help the domain experts visually analyze the data. During the collaboration, we realized that the racquet sports data could be modeled as multivariate event sequences. Many analysis problems of the racquet sports data can be transformed to those of multivariate event sequences, especially the tactical pattern analysis.

The following section presents the data model, followed by the analysis tasks acquired from different racquet sports. Finally, we present the overview of our system.

3.1 Data Model

Our tennis/badminton datasets are provided by the two teams of domain experts who work for the national teams. The data of each match is manually recorded by undergraduates in sports science. As far as we know, the data quality is high, and we have not encountered any data quality/uncertainty issue. The two original datasets are in different formats. Thus, we designed a unified data format to normalize the two datasets. In the unified data format, each match includes a set of sequences. Each sequence S represents a rally and comprises an ordered list of events $S = (e_1, e_2, e_3, ..., e_n)$ and information about who serves, who wins, whether it is in a tiebreaker, and so on. Each event represents a hit and is described by multiple attributes, denoted as $e_i = \{a_1 = v_1^i, a_2 = v_2^i, ..., a_n = v_n^i\}$. The details of all the event attributes and their possible values for two different racquet sports, namely, tennis and badminton, are summarized in the appendix.

From the above definition, we introduce an algorithm (*Sec. 4*) that can identify sequential tactical patterns from the multivariate event sequence data. Specifically, a tactical pattern P is a subsequence that frequently presents in several sequences in the dataset.

3.2 Task Analysis

We collected the requirements for visually analyzing the multivariate event sequences in racquet sports from our domain experts. Based on the requirements, we summarized the analysis tasks as follows.

T1 **Adjust the weights of different attributes.** Domain experts usually focus on only one attribute or a weighted combination of multiple attributes for analysis. For example, a domain expert may only want to know about the position of a player, or may be more interested in knowing the relative position between the player and the ball. Analysts should be allowed to adjust the weight of different attributes to focus on the attributes of interest.

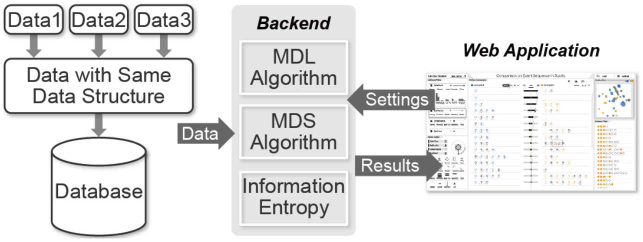

Figure 2: Architecture of the framework. It consists of three parts: a database with preprocessed data from different racquet sports, a backend that runs the core algorithms, and a web application for visualizations and interactions.

T2 **Display multiple attributes simultaneously.** Domain experts desire to obtain insights comprehensively by seeing multiple attributes of an event simultaneously. Studies on visual analytics of sports data prove that an intuitive glyph is preferred [13, 30, 43, 49] because it can use visual metaphors that can correspond to the real scene to show multiple attributes simultaneously. Moreover, the system should provide design templates of glyphs suitable for many sports.

T3 **Merge the patterns.** Domain experts prefer to discern different playing styles when analyzing a player. A playing style can be depicted by multiple similar patterns. For example, the successive use of technique Drive and that of technique Volley are similar, both of which reflect the offensive playing styles in tennis. Merging similar patterns based on domain knowledge helps domain experts evaluate a player's playing styles quickly. Therefore, the system should support the domain experts to merge similar patterns based on their knowledge.

T4 **Provide a comparison with multiple levels of detail.** Domain experts need to quickly compare a player with others to find his/her individual tactical patterns. They also need to compare the performance of a player in different situations to obtain insights about his/her playing styles comprehensively. Thus, the proposed system should allow users to divide the dataset into two subsets with different conditions, such as who serves and whether it is a crucial time. Then, interactive visualization is used to visually and interactively compare the two subsets and find the differences. However, comparing multivariate event sequences can lead to visual clutter because of the massive information. Therefore, the proposed system should support a comparison with multiple levels of detail.

T5 **Show detailed sequences within a pattern.** Domain experts may examine the raw data for detailed analysis and verification. Thus, a detailed view should be provided to show the raw data.

3.3 System Overview and Implementation

We introduce a generic visual analytics framework to support these analysis tasks. Users can define the weights of attributes (**T1**), tailor glyphs (**T2**), and group similar values according to their domain knowledge (**T3**). After users set the two subsets to be compared, the system uses a steerable MDL algorithm to detect sequential tactical patterns. The resulted patterns are then interactively visualized and are compared with multiple levels of detail (**T4**). Users can view detailed information on their demand (**T5**).

The framework consists of a database based on *MongoDB*, a backend based on *Flask*, and a web application based on *React* (*Fig.2*). The database stored data that is collected from different sports. The backend reads data from the database, communicates with the web application, and runs three core algorithms, namely, the MDL algorithm (*Sec. 4*), multiple dimensional scaling (MDS) algorithm (*Sec. 5.3.1*), and the one for information entropy (*Sec. 5.1*). The web application provides interactive visualizations for the domain experts to analyze the multivariate racquet sports data.

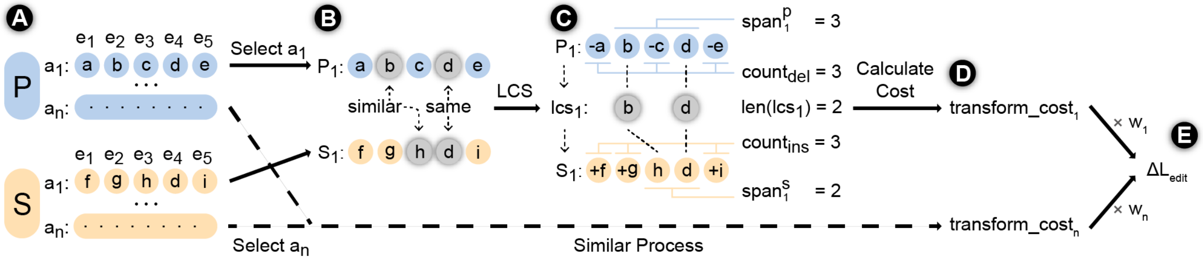

Figure 3: Example to showcase the process of our algorithm. The algorithm calculates the editing cost of mapping sequence S to pattern P. In the example, both S and P have five events, and each event has n attributes (A). The algorithm calculates the cost over each attribute. When traversing an attribute, such as a_1, we extract the value of the attribute from each event to form a univariate sequence S_1 and a univariate pattern P_1 (B). We use the LCS algorithm to find the common part of S_1 and P_1, namely lcs_1 (C). We use lcs_1 as a bridge to estimate the transform cost (D). Finally, we add the transform cost of each attribute with weight to obtain the editing cost (E).

4 MDL for Multivariate Event Sequences

This section introduces our MDL-based algorithm for multivariate sequential pattern mining, which has two main technical contributions. First, the task of mining patterns from multivariate event sequences is one of the most critical tasks in racquet sports data analysis. Our algorithm supports such a task effectively by solving three domain-specific issues, namely, 1) adjusting the weights of different attributes (**T1**), 2) merging similar patterns (**T3**), and 3) controlling the length/authenticity/continuity of the extracted patterns to meet domain's needs. To the best of our knowledge, our algorithm is novel and no other existing algorithms can solve all the three issues. Second, our algorithm represents a new general way for pattern mining of multivariate event sequences. It can be generalized to many applications for analysis and exploration of multivariate event sequences, rather than limited to racquet sports data analysis.

We improved an algorithm called *MinDL* [12], which employs the MDL principle to extract patterns in event sequences. *MinDL* uses a two-part representation to describe each sequence, namely, the extracted pattern and the corrections needed to reconstruct the original sequence from the pattern (*e.g.*, insertion and deletion of events). *MinDL* initially regards each sequence as a pattern without any corrections and continuously merges two patterns to obtain a new pattern with additional corrections. *MinDL* makes a trade-off between the description cost of the two parts to obtain appropriate patterns. However, *MinDL* cannot be directly applied to racquet sports data analysis as it cannot meet the above three issues. Thus, we introduce a new measure (*Algorithm 1*) to calculate the description cost, such that the domain requirements can be satisfied.

We introduced three pieces of additional information in our measure to address the three issues, respectively. **First**, the weight of each attribute. Analysts can input the weights of different attributes to adjust their focus of analysis (**T1**). Our measure considers each attribute with weight to enable the multivariate pattern mining. The weight of the $i-th$ attribute a_i is denoted as w_i.

Second, the groups of similar values. Experts can group similar values to merge similar patterns (**T3**). In our algorithm, within a certain attribute a_i, each value is mapped to one and only one group, where one group is a set of similar values. More specifically, each group $G_{categorical}$ in a categorical attribute is a set of discrete values that can be considered similar (e.g., the *Offensive* group includes all techniques to attack). The numerical attributes are quantized to discrete ranges, such as $G_{numerical} = \{v \mid min \leq v < max\}$, where min and max are the boundary value of the range.

Third, three parameters that control the characteristics of the extracted patterns. The following parameters control three features of the extracted patterns, which are essential for sports analysis.

- ***Cost***$_{ins}$ is defined to control inserted events (*i.e.*, events that appear in the original sequence but disappear in the mapped pattern), which further affect the **length** of patterns.
- ***Cost***$_{del}$ is defined to control deleted events (*i.e.*, events that appear in the pattern but miss in the original sequence), which reveal the **authenticity** of patterns.
- ***Cost***$_{con}$ is defined to control the discontinuous events (*i.e.*, two events adjacent in patterns but not adjacent in original sequence), which further affect the **continuity** of patterns.

Algorithm 1: Editing Cost Calculation

```
Input: $S$: the sequence
       $P$: the pattern
Output: $\Delta L_{edit}$: Editing Cost
$\Delta L_{edit} = 0$
$w_{total} = Sum(w_1, w_2, ...)$
for $i \leftarrow 1$ to $len(attributes)$ do
    $P_i$ := the sequence of $v_i$ in $P$
    $S_i$ := the sequence of $v_i$ in $S$
    $lcs$ = LCS($P_i$,$S_i$)
    $transform\_cost = transform(S_i, lcs, P_i)$
    $\Delta L_{edit}$ += $transform\_cost \times w_i / w_{total}$
return $\Delta L_{edit}$
```

Analysts can adjust the parameters to fine-tune the results. We evaluate the influence of the three parameters in *Sec. 6.3.2*.

Our measure uses these three pieces of information for editing cost calculation to extract tactical patterns that meet the requirements of domain experts. We introduce *Algorithm 1* and use an example (*Fig. 3*) to illustrate our measure. The measure takes sequence S and pattern P as input (*Fig. 3(A)*) and outputs cost ΔL_{edit} of mapping S to P (*Fig. 3(E)*). The core idea is to iterate through all attributes, calculate the cost over each attribute, and add them together with weights. We choose a_1 as an example to demonstrate this process. First, we extract value v_1 of each event to form a univariate sequence and a univariate pattern, denoted by S_1 and P_1 (*Fig. 3(B)*). In our example, we assume that b and h in P_1 and S_1 are similar (in the same $G_{categorical}$ or $G_{numerical}$), and P_1 and S_1 share the same value d. We then calculate the longest common subsequence (LCS) of S_1 and P_1, denoted as lcs_1. To merge similar sequences in one pattern (**T3**), we count both similar values and same values as the common part of S_1 and P_1. Thus, lcs_1 is bd (or hd). We further calculate the cost of a two-step transformation from S_1 to P_1 with lcs_1 as a bridge, named $transform\_cost_1$ (*Fig. 3(D)*). Finally, we multiply $transform\_cost_1$ with w_1 and accumulate the products of every attributes to obtain ΔL_{edit} (*Fig. 3(E)*).

We calculate $transform\_cost_1$ with the three parameters (*Fig. 3(A)*). Similar to Levenshtein distance, our measure also counts simple edits, such as insertion and deletion, to present editing cost $transform\_cost_1$. But our measure further use insertion/deletion cost to control the length/authenticity of patterns. Moreover, our measure detects where the event inserts/deletes to control the continuity of patterns. First, we locate the first event and the last event of lcs_1 in S_1 and P_1 and identify the span of lcs_1 in S_1 and P_1, namely, $span_1^s$ and $span_1^p$, which are 2 and 3, respectively, in the example. Then

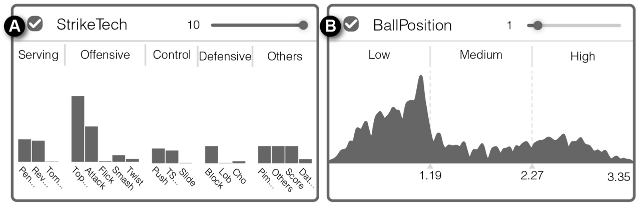

Figure 4: Distribution of values. (A) The distribution of categorical values is visualized by the bar chart. (B) The distribution of numerical values is visualized by the area chart.

we obtain $transform\_cost_1$ by adding the costs of three parts (*Eq. 4*). 1) We count the insertion events by $len(S_1) - len(lcs_1)$, which is 3 in the example, and multiply it by $Cost_{ins}$ to obtain the insertion cost (*Eq. 1*). 2) We count the deletion events by $len(P_1) - len(lcs_1)$, which is 3 in the example, and multiply it by $Cost_{del}$ to obtain the deletion cost (*Eq. 2*). 3) We count the events in-between lcs_1, breaking the continuity in two steps, by $span_1^s - len(lcs_1)$ and $span_1^p - len(lcs_1)$, which are 1 and 0 in the example, and multiply their sum by $Cost_{con}$ to obtain the continuity cost (*Eq. 3*).

$$\Delta L_i^{ins} = (len(S_i) - len(lcs_i)) \times Cost_{ins} \tag{1}$$

$$\Delta L_i^{del} = (len(P_i) - len(lcs_i)) \times Cost_{del} \tag{2}$$

$$\Delta L_i^{con} = (span_i^s - len(lcs_i) + span_i^p - len(lcs_i)) \times Cost_{con} \tag{3}$$

$$transform\_cost_i = \Delta L_i^{ins} + \Delta L_i^{del} + \Delta L_i^{con} \tag{4}$$

The proposed measure leads *MinDL* to extract tactical patterns with enough configurability to satisfy domain requirements. The results are compared in our system to identify players' tactical patterns.

5 System Design

Following the analysis tasks, we present the web application to support visual analytics with four views. *Attribute Editor* allows users to adjust the importance of different attributes and group similar values according to their domain knowledge (**T1**, **T3**). *Glyph Editor* displays the glyph design and provides simple interactions to edit it (**T2**). *Comparison Views* compares the two subsets with multiple levels of details (**T4**). *Instance View* shows the raw sequences (**T5**).

5.1 Attribute Editor

We design *Attribute Editor* (*Fig. 1(A)*) to adjust the weights of different attributes and group similar values (**T1**, **T3**). We use a list to show all attributes because the list is clear and concise for domain experts. For each attribute, we display the three-level data structure of "attribute → groups of similar values → values" (*Fig. 1(A2)*). At the top, the name of the attribute is shown on the left. The checkbox before the name controls whether the attribute is used in pattern mining and visualized in the system. On the right, a slider is used to control the weight of the attribute (**T1**). To recommend a suitable weight as default, we calculate the diversity of the attribute based on information entropy [36]. As the information entropy of an attribute increases, its amount of information also increases and thus should be assigned a higher weight. The diversity is encoded with the vertical line on the slider. Under the top line, all the groups are shown and can be renamed.

Users can expand the bottom to view detailed information of values in an attribute (*Fig. 1(A1)*) and group similar values (**T3**). The distribution of each attribute is visualized as either a bar chart or an area chart, depending on whether it is a numerical or categorical attribute (*Fig. 4*). In the bar chart, each bar encodes the frequency of a discrete value, and the bars of similar values are grouped. Users can drag the bar and drop on another group or an empty area to regroup values or create a new group. Similarly, the area chart encodes the distribution of numerical values, where several vertical lines split the whole X-axis into several ranges. Users can click on the area chart to add a line, right-click on the line to delete it, and drag the line to redefine the boundary of a range. To reduce users' operations, we provide default groups of similar values based on domain knowledge.

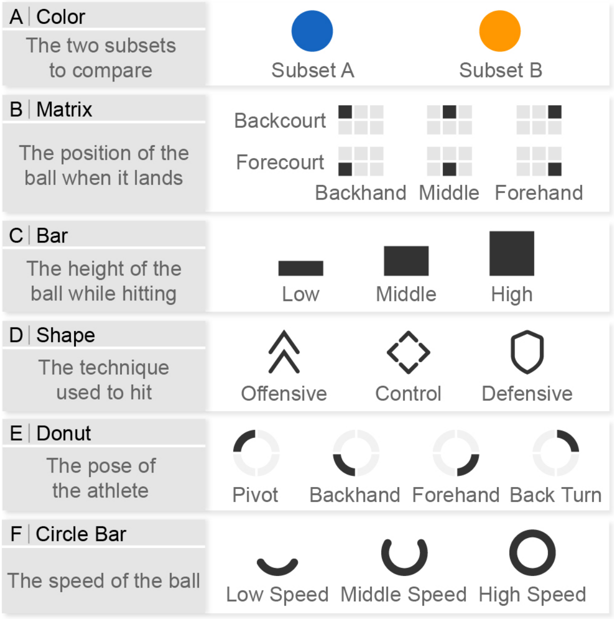

Figure 5: Six encoding methods. (A) Color encodes two subsets. (B) Matrix encodes the ball position. (C) Bar encodes the height of the ball. (D) Shape encodes the techniques. (E) Donut encodes the athletes' pose. (F) Circular Bar encodes the speed of the ball.

5.2 Glyph Design and Glyph Editor

We use glyphs to show multiple attributes of an event simultaneously (**T2**) because of their effectiveness [30] and the common usage for encoding multivariate sports data [13, 43, 49]. Our glyph designs use six encoding methods, namely, color, matrix, bar, shape, donut, and circular bar, to cover all the attributes in our datasets, following the design principles proposed by Chung et al. [14]. The encoding methods and their justifications are as follows.

- **Color** encodes the two subsets to be compared as color is effective for distinguishing different categories. The use of color follows the principle of *searchability* [14].
- **Matrix** encodes two-dimensional spatial information, such as the ball position and the players' position on the court. A grid in a matrix can indicate the position on the court intuitively, which follows the principles of *typedness* and *learnability* [14].
- **Bar** encodes numerical data with spatial information, such as the height of the ball. The height of the ball can be naturally mapped to the height of the bar, which follows the principles of *typedness*, *visual orderability*, and *learnability* [14].
- **Shape** encodes abstract information, such as the techniques and the spin of the ball. The metaphor shape, such as a shield for a defensive technique and an arrow indicating a spin direction, helps users recall its meaning, which follows the principles of *typedness* and *learnability* [14].
- **Donut** encodes categorical attributes, such as the poses of athletes. We used Donut to reserve the central space for more important attributes, which follows the principles of *typedness* and *attention balance* [14].
- **Circular Bar** encodes numerical attributes, such as the speed of the ball. Circular bar can also reserve the central space for more important attributes, which follows the principles of *typedness* and *attention balance* [14].

Following the fifth design guideline introduced by Borgo et al. [4], there are three ways to map attributes to visual channels in a glyph, namely the one-to-one, one-to-many, and many-to-one mappings. Our glyph design uses the one-to-one mapping based

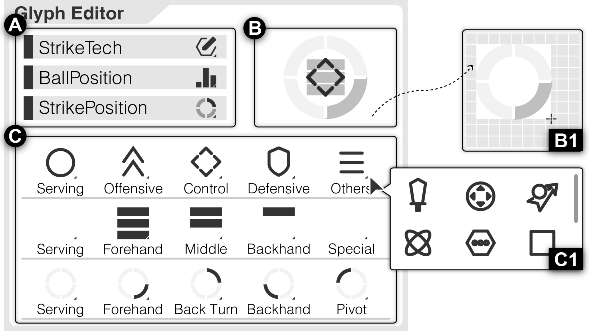

Figure 6: The *Glyph Editor*. (A) The attribute list. (B) The preview layout. (B1) Users can modify the layout of each attribute. (C) The symbol legends. (C1) An example of modifying the Shape encoding.

on two considerations. 1) *Design Complexity*. The one-to-many mapping encodes one attribute with multiple visual channels to emphasize its importance. However, our system allows users to change their focuses on different attributes, which leads to complex design adjustments and totally different glyphs. Moreover, the one-to-many mapping may not be affordable due to the many attributes but limited visual channels. 2) *Incomparability between attributes*. The many-to-one mapping encodes multiple attributes with one visual channel for inter-attribute comparison. However, the attributes in sports are incomparable, e.g., the position of the ball is a spatial attribute, while the technique of the hit is a categorical attribute.

We design *Glyph Editor* to configure tailored glyphs in different analysis situations (*Fig. 6*). *Glyph Editor* consists of the following three parts. 1) The attribute list (*Fig. 6(A)*) allows users to select one of the five symbols through the selector on the right for each attribute to be encoded. 2) The preview layout (*Fig. 6(B)*) shows the current glyph and becomes editable after choosing an attribute in the attribute list (*Fig. 6(B1)*). For each attribute, users can drag an area on the 10 × 10 grids to delimit the position where the attribute displays. 3) The symbol legends (*Fig. 6(C)*) work for choosing encoding for each group. Each row is a list of symbols for groups in one of the attributes, where rows have the same order as the corresponding attribute in the attribute list. Users can click on any symbol to modify it using an edit panel of the certain encoding method (*Fig. 6(C1)*).

To simplify user's operations, the system includes many templates of glyph designs and can automatically choose a suitable one according to the selection of attributes and the name of value groups.

5.3 Comparison Views

To compare the patterns in different scenarios (**T4**), we design two-level visualizations for comparing two subsets, namely, *Scatterplot* (*Fig. 1(D)*) and *Pattern Comparator* (*Fig. 1(C)*). *Pattern Comparator* is the most important view in our application, which provides one-to-one comparison of patterns between two subsets. Users start the comparative analysis and obtain valuable insights from *Pattern Comparator*. In contrast, *Scatterplot* is an assistant view that can provide a coarse-level overview of the patterns and filter patterns of interest.

First, users can add filters (*Fig. 1(C2)*) to construct two subsets, such as who serves and whether the rally is in a tiebreaker. The first button in *Fig. 1(C1)* is used to run the MDL algorithm (*Sec. 4*). Users can adjust the three parameters in the algorithm by clicking the fourth button in *Fig. 1(C1)* to expand a control panel. *Scatterplot* and *Pattern Comparator* update the extracted patterns for comparison.

5.3.1 Scatterplot

Scatterplot help users overview the patterns of two subsets and quickly focus on interesting ones. Projection technology can efficiently abstract data. Among the commonly used dimensionality reduction algorithm for feature projection, such as PCA, LDA, t-SNE, etc., we choose the metric MDS algorithm to project patterns for two considerations [3]. 1) Compared to PCA, LDA, and non-metric MDS, metric MDS can use the distance matrix to project multidimensional data instead of the Euclidean distance. We can hardly define the Euclidean distance between two patterns because patterns have different lengths, and some categorical attributes can hardly define Euclidean distance between values. By contrast, we can use the editing cost between each pair of two patterns to construct the distance matrix. 2) Although t-SNE can also accept a distance matrix, metric MDS is much efficient because t-SNE requires multiple iterations, but metric MDS does not.

We project patterns into a plane coordinate system, where each point represents a pattern, and the distance between them shows the similarity between the patterns. For each point, a pie chart encodes statistical data of the pattern, which helps users quickly compare the performance of a pattern in two subsets. The size of the pie encodes the frequency that a pattern appears in two subsets. The entire pie is divided into two sections, each of which represents a subset with the corresponding color. The radian of a section encodes the frequency of the pattern in the corresponding subset. The length of the arc outside the pie chart with color encodes the winning percentage of the pattern in the subset, where a semicircle represents all the winnings.

5.3.2 Pattern Comparator

We design *Pattern Comparator* to help experts compare patterns one-to-one, where juxtaposition is a commonly used method [21,34]. In our system, each subset occupies one column (*Fig. 1(C9)*), and each pattern occupies one row (*Fig. 1(C4)*), where patterns are arranged in descending order according to their frequency in the subset. For each pattern, a series of glyphs represent the events in the pattern and are placed from left to right in the order in which events occur. The color of the glyph encodes the player who hit the ball, where the color of the subset represents the server of the subset, and the dark color represents the opponents. Bar charts in the middle support comparison on frequency (*Fig. 1(C7)*). The height of a bar encodes the frequency of the corresponding pattern. The scale can be adjusted by the controller (*Fig. 1(C3)*) above the bar charts, where the two numbers on the side show the total frequency of the corresponding subsets. The luminance of the bar encodes the winning rate naturally because the winning rate is a numerical value ranging from 0 to 1 [37]. When users hover on a bar, the frequency and the winning rate are shown in a tooltip. To compare the frequency of a specific pattern (especially when its glyphs in two subsets are not at the same row, *e.g.*, C4 and C6 in *Fig. 1*) and further compare the distribution of pattern frequency, we design a line chart with two lines (*Fig. 1(C5)*). The solid line with the color of the subset connects the tops of all the bars on the hovered side. The dashed line with the color of the other subset shows the frequency of each pattern in the other subset. To avoid visual clutter, users need to hover on the area of bar charts to view the line chart.

5.4 Instance View

All the sequences with a certain pattern are shown in the *Instance View* (*Fig. 1(E)*) for detailed information (**T5**), when either the point or the row of glyphs is clicked. In the view, each row displays one sequence. For each row, the rectangle on the left encodes the outcome of the rally, where a solid one represents that the server wins, and a hollow one represents that the opponent wins. A circle in the row represents an event in the sequence. We use a solid circle to encode the event in the pattern and a hollow circle for the event not in the pattern. For a solid circle, the number on it represents the index of the event in the pattern.

5.5 User Interactions

We design user interactions to help users explore the data.

Common interactions. To analyze the context of an event, users can click on a glyph in *Pattern Comparator* to align patterns at this event. The alignment operation has two improvements to accommodate visual analysis of racquet sports. 1) With a lock (the second button in *Fig. 1(C1)*), users can choose to align only patterns of

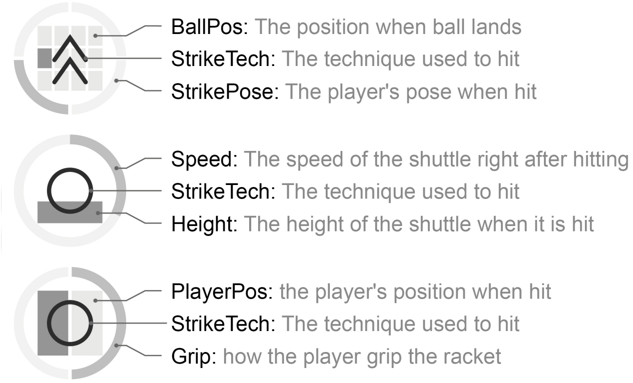
BallPos: The position when ball lands
StrikeTech: The technique used to hit
StrikePose: The player's pose when hit
Speed: The speed of the shuttle right after hitting
StrikeTech: The technique used to hit
Height: The height of the shuttle when it is hit
PlayerPos: the player's position when hit
StrikeTech: The technique used to hit
Grip: how the player grip the racket

Figure 7: Glyph design in two usage scenarios. Glyph designs for (A) the first usage scenario, (B1) the first step of the second usage scenario, and (B2) the second step of the second usage scenario.

the current subset or the two subsets synchronously, which enables comparing context. 2) In sports events, the ordinal index of an event has a different analysis value from its reverse index. Thus, the user can choose to align patterns at an event with an ordinal index or a reverse index (*Fig. 1(C8)*). To simplify the patterns, users can click on the merge button (the third button in *Fig. 1(C1)*) and select several similar patterns to be merged. After merging, the pattern with the highest frequency is retained, and the sequences of other patterns are merged into this pattern. To focus on interesting patterns, users can use the box selection on *Scatterplot* to filter interesting patterns and right-click to cancel selections.

Linked highlight. To guide users to switch between visualizations, we designed three linked highlights with shadows. 1) To focus on an event, users can right-click on a glyph in *Pattern Comparator* to highlight the same glyph in other patterns. 2) To focus on one pattern, the point in *Scatterplot* and two rows of glyphs in *Pattern Comparator* are linked. Hovering on any one, all three are highlighted. 3) To quickly find similar patterns, users can right-click on a pattern in *Scatterplot* to highlight patterns near to it.

Details on Demand. In *Pattern Comparator*, when users hover on a glyph, a tooltip is shown, which displays all the pairs of attributes and values in tabular form. In *Instance View*, the tooltip works in the same way when users hover on a circle. To obtain the complete detail of a sequence, users can click on the row to expand a table with the records of all events (*Fig. 1(E1)*).

6 Evaluation

We use two example usage scenarios to evaluate the usability and effectiveness of our system. We further hold interviews with the domain experts to gather the feedback. Finally, a laboratory experiment is introduced to evaluate our model.

6.1 Example usage scenarios

We deployed the system on the web and conducted two usage scenarios together with domain experts through online meetings to evaluate the usability of our system and showcase the insights it can bring to the domain. Each usage scenario lasts for about 20 minutes.

6.1.1 Usage scenario of Tennis

We conducted the first usage scenario together with two domain experts in tennis. One is a professor majoring in tennis, and the other is his Ph.D. student, who used to be a tennis athlete. The usage scenario explores how *Djokovic*, one of the best tennis players in the world, chooses tactical patterns. We analyzed five matches in Australian Open 2019 and French Open 2019 (all quarterfinals or later). In tennis, after deuces in a game, the first player to score in two consecutive rallies wins the game. This rule makes the match more intense and encourages athletes to adopt special patterns to win the game. Therefore, we filtered the sequences served by *Djokovic* and divided them into two subsets according to whether the rally is after the first deuce in a game. *Djokovic* served in 414 filtered sequences, of which the average length is 6.20, and the max length is 25. Among these sequences, 60 ones are after deuces, and 354 ones are before deuces. There exist 1320 distinct types of events in these sequences. Based on the sequences, we compare the performance of *Djokovic* with the first deuce as the dividing point.

To evaluate players' performance, four categorical attributes are used to describe each hit. We group similar values in each attribute under the guidance of domain experts. *BallPos* records where the ball fell on the court. We group similar locations based on the distance between them. *StrikeTech* is the technique used to hit the ball. Domain experts suggest that they can be classified into serve, offensive, defensive, control, and other techniques. *StrikePose* records players' pose while hitting. *Spin* shows how the ball rotates after the hit.

Attribute editing. Initially, the system calculated the diversity of four attributes and recommended the weight for each one, where *BallPos* is 10, *StrikePose* is 5, *StrikeTech* is 4, and *Spin* is 1. *BallPos* is the most important attribute because the court is so large that many patterns base on *BallPos*, such as consistent groundies (a player continuously hits the ball to the same place on his opponent's half-court) and side-to-side running. However, the domain experts adjusted the importance of *StrikeTech* to 7 and the one of *StrikePose* to 2. They explained that in most cases, *StrikeTech* could be limited to a few techniques, but once it changes, the pattern is well worth exploring. *StrikePose* is often casual because athletes often choose the most comfortable pose to hit the ball. Although many changes transpired, digging deeper is unnecessary. The domain experts canceled the selection of *Spin*, which is less important.

Glyph design. The domain experts accepted the template encodings without any adjustments. The glyph design uses a matrix to encode *BallPos*, which achieves the spatial mapping. The shape superimposed on the matrix encodes *StrikeTech*, and the donut outside encodes *StrikePose* (*Fig. 7(A)*).

Pattern comparing. Then, the domain experts turned to the *Comparison View*. First, they focused on the *Scatterplot* (*Fig. 8*(A)) and found an area with dense scatters, which represented the usual patterns. They found that, except patterns in the area, more scatters are with orange color, which encoded the after-deuces subset. They believed that *Djokovic* did change his strategies after deuces. Then, they observed the arc outside the pie, which encodes the rate of winning. They found that *Djokovic* had a high winning rate in most patterns after deuces, which meant that winning a game against him after deuces would be even harder. Domain experts speculated that the rich experience in decisive moments, the most potent competitive skills, and the strong self-confidence help Djokovic win the game after deuces, whether or not he serves. They further used the box selection to filter patterns of interest, and the *Pattern Comparator* updated the filtered patterns synchronously. In Pattern Comparator, domain experts found no patterns need to be merged manually because the algorithm had merged patterns well. When the experts found a pattern of interest, he would turn to the *Instance View* for detailed information. Two findings are shown as follows.

Finding 1. As shown in *Fig. 8(B)*, the domain experts selected the area with dense scatters. They wanted to know how *Djokovic* used usual tactical patterns. First, they observed the bar charts in *Pattern Comparator*. On the left, the before-deuces subset had a more even distribution on the top patterns, which meant *Djokovic* was more inclined to use different patterns in a balanced manner. However, on the right, he focused on a few patterns after deuces. Domain experts speculated that after deuces, *Djokovic* hoped to defeat the opponent through some unusual patterns. To kill in one hit through these unusual tactical patterns, he began to reuse a common pattern to cultivate the opponents' thinking set and reduce their vigilance, which involves psychology in sports.

Finding 2. As shown in *Fig. 8(C)*, the domain experts selected three patterns to observe, which were all unusual strategies occurring in the after-deuces subset. In the *Scatterplot*, they were close to each other and all far from areas with dense scatters. One of the three

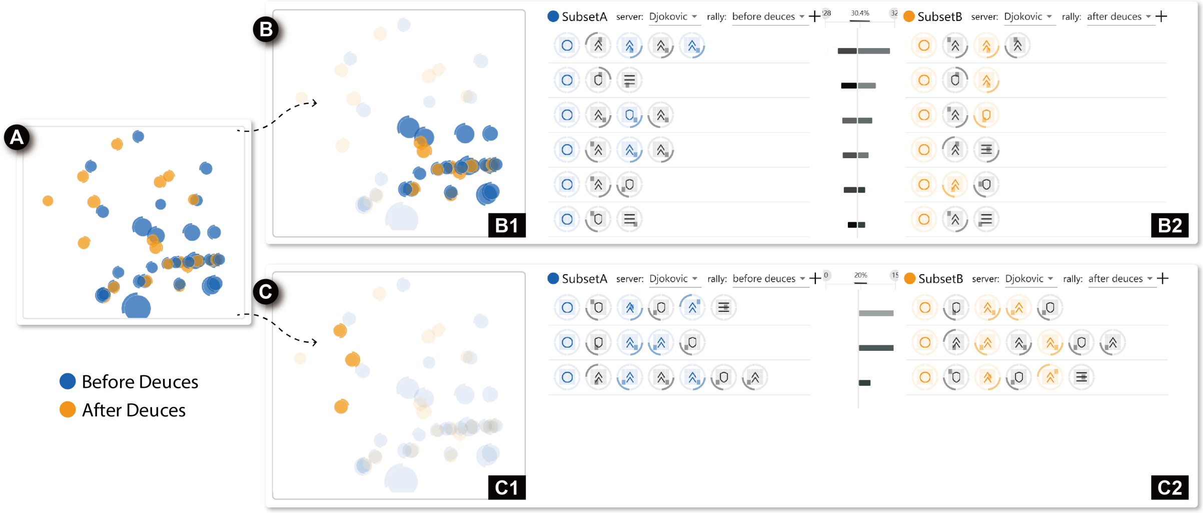

Figure 8: Screenshot for the usage scenario in tennis. Blue and orange encode rallies before and after deuces, respectively. (A) is the original scatterplot. (B) and (C) showcases the result when filtering the patterns in the area with dense points and the unusual patterns after deuces, respectively.

patterns had a low winning rate of 43%, whereas the other two had high ones of 83% and 100%. To find out the reason, they turned to the *Pattern Comparator*. They initially focused on *StrikeTech*. *Djokovic* kept using offensive techniques, and opponents had to defend, which was the same in the three patterns. After looking at *BallPos*, they found that *Djokovic* was forced to run side-to-side in all the three patterns. However, in the pattern with a low winning rate, *Djokovic* tried to use the tactical pattern of consistent groundies by pounding the ball to his opponents' backhand, which was a weakness of most athletes. In the two patterns with a high winning rate, he also inflicted the side-to-side running on his opponents. Domain experts thought that consistent groundies is a tactical pattern to attack the opponent's shot weakness. However, his opponents, who are also among the best athletes in the world, may have specially trained backhand techniques to compensate for the shortcomings. *Djokovic* incorrectly judged the opponent's weakness and made many mistakes while running. However, when both were running, *Djokovic*'s better athleticism earned him additional points.

Through the usage scenario, the domain experts concluded that *Djokovic* is an all-court player, who is good at using many tactical patterns to win after deuces. When he forces his opponents to run side-to-side, he can easily wear them down physically. However, he should use the pattern of consistent groundies with caution.

6.1.2 Usage scenario of Badminton

The usage scenario is conducted with two domain experts in badminton. One domain expert is a professor majoring in badminton, and the other is his Ph.D. student, who used to be a badminton player. In this usage scenario, we aim to find the personal strategies of a player by comparing him with other athletes. We choose *Momota Kento* as the case player and compare him with *Chen Long*. Our dataset contains five matches of each player on *Indonesia Master 2019*, *China Open 2019*, *The All England Open Badminton Championships 2019*, which are all quarterfinals or later. *Momota* served in 206 filtered sequences, of which the average length is 10.82, and the max length is 58. *Chen* served in 316 sequences, of which the average length is 11.79, and the max length is 48. There exist 180 distinct types of events in our usage scenario.

The data describe each hit with five attributes, and the similar values are also grouped under the guidance of domain experts. *StrikeTech* describes the technique used to hit the shuttle, which is categorical. Domain experts suggest classifying them into *Service*, *Forecourt*, *Backcourt*, *Midfield*, and *Others*. Unlike tennis, the technical classification in badminton is related to the distance between the player and the net. *Grip* describes whether the player hit the shuttle using forehand or backhand, which is categorical. *Height* is numerical data, which describes how high the shuttle is hit. *PlayerPos* describes where the player hit the shuttle, which is categorical. *Speed* is the speed at which the shuttle was just hit, which is numerical. Domain experts applied a three-level quantization to the numerical data *Height* and adjusted the range for each level. They also checked the quantization of the numerical data *Speed* and accepted it.

Technique-based patterns. First, the domain experts explored technique-based patterns, which revealed how the player hit the shuttle. They selected three attributes, *StrikeTech*, *Height*, and *Speed*, and then adjusted their importance to 10, 6, and 5, respectively. They further used shapes to encode *StrikeTech* in the center, a bar beneath the shapes with variable height to encode *Height*, and a circular bar outside to encode *Speed* (*Fig. 7(B1)*). The domain experts turned to the *Pattern Comparator* and observed the top patterns (*Fig. 9(A2)*). They found that the bars on the side of *Momota* were almost darker than the ones of *Chen*, which meant that *Momota* had a better performance on patterns commonly used than *Chen*. The bars on the side of *Momota* had a more similar length, which meant that *Momota* used more changeable patterns. Furthermore, they found that once *Momota* used a backcourt technique to hit a high shuttle with high speed, the pattern would have a higher winning rate. For *Chen*, however, this event was not always the case. Experts believed that because *Chen* is older, and therefore, his movements are limited. Then, they aligned the powerful hit. With the hover interaction, they found it interesting that, before the aligned hit, *Momota*'s opponent all used a frontcourt technique to hit the shuttle with a medium speed at medium height. Domain experts speculated that *Momota*'s overall strength was stronger than his opponent in these games, so he would try to use powerful backcourt techniques, such as *smash* or *drive* to win. To induce his opponent to hit the ball to a suitable position for his powerful attack, he hit the shuttle to frontcourt with medium height. To avoid *Momota* catching the shuttle in midfield or backcourt easily, domain experts thought that *Momota*'s opponent could use the techniques *high clear* and *lift*.

Space-based patterns. Then, the domain experts shifted their focus to spatial information, which revealed how the player ran. They added a selection of *Grip* and *PlayerPos* based on previous findings. They adjusted the importance of the two adding attributes to 4 and 8, lowered the weights of *Height* and *Speed* to 1, and maintained the one of *StrikeTech*, which is also spatially relevant, at 10. To adapt to the modification of the attributes, they made some changes to the glyph (*Fig. 7(B2)*). They hid the symbols for *Height* and *Speed*. A matrix and a donut were added to encode *PlayerPos* and *Grip*, respectively. The domain experts focused on

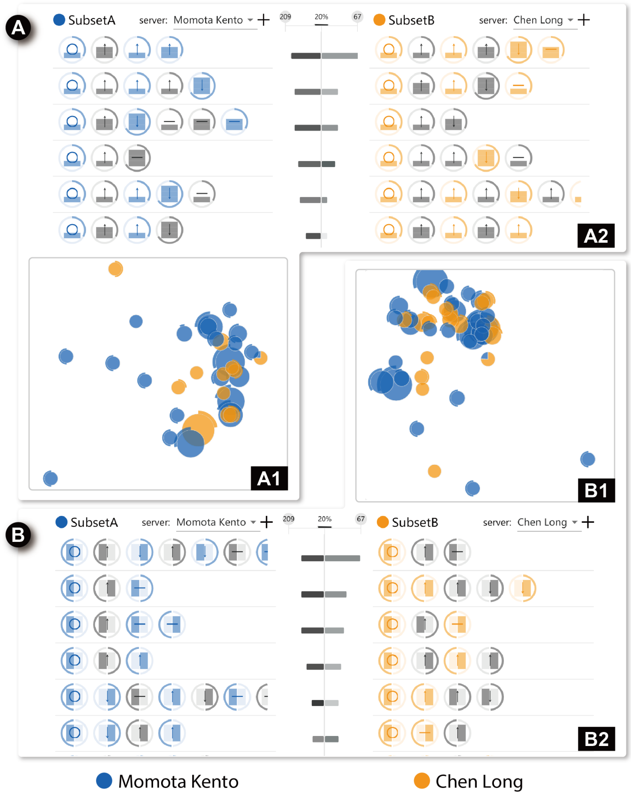

Figure 9: Screenshot for the usage scenario of badminton. Blue and orange encode *Momota* and *Chen*, respectively. The domain experts compare the (A) technique-based and (B) space-based patterns of *Momota* and *Chen*.

several patterns at the top of *Pattern Comparator* (*Fig. 9(B2)*). What interested them is that, in most patterns, *Chen* stayed at the right of the court and used the backhand grip to hit the shuttle. However, *Momota* hit the shuttle at the whole court with the full use of his backhand and forehand. Taking the first conclusion further, domain experts concluded that *Momota* could handle the opponent's attack from various directions. However, *Chen* mistakenly favored using the backhand to hit the ball in the right court, wasting his own forehand advantage, which resulted in a low winning rate.

Through the usage scenario, the domain experts concluded that *Momota* has a unique technical pattern to defeat his opponent and has few shortcomings at his forehand and backhand. By contrast, *Chen* can hardly copy Momota's tactical pattern because of his age, but he can improve his performance by using his forehand frequently.

6.2 Expert Interview

After the meetings for the usage scenarios, the web application was still available for the domain experts for one day. The experts used the tool to analyze the data of other athletes in their own sports domain. We further conducted an in-depth interview with the four domain experts one by one and asked questions about the usability and the suggestions. They all agreed that our system could solve the analytical problem in their own sports field, and they could gain new insights from the system. Their feedback is summarized as follows.

All four domain experts believe that our system brought them an innovative approach to analyzing data. Two experts felt that, with the adjustment of attribute weights, they could obtain insights comprehensively. Two experts praised the *Scatterplot*, which has saved them analysis time to find patterns with interest. Three domain experts thought that, with the *Pattern Comparator*, they could easily focus on the top-used patterns and find the difference between different athletes or moments. Two domain experts strongly recommended that we should link each sequence to the raw video. One expert suggested that some abnormal patterns should be detected and deleted automatically.

Method	AttrCnt	EvtCnt	SeqCnt	AvgSeqLen	PatCnt	AvgPatLen	T
MinDL + LSH	1	12	528	11.403	46	5.18	4.11s
Our algorithm	1	12	528	11.403	48	5.01	4.43s
	2	24	528	11.403	43	5.24	5.01s
	3	72	528	11.403	49	4.98	5.74s
	4	360	528	11.403	53	4.79	6.97s
	5	1080	528	11.403	60	4.62	8.56s

Figure 10: Algorithm performance comparison. AttrCnt is the count of attributes. EvtCnt is the count of distinct events. SeqCnt is the count of sequences in the dataset. AvgSeqLen is the average sequences length. PatCnt is the count of extracted patterns. AvgPatLen is the average patterns length. T indicates how long the algorithm runs, which is the average of five runs.

All four domain experts gave feedback about the glyph design. Two domain experts who have used the system iTTVis [49] can quickly configure tailored glyphs and believe that they are familiar with our intuitive glyphs. By contrast, for the two domain experts without prior knowledge about glyphs, they needed training to learn how to configure and tailor the glyphs. One domain expert pointed out that creating tailored glyphs was a difficult task for him. However, the two domain experts could also configure the glyph design after a quick training (about five minutes) since we simplified the configuration process in two ways. First, we provide a set of templates, which can save users' time and effort for designing glyphs. The domain experts said that thanks to the templates, they rarely needed to alter the glyphs when analyzing different attributes. Second, we provide six typical visual encodings to cover all attributes in our dataset. With these predefined encodings, the domain experts can easily create a suitable glyph for multiple attributes in an event. All four domain experts agreed that the glyph design contributed to gaining insights. According to their feedback, the glyph design had two advantages. 1) It can visualize multiple attributes simultaneously so that the domain experts can obtain insights without information loss. 2) Its intuitive encodings based on visual metaphors make it easy to recognize and understand, so that the domain experts can focus on the data analysis. For example, the shield encodes the defensive techniques, and the matrix encodes the ball position on the court.

6.3 Model Evaluation

We conducted two quantitative experiments to evaluate our model from two aspects, namely the runtime complexity and the effectiveness of the three parameters. The results demonstrated the performance of our algorithm.

6.3.1 Runtime Complexity

We compared the runtime complexity of our algorithm and MinDL through a controlled experiment (Fig. 10), where MinDL was implemented with LSH optimization following the paper [12]. We ran MinDL on a univariate dataset and ran our algorithm on one univariate dataset and four multivariate datasets (AttrCnt and EvtCnt in Fig. 10). For fair comparison, the six datasets shared the same sequences (SeqCnt and AvgSeqLen in Fig. 10). We ran algorithms on a P.C. with Intel Core i7 CPU with 4GB RAM.

The results of the experiment are shown as *Fig. 10*. As the count of attributes (AttrCnt) increases, the count of distinct events (EvtCnt) increases rapidly, but the time cost (T) increases slowly. Moreover, our algorithm can generate patterns with similar counts (PatCnt) and average length (AvgPatLen). The results prove that our new algorithm strikes a good balance between performance on time (T in Fig. 10) and functionality for multivariate pattern mining.

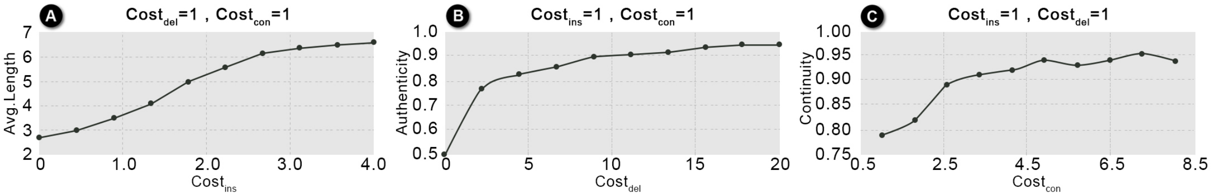

Figure 11: The results of model evaluation. The three line charts illustrate the relation between (A) the length of patterns and $Cost_{ins}$, (B) the authenticity of patterns and $Cost_{del}$, and (C) the continuity of patterns and $Cost_{con}$, respectively.

6.3.2 Effectiveness of The Three Parameters

We evaluated the effectiveness of the three adjustment parameters in our algorithm, including $Cost_{ins}$, $Cost_{del}$, and $Cost_{con}$, on the length, the authenticity, and the continuity of the patterns identified. The experiment uses the dataset from the first usage scenario (*Sec. 6.1.1*) with 414 filtered sequences, of which the average length is 6.20 and the max length is 25. The default weight of each attribute and the default groups of similar values are used for the experiment. We adjusted each parameter to 10 evenly distributed values in a reasonable range to evaluate its effect on the patterns identified with the other two parameters fixed. To calculate the length of patterns, we simply counted the average length of the extracted patterns. To estimate the authenticity of patterns, we evaluated the authenticity for each pattern and calculated the mean value. For the authenticity of each pattern, we counted the number of events that are also in the sequences mapped to the pattern and divided it by the length of the pattern. To estimate the continuity of patterns, we also evaluated the continuity and calculated the mean value. For each pattern, we divided the length of a pattern by the average span of the pattern in all sequences mapped to it as the estimated value of continuity.

The results in Fig. 11 present two findings. First, the three parameters can adjust the patterns as expected. The greater the $Cost_{ins}$ (0~4) parameter, the longer the length of patterns (2.8~6.5) is. The greater the $Cost_{del}$ (0~20) parameter, the higher the authenticity of patterns (0.51~0.94) is. When $Cost_{con}$ is between 1 and 8, the continuity of patterns is between 0.79 to 0.95. However, the $Cost_{con}$ parameter and the pattern continuity are not always positively related. When we increase $Cost_{con}$, the span of patterns in sequences (the denominator) decreases. However, continuous similar patterns are rare in real-world data. As a result, the length of patterns (the numerator) decreases at the same time. Thus, the estimated value (the quotient) may not necessarily increase. Thus, we limit the range of $Cost_{con}$ in our algorithm to avoid patterns of short length. Second, the relationship between the adjustment parameters ($Cost_{ins}$, $Cost_{del}$, and $Cost_{con}$) and the corresponding effect on the length, authenticity, and continuity of the patterns is not linear. The nonlinear relationship may confuse users when adjusting the parameters. Based on the second finding, we added a mapping function between the users' input and the three parameters to ensure a linear relationship between the users' input and the corresponding effect on the patterns.

7 Discussion

Tactical pattern analysis is essential for racquet sports. We propose a novel visual analytics framework for analyzing tactical patterns in racquet sports to help domain experts identify players' advantages and weaknesses. Through the usage scenarios, domain experts discovered the novel playing styles of players and counter strategies. This study also introduces the racquet sports data and the multivariate pattern mining algorithm to visual analytics of event sequences. This may inspire future studies to conduct a comprehensive analysis of multi-dimensional data. We will discuss the design lessons learned, generalizability, and limitations of this work as follows.

Design lessons learned. We have learned two lessons from this design study. First, we need to improve the algorithm efficiency to enable smooth user interactions, as users prefer to see the changes of patterns immediately after each adjustment. However, the original MDL algorithm is slow. Thus, we implemented the MDL algorithm in *C* to optimize its performance, which was greatly appreciated by the users. Second, metaphorical visual encodings are helpful for lowering the bar for domain experts to use a visual analytics system. We started this system design with several abstract glyphs to summarize the data. However, users cannot quickly recognize and understand the glyphs. Thus, we employ metaphorical visual encodings to design the glyphs. For instance, we employ a shield for defensive technique and a matrix for spatial data.

Generalizability. In the expert interviews, one expert, who also has knowledge about soccer, believes that our work can also be extended to the broader field of sports, such as basketball and soccer. In these sports, data can also be modeled as event sequences, *e.g.*, each shot and pass can be regarded as an event, and the sequential passes from one team getting the ball to the last shot can be regarded as a sequence. Tactic analysis can be applied to identifying and analyzing the personal strategies of each player and the team's coordination.

Limitations. First, domain experts in sports prefer to observe multiple attributes simultaneously to obtain comprehensive insights. However, the massive information leads to visual clutter. Although we try to reduce visual clutter through multi-level abstraction, the analysts still need to observe about 20 patterns. We plan to collect user behavior data to improve our system to reduce visual clutter. Second, the low efficiency of the MDL algorithm can hardly support the pattern extraction from a large dataset. We need to implement some optimization or find a more efficient alternative algorithm.

8 Conclusion

This paper presents a generic visual analytics framework to analyze multivariate event sequences in racquet sports. First, we summarize the general requirements in different racquet sports and use event sequences to model the data from different sports. Furthermore, we propose a multivariate sequential pattern mining algorithm with adjustable weights over multiple event attributes. To display multiple attributes in an event simultaneously, we implement six general encoding methods and combine them to construct an intuitive glyph. We then propose comparative visualizations with multiple levels of detail to identify the differences between the tactical patterns of different athletes or different moments. We also conduct two usage scenarios in tennis and badminton and collect the feedback through expert interviews to demonstrate the effectiveness of the proposed approach.

In the future, we plan to extend our framework to a wider range of sports. For example, in basketball and soccer, tactical analysis is also valuable to gain insights. We can also model the data as event sequences and extract the tactical patterns. Another interesting task is to find the hits resulting in a winning sequence. The pattern mining algorithm can find the hits shared by many sequences but may miss important hits. We plan to apply causality analysis to finding the causality between each hit and the results.

Acknowledgments

The work was supported by National Key R&D Program of China (2018YFB1004300), NSFC (61761136020), NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization (U1609217), Zhejiang Provincial Natural Science Foundation (LR18F020001) and the 100 Talents Program of Zhejiang University.

References

[1] N. A. S. Abdullah. Expert system for dota 2 character selection using rule-based technique. In *Advances in Visual Informatics: 6th International Visual Informatics Conference, IVIC 2019, Bangi, Malaysia, November 19–21, 2019, Proceedings*, vol. 11870, p. 318. Springer Nature, 2019.

[2] R. Bertens, J. Vreeken, and A. Siebes. Keeping it short and simple: Summarising complex event sequences with multivariate patterns. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 735–744, 2016.

[3] I. Borg and P. J. Groenen. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005.

[4] R. Borgo, J. Kehrer, D. H. Chung, E. Maguire, R. S. Laramee, H. Hauser, M. Ward, and M. Chen. Glyph-based visualization: Foundations, design guidelines, techniques and applications. In *Eurographics (STARs)*, pp. 39–63, 2013.

[5] D. Borland, W. Wang, J. Zhang, J. Shrestha, and D. Gotz. Selection bias tracking and detailed subset comparison for high-dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):429–439, 2019.

[6] I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White. Model-based clustering and visualization of navigation patterns on a web site. *Data Mining and Knowledge Discovery*, 7(4):399–424, 2003.

[7] T. Calders, C. W. Günther, M. Pechenizkiy, and A. Rozinat. Using minimum description length for process mining. In *Proceedings of the ACM symposium on Applied Computing*, pp. 1451–1455, 2009.

[8] B. C. Cappers, P. N. Meessen, S. Etalle, and J. J. Van Wijk. Eventpad: Rapid malware analysis and reverse engineering using visual analytics. In *IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, 2018.

[9] B. C. M. Cappers and J. J. van Wijk. Exploring multivariate event sequences using rules, aggregations, and selections. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):532–541, 2017.

[10] 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 the 8th MIT Sloan Sports Analytics Conference, Boston, MA, USA*, vol. 28, p. 3, 2014.

[11] 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.

[12] Y. Chen, P. Xu, and L. Ren. Sequence synopsis: Optimize visual summary of temporal event data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):45–55, 2017.

[13] D. Chung, P. Legg, M. Parry, I. Griffiths, R. Brown, R. Laramee, and M. Chen. Visual analytics for multivariate sorting of sport event data. In *Workshop on Sports Data Visualization*, vol. 3, 2013.

[14] D. H. Chung, P. A. Legg, M. L. Parry, R. Bown, I. W. Griffiths, R. S. Laramee, and M. Chen. Glyph sorting: Interactive visualization for multi-dimensional data. *Information Visualization*, 14(1):76–90, 2015.

[15] Z. Deng, D. Weng, J. Chen, R. Liu, Z. Wang, J. Bao, Y. Zheng, and Y. Wu. Airvis: Visual analytics of air pollution propagation. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):800–810, 2019.

[16] C. Dietrich, D. Koop, H. T. Vo, and C. T. Silva. Baseball4D: A tool for baseball game reconstruction & visualization. In *IEEE Conference on Visual Analytics Science and Technology*, pp. 23–32, 2014.

[17] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Eventaction: Visual analytics for temporal event sequence recommendation. In *IEEE Conference on Visual Analytics Science and Technology*, pp. 61–70, 2016.

[18] J. A. Fails, A. Karlson, L. Shahamat, and B. Shneiderman. A visual interface for multivariate temporal data: Finding patterns of events across multiple histories. In *IEEE Symposium On Visual Analytics Science And Technology*, pp. 167–174, 2006.

[19] P. Fournier-Viger, J. C.-W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng, and H. T. Lam. The SPMF open-source data mining library version 2. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 36–40. Springer, 2016.

[20] P. Fournier-Viger, J. C.-W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas. A survey of sequential pattern mining. *Data Science and Pattern Recognition*, 1(1):54–77, 2017.

[21] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. Visual comparison for information visualization. *Information Visualization*, 10(4):289–309, 2011.

[22] K. Goldsberry. Courtvision: New visual and spatial analytics for the NBA. In *MIT Sloan sports analytics conference*, vol. 9, pp. 12–15, 2012.

[23] R. Guo, T. Fujiwara, Y. Li, K. M. Lima, S. Sen, N. K. Tran, and K.-L. Ma. Comparative visual analytics for assessing medical records with sequence embedding. *Visual Informatics*, 2020.

[24] S. Guo, Z. Jin, D. Gotz, F. Du, H. Zha, and N. Cao. Visual progression analysis of event sequence data. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):417–426, 2018.

[25] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, and N. Cao. Eventthread: Visual summarization and stage analysis of event sequence data. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):56–65, 2017.

[26] Y. Ishikawa et al. Tidegrapher: Visual analytics of tactical situations for rugby matches. *Visual Informatics*, 2(1):60–70, 2018.

[27] J. Kiernan and E. Terzi. Constructing comprehensive summaries of large event sequences. *ACM Transactions on Knowledge Discovery from Data*, 3(4):1–31, 2009.

[28] M. Lage, J. P. Ono, D. Cervone, J. Chiang, C. Dietrich, and C. T. Silva. Statcast dashboard: Exploration of spatiotemporal baseball data. *IEEE Computer Graphics and Applications*, 36(5):28–37, 2016.

[29] H. Lam, D. Russell, D. Tang, and T. Munzner. Session viewer: Visual exploratory analysis of web session logs. In *IEEE Symposium on Visual Analytics Science and Technology*, pp. 147–154, 2007.

[30] P. A. Legg, D. H. S. Chung, M. L. Parry, M. W. Jones, R. Long, I. W. Griffiths, and M. Chen. Matchpad: Interactive glyph-based visualization for real-time sports performance analysis. In *Computer Graphics Forum*, vol. 31, pp. 1255–1264. Wiley Online Library, 2012.

[31] Z. Liu, Y. Wang, M. Dontcheva, M. Hoffman, S. Walker, and A. Wilson. Patterns and sequences: Interactive exploration of clickstreams to understand common visitor paths. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):321–330, 2016.

[32] M. H. Loorak, C. Perin, N. Kamal, M. Hill, and S. Carpendale. Timespan: Using visualization to explore temporal multi-dimensional data of stroke patients. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):409–418, 2015.

[33] S. Malik, F. Du, M. Monroe, E. Onukwugha, C. Plaisant, and B. Shneiderman. An evaluation of visual analytics approaches to comparing cohorts of event sequences. In *EHRVis Workshop on Visualizing Electronic Health Record Data at VIS*, vol. 14, 2014.

[34] S. Malik, F. Du, M. Monroe, E. Onukwugha, C. Plaisant, and B. Shneiderman. Cohort comparison of event sequences with balanced integration of visual analytics and statistics. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pp. 38–49, 2015.

[35] S. Malik, B. Shneiderman, F. Du, C. Plaisant, and M. Bjarnadottir. High-volume hypothesis testing: Systematic exploration of event sequence comparisons. *ACM Transactions on Interactive Intelligent Systems*, 6(1):1–23, 2016.

[36] L. Masisi, V. Nelwamondo, and T. Marwala. The use of entropy to measure structural diversity. In *IEEE International Conference on Computational Cybernetics*, pp. 41–45, 2008.

[37] T. Munzner. *Visualization analysis and design*. CRC press, 2014.

[38] J. P. Ono, C. Dietrich, and C. T. Silva. Baseball timeline: Summarizing baseball plays into a static visualization. In *Computer Graphics Forum*, vol. 37, pp. 491–501. Wiley Online Library, 2018.

[39] A. Perer and F. Wang. Frequence: Interactive mining and visualization of temporal frequent event sequences. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pp. 153–162, 2014.

[40] 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.

[41] C. Perin, R. Vuillemot, C. D. Stolper, J. T. Stasko, J. Wood, and S. Carpendale. State of the art of sports data visualization. In *Computer*

Graphics Forum, vol. 37, pp. 663–686. Wiley Online Library, 2018.

[42] 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, 2019.

[43] 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.

[44] Y. Ren, L. Shen, G. Yang, and H. Chai. Space-utility analysis of badminton players in three-dimensional space—taking lin dan and viktor axelsen as an example. *China Sports Science*, (3):10, 2018.

[45] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of IEEE symposium on visual languages*, pp. 336–343, 1996.

[46] J. Stasko and E. Zhang. Focus+ context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of IEEE Symposium on Information Visualization*, pp. 57–65, 2000.

[47] K. Vrotsou, J. Johansson, and M. Cooper. Activitree: Interactive visual exploration of sequences in event-based data using graph similarity. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):945–952, 2009.

[48] 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, 2019.

[49] 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, 2017.

[50] Y. Wu, D. Weng, Z. Deng, J. Bao, M. Xu, Z. Wang, Y. Zheng, Z. Ding, and W. Chen. Towards better detection and analysis of massive spatiotemporal co-occurrence patterns. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[51] 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, 2018.

[52] 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(1):To appear, 2021.

[53] S. Ye, Z. Chen, X. Chu, Y. Wang, S. Fu, L. Shen, K. Zhou, and W. Yingcai. ShuttleSpace: Exploring and analyzing movement trajectory in immersive visualization. *IEEE transactions on visualization and computer graphics*, p. To appear, 2021.

[54] J. Yuan, C. Chen, W. Yang, M. Liu, J. Xia, and S. Liu. A survey of visual analytics techniques for machine learning. *Computational Visual Media*, 7(1), 2021. doi: https://doi.org/10.1007/s41095-020-0191-7.

[55] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. MatrixWave: Visual comparison of event sequence data. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 259–268, 2015.