HistoryTracker: Minimizing Human Interactions in Baseball Game Annotation

Jorge Piazentin Ono, Arvi Gjoka, Justin Salamon, Carlos Dietrich, Claudio T. Silva
New York University
Brooklyn, New York

ABSTRACT

The sport data tracking systems available today are based on specialized hardware (high-definition cameras, speed radars, RFID) to detect and track targets on the field. While effective, implementing and maintaining these systems pose a number of challenges, including high cost and need for close human monitoring. On the other hand, the sports analytics community has been exploring human computation and crowdsourcing in order to produce tracking data that is trustworthy, cheaper and more accessible. However, state-of-the-art methods require a large number of users to perform the annotation, or put too much burden into a single user. We propose HistoryTracker, a methodology that facilitates the creation of tracking data for baseball games by *warm-start*ing the annotation process using a vast collection of historical data. We show that HistoryTracker helps users to produce tracking data in a fast and reliable way.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**; *HCI design and evaluation methods*; *Empirical studies in HCI*; **Interaction design**; *Empirical studies in interaction design*;

KEYWORDS

Sports Analytics, Sports Tracking, Baseball, Hand Annotation

ACM Reference Format:
Jorge Piazentin Ono, Arvi Gjoka, Justin Salamon, Carlos Dietrich, Claudio T. Silva. 2019. HistoryTracker: Minimizing Human Interactions in Baseball Game Annotation. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4–9, 2019, Glasgow, Scotland Uk.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3290605.3300293

CHI 2019, May 4–9, 2019, Glasgow, Scotland Uk

ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300293

1 INTRODUCTION

Sports analytics changed the way sports are played, planned and watched. Furthermore, the demand for precise, accurate and consistent data is higher than it ever was. While teams and sport organizations rely on multiple sources of data, such as smart watches, heart rate monitors and sensing textiles [24, 26], tracking data produced by specialized tracking systems may be considered the primary source of data in professional sports. Modern tracking systems make use of specialized sensors, such as high-definition cameras, speed radars or RFID technology, in order to collect movement data with precise measumerements and high sampling rates [12, 32]. Some examples of commercial tracking technologies are Pitch F/X and ChyronHego for baseball [10, 36], and STATS Sport VU for soccer, basketball and American football [37].

Tracking systems produce a valuable stream of data for analysis by sports teams. However, implementing and maintaining these systems pose three major difficulties. 1) They are expensive: Major League Baseball's Statcast, for example, was an investment of tens of millions of dollars [42]. Such cost may not be a problem for professional sports teams and leagues, but they are likely unattainable for smaller organizations or amateurs. 2) The quality of the tracking data is often affected by multiple hard-to-control factors [2, 12, 17, 27], including changes in lighting, camera position in relation to the field, occlusion and small objects, all of which can result in missing or noisy data, and 3) these systems cannot be used to produce tracking data for historical plays. At the same time, commentators and analysts often reference older games during their analysis. However, if the game happened before the tracking system was implemented, it is not possible to quantitatively compare the plays.

Adding manual annotation is a promising direction to address these issues, and a number of studies have explored how human annotators can be used to create reliable sports data from scratch. Manual annotation can be done by a single annotator [6, 33, 35] or by a collection of annotators through crowdsourcing [27, 44]. While individual manual annotation can be a reliable source of tracking data, it puts a major burden into a single person. Meanwhile, crouwdsourcing systems can split the annotation process into many tasks that can be completed quickly. The downside of this approach is that a large number of volunteers might be necessary to produce reliable data.

In this paper, we propose a novel methodology for manual tracking of baseball plays that reduces the annotation burden and is more enjoyable to users, in comparison to manual annotation from scratch. Our approach reduces the time needed to produce reliable tracking data by warm-starting the annotation process: instead of annotating trajectories on an empty canvas, users modify existing trajectories to reflect the play they want to annotate. The term "warm-start" is borrowed from machine learning, where it means that the model training started from a better initial point [22]. More specifically, we quickly collect a summary of the play by asking a few easy-to-answer questions, and use this information to recommend a set of similar plays that have already been tracked and can be used as an initial approximation. Our method produces reliable annotations at a lower cost and can be used to annotate historical plays that would be otherwise lost for quantitative analysis. Our user studies show that warm-starting the annotation of baseball plays reduces the time needed to generate the hand-annotated tracking data and has an equivalent performance to manually annotating plays from scratch.

Contributions: Our contributions are three-fold: 1) we present a novel methodology for acquiring tracking data that is more reliable and faster than manual annotation from scratch. 2) we describe HistoryTracker, a system that implements our warm-starting methodology for baseball tracking. 3) we present our quantitative and qualitative results, showing that our method is able to produce reliable annotations in a shorter amount of time, and make the annotation process more enjoyable to the users.

The rest of the paper is organized as follows. Section 2 presents the related work on sports tracking and crowdsourcing for sports. Section 3 presents a brief overview of baseball. Section 4 presents our tracking methodology and the design choices we made to implement it in HistoryTracker. We evaluate our methodology in Section 5 and discuss our results in Section 6.

2 RELATED WORK

Tracking data is commonly used in a wide array of applications in sports, both for entertainment purposes and for expert analysis. In the United States, some of the major examples are Major League Baseball (MLB), National Football League (NFL) and National Basketball Association (NBA). Since 2015, MLB has been using its tracking infrastructure, MLB StatCast, to augment its broadcasting videos and generate new content to the public [17, 42]. NFL and NBA also deploy tracking technologies to augment their broadcastings and compute statistics for fans [13, 23]. Sports teams and leagues use tracking data to analyze and improve player performance and game strategies.

A vast collection of works in the literature show how tracking data can be used to inspect games in more detail: information visualization techniques enable the visual spatial analysis of games, while machine learning and statistics allow for predictions and inferences to be computed on games. Much of the recent work in sports visualization is based on trajectory data. Some examples in the include tennis [31], baseball [12, 17, 25], basketball [15, 34, 41], soccer [28, 38, 39], hokey [30] and rugby [8, 9]. While each of those works are adapted to better illustrate their respective sports, their main focus is on clearly conveying the trajectories, or metrics computed from trajectories, to the user.

Meanwhile, statistics and machine learning are used to make predictions and inferences on top of the sports tracking data. Ghosting, a technique that uses machine learning to compute optimal player trajectories and predict play outcomes, has been applied to basketball [33] and soccer [18] tracking data. Statistical analysis has been applied to basketball to evaluate players shooting ability and compare defensive strategies [7, 20]. Cross et al. [11] studied baseball tracking data to evaluate batter's hot and cold zones. Bialkowski et al. [4] used expectation maximization on soccer tracking data to detect play formations across time, and discovered that teams play differently at home and away, being more forward at home.

Currently, most of the sports tracking data produced by mainstream media are generated by automated methods. Commercial systems, such as Pitch F/X [36], ChyronHego TRACAB [10], and STATS Sport VU [37] are used at every game from major league sports teams, producing huge amounts of data for analysis. For a review on automatic tracking methodologies, please refer to the surveys by Santiago et al. [32] and Kamble et al. [16]. In the rest of this section, we focus on works related to manual annotation of sports games.

Before the development of automatic tracking systems, experts had to perform the annotation of players and ball position manually [32]. While professional sports leagues have shifted towards automated methods, they are very protective of their data, only sharing small aggregated statistics with the public. Therefore, manual tracking is still used when the data is not readily available, e.g., academic research and amateur teams [27, 29]. Spencer et al. [35] hand annotated hockey players movement and speed throughout multiple games in order to analyze how player performance changes during a tournment. Bogdanis et al. [6] hand annotated basketball games in order to compare the effects of training programs on players. The annotation was made offline, using video footage of the game and training sessions, and the experts had to collect and annotate both player trajectories and actions: e.g., dribbles, and offensive/defensive moves.

Crowdsourcing has also been used to generate sports data [27, 40, 43, 44]. Crawling twitter streams enable the extraction of game highlights, where hashtag peaks might indicate the most exciting moments in the game [40, 43]. While this technique does not produce tracking data, highlights are a valuable data source that can be gathered from a publicly available platform. Vondrick et al. [44] investigated the use of crowdsourcing interfaces to annotate basketball videos. The authors divided the work of labeling video data into micro-tasks that could be completed by a large number of human annotators, and showed that combining the output of of the multiple users resulted in more accurate tracking data. Perin et al. [27] followed the same principles, but extended this approach to enable the real-time annotation of games. In their system, each person is asked to annotate either one player or one event, and high accuracy was obtained by averaging annotations. While micro-tasks made the annotation process easier, it has the downside of requiring a large number of users to produce a single play annotation.

Another research area closely related to play annotation is called sports information retrieval. With massive amounts of sports tracking data being generated every year by automated systems, designing ways to organize and search sports data became a challenging task. Many methods have been developed to retrieve particular games based different types of queries, or constraints. Most of the sports information retrieval literature focus on retrieving games based on textual queries. Fleischman et al. [14] developed a language model for baseball game retrieval built on top of closed captions. Their system enabled users to query specific game events, as long as they were described by the commentator. The automatic classification of video footage can also be used to enable quick retrieval of games. Zhou et al. [45] proposed a basketball video classification system based on decision trees, and used this system to retrieve games based on a set of constraints, such as player position (right field, left field), scores, and types of offense/defense. One of the most advanced methods of retrieval is based on sketches: Sha et al. [34] designed a system for querying basketball games using sketches that use the top view of the field. While this method was very successful in retrieving similar trajectories, it has the downside of essentially asking a user to draw the entire trajectory of the play by hand.

3 BASEBALL OVERVIEW

While our annotation methodology is general and can be applied to other team sports, in this paper we focus on baseball.

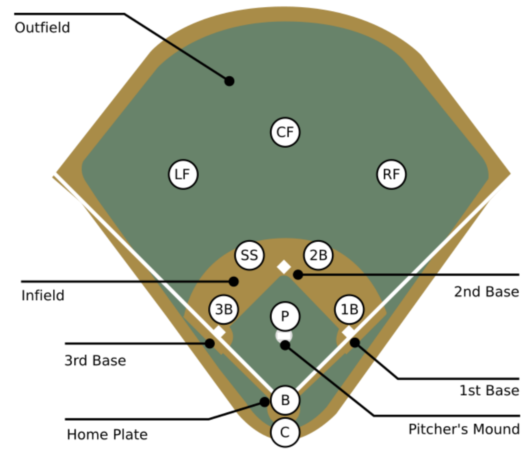


Figure 1: Baseball field schema.

Here, we briefly describe the sport and its basic rules. For more details, please see [3] and [21].

Baseball is a bat-and-ball game that is played on a field shaped like a circular quadrant, also called diamond. Four bases are placed at the corners of a ninety-foot square at the bottom of the diamond. The bases are labeled in counter-clockwise order starting at the bottom as home plate, first, second, and third base. The area right above the square is called *infield*, while the area above the dirt is called *outfield*. During the game, the teams alternate between the nine defensive and the four offensive roles. The defensive roles are the pitcher (P), the catcher (C), the basemen (1B, 2B and 3B), the short stop (SS) and the outfielders to the left (LF), center (CF) and right (RF). As the offensive roles, there are the batter (B) and zero to three runners on bases (R@1, R@2 and R@3).

Figure 1 shows a diagram of the field with the players located at their average positions. The runners are not shown in the picture for conciseness, but their starting positions are next to the first, second and third bases.

The game is divided in nine innings, each of which are split into two halves with teams taking turns on attack and defense. In general, a play starts when the pitcher makes the first movement and finishes when the ball returns to the pitcher's glove or goes out of play. Every player has a fixed initial position, and the set of actions they perform is relatively limited. Players in the offensive role try to touch all four bases in anti-clockwise order (1st, 2nd, 3rd and home plate). Meanwhile, players in the defensive role try to catch the ball and eliminate the attackers, before they are able to save bases and score runs.

Every offensive player starts at the batting position. A pitcher throws the ball at the batter, who then decides if he will swing and attempt to hit the ball, or take the pitch and let the catcher catch it. If he swings and hits, the batter becomes a runner and will try to save bases, touching each one them in counterclockwise order. Otherwise, if he misses, it counts as a strike. If he takes the pitch, the umpire can decide if the ball was valid (went through a strike zone). If it was, the batter receives a strike. Otherwise, the batter receives a ball. If a batter receives three strikes, he is out. If the batter receives four balls, he can walk to the first base safely. If the ball is hit and caught in the air by a defensive player, the batter is also out.

4 HISTORYTRACKER: TRACKING SYSTEM WITH WARM-START

Hand annotating sports from scratch is a difficult and time-consuming task commonly done offline by experts, who have to repeatedly watch recordings of the games in order to produce a good approximation of the players movement [32]. While more recent work has focused in transforming the annotation effort into micro-tasks distributed across a large number of crowdsourced annotators, this approach relies on a massive number of workers to produce reliable tracking data. We propose a methodology to enable quick single-user manual tracking of baseball plays, by introducing a warm-starting step to the annotation process.

Our approach consists of three steps: 1) *Fast play retrieval*: we present a video of the play of interest to the user, and ask them questions that they can quickly answer based on the footage (Figure 4A). This information is used to retrieve a collection of similar trajectories from the game corpus dataset (Figure 4C). 2) *Automatic tuning*: the user can refine the search by aligning the event icons with the events in the video and performing a temporal query in the data, i.e. a query indexed by the event times. We use the aligned events to automatically tune the retrieved trajectory and make it more similar to the video (Figure 4D). 3) *Refinement on demand*: the retrieved trajectory is used to warm-start the manual annotation and the user is asked to manually fix the trajectory where it does not match the video.

Below, we describe this methodology for play annotation.

Play Description and Fast Retrieval

In order to warm-start the annotation process, we search our historical trajectory dataset for plays with a similar structure as the one being annotated. In this paper, we used a query based approach similar to [45] in order to retrieve the similar plays. Our approach, however, is not based on video features, but historical tracking data instead. The broadcasting videos we take as input are focused on actions, and show only the players that have an impact on the play outcome. On baseball broadcasting, specifically, these actions usually include players contouring bases, throws, catches, tags, etc. The challenge is then to build a mapping from actions that

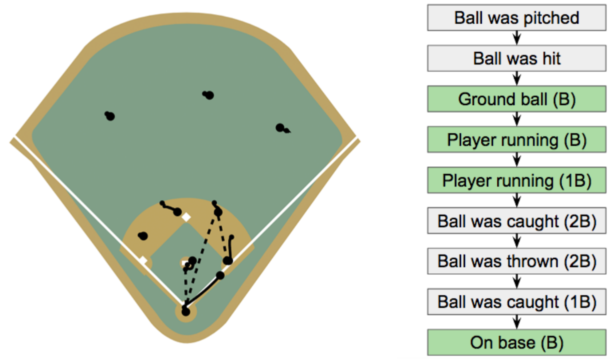


Figure 2: An example of a play (left) and the resulting set of events (right). The original set of events, shown in gray, is focused on the representation of the interaction between the players and the ball. We propose the representation of plays by an augmented set of events, shown in green, which encompass information about both the actions and the movements of the players.

may be identified on videos to a list of plays. These plays should be similar to the play from where the actions were identified, on both the actions performed and on the movements of the players. In order to implement such a mapping, we need a way to represent baseball plays by the actions that are performed by the players. In baseball, just like most sports, the tracking data of a play is given as a colletion of 2D time series data representising player movement, 3D time series of ball positioning, high-level game events and play metadata (see [12] for details).

The game events are pairs (action, player) that refer to specific actions that give context to the tracking data, like the moment the ball was pitched, hit, caught or thrown by a player, etc. By themselves, the game events offer a high level representation of the play that is close to what is necessary for building the query. This representation only lacks information about the geometry of the play (trajectories of the targets), which would help to narrow the search down to plays where the targets movement resembles what is observed in the video. We then propose an augmented set of events to represent plays, with new events that represent more details of the way the players move on top of the the original set of events as illustrated in Figure 2.

Once the play representation is defined, the straightforward approach would be to ask the user about the events that may be seen on the video (Figure 4A). The query is then built on questions that guide the user in the process of looking for the events that would lead to similar plays on the database. We have worked with baseball experts in order to select a group of questions that effectively summarize baseball plays: 1) Who ran? 2) Who are stealing bases? 3) What are the runners end bases? 5) Who caught the batted ball in flight?

6) Who threw the ball? and 7) What is the hit type? The questions are ordered by the impact on the overall trajectory data, allowing for a trajectory approximation to be generated as early as possible in the process. We accomplish this by first asking questions directly related to the play outcome (i.e. number of runs), and leaving play detail questions to the end. The set of events is then converted to a play index where each pair (event, target) is associated to a bit sequence, as illustrated in Figure 3.

This approach leads to the clustering of plays by similarity, given by the way the augmented set of events was designed. Since the augmented set of events contain information about both the actions and the geometry of the play, each cluster contains plays that are similar in both actions and geometry. The events and the clusters of plays were designed to accommodate small differences in the play geometry, in a trade-off between the amount of information that will be requested from the user for the query and the usefulness of the plays returned by the query. Empirically, the first play returned by the system is a good approximation of the actions and movements observed in the video. If the user chooses to inspect other plays in searching for a better one, the variability among them reduces the number of plays to be inspected.

The user query might result in an index for which there are no exact cluster matches in the database. In order to retrieve the most similar cluster to the user query, we select the cluster with the largest number of bits in common with the query. We also allow the users to increase the importance of some of the questions. For example, if the user wants to make sure only the selected players ran during the play, he can increase the weight of the question "Who ran?". Let n be the number of questions, Q be the query bits, W be the bits' weights, X be a cluster in the database, and $\mathbb{1}$ be the indicator function, the similarity between Q and X is given by:

$$S = \sum_{i=1}^{n} W_i \times \mathbb{1}(Q_i = X_i)$$

Automatic Trajectory Tuning Based on Play Events

After a description of the play is collected using our query interface (Figure 4A), our system recommends a cluster of trajectories that respects the specified event constraints. A random trajectory within this cluster is displayed to the user, as shown in Figure 4C. If this trajectory does not represent correctly the play, the user has three options: 1) change the weight of the questions in Play Description, in order to retrieve a better cluster for the play. 2) Click the switch button (top left corner of Figure 4C) to select another random play from the cluster. 3) Use the Events Chart (Figure 4D) to query this cluster based on event times.

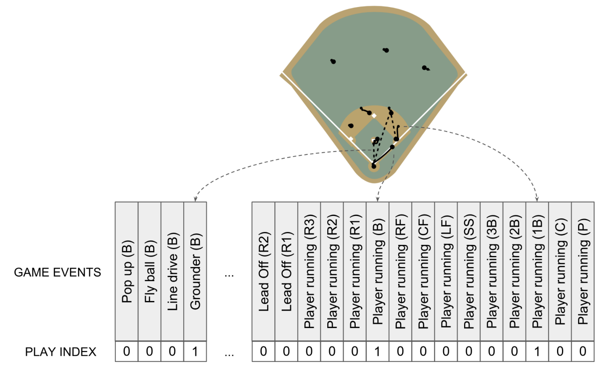


Figure 3: The events of a play are converted to a index, where each bit is associated to a pair (event, player). The index is the key to retrieve similar plays, where the similarity is given by how much information about the play is captured by this group of events.

In the Events Chart view, the main game events are displayed. In order to align the events of the trajectory data with the video, we use the *sound of the baseball hit in the video*. If the video contains a batting event (bat hits ball), we can detect the precise moment of the batting event in the audio signal and we can use this information to align the event data with the video content. To achieve this, we treat the problem as an audio onset detection problem under the assumption that the batting event corresponds to the strongest onset (impulsive sound) in the audio signal. We use the superflux algorithm for onset detection [5] as implemented in the librosa audio processing library [19] to compute an onset strength envelope representing the strength of onsets in the signal at every moment in time. For the analysis we use a window size of 512 samples and a hop size of 256 samples, where the sampling rate of the audio signal is 44,100 Hz, leaving all other parameters at their default values. To evaluate the approach, we manually annotated a validation set of 311 audio recordings with the timestamp of the batting event, and compared the output of our detection method to the annotations, where we consider the output to be correct if it is within 100 ms of the annotated value. Applying the approach the audio recordings achieved an accuracy of 94.5%, which we deem appropriate for our application. If the video does not have a batting event, we let the user perform the event alignment manually.

The user can drag and drop game events across the time axis and query for a play that respects the time at which these events happened. Once the user starts dragging an event, an image with the current video frame will be positioned over the user's mouse, enabling him to identify exactly when this event happened in the play. For example, if the user wants to specify the time at which the ball was caught and search

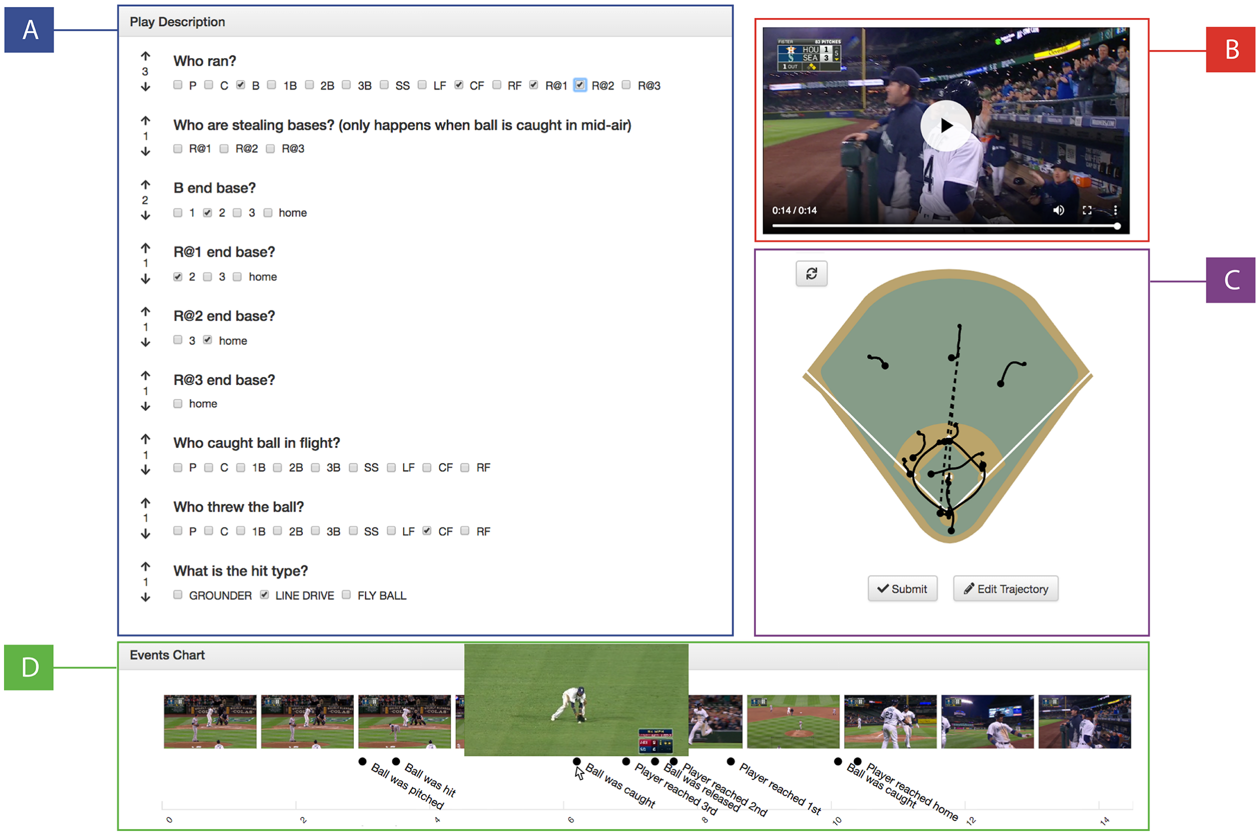


Figure 4: HistoryTracker system. A: Users can create a description of the play based on simple questions. B) A video of the play to be annotated. C) Trajectories are recommended based on the play description provided by the user. D) Events can be used to create a more fine grained query of the play.

for this event in the cluster, he should drag the event "Ball was Caught" in the Events Chart so that it aligns with the player action in the video (Figure 4D).

Our system adapts the retrieved trajectory so that it respects the event "Ball was pitched" in the Events Chart. In order to do so, we shift the retrieved trajectory so that the pitched event matches the one specified in the trajectory view. This action is a simple trajectory preprocessing step, but it allows us to quickly align the begin-of-play on the retrieved trajectory with the begin-of-play in the video.

By querying the data with the Events Chart, the user can obtain a better initial trajectory to warm-start their annotations. After this step is completed, the user has two options: If they are satisfied with the retrieved trajectory and think it perfectly matches the play in the video, they can click the submit button to save the new trajectory to the disk. Otherwise, the user can click the "Edit Trajectory" button and manually change the positions of players or ball that do not reflect the elements in the video.

Refinement on Demand: Manual Annotation

We implemented a hand annotation system that allows users to edit and refine the previously recommended trajectories (Figure 5). The system is comprised of four parts: A) the video playback screen; B) the play diagram in which the user annotates the current player position; C) the video playback slider; and D) the tracking element selector.

The trajectory annotation process is straightforward. The user positions the video at a frame of interest (keyframe) using the playback slider (Figure 5C), and marks the player position in the field by selecting the same position in the play diagram (Figure 5B). Consecutive keyframes are linearly interpolated, generating the tracking data. After the annotation of a player / ball is completed, the user can annotate the next player by selecting it in the Tracking element selector (Figure 5D).

If the user determines that the warm-start trajectory for an element is wrong, they can click on the button "Clear Trajectory" to delete the keyframes from the current element trajectory and start the annotation again.

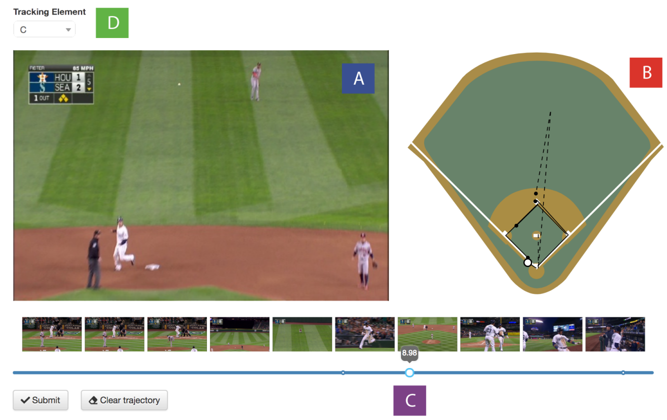


Figure 5: Manual tracking system. A: video playback screen. B) Play diagram for position input. C) Video playback slider. D) Tracking element selector.

A demonstration video of the HistoryTracker system is available in the supplementary material.

5 EVALUATION

In order to evaluate our annotation methodology, we compared it to manual tracking with no warm-start, hereby called Baseline. Ten plays were selected to be used in our evaluation: we attempted to maximize the variability of the play configurations in our sample, regarding the number of players, events, and outcomes. Our system was evaluated with 8 users: half of them (type A) annotated the odd plays using HistoryTracker and the even plays using Baseline, while the other half (type B) did the inverse (even plays using HistoryTracker, odd plays using Baseline). In total, 80 play annotations were produced, 40 with HistoryTracker and 40 with the Baseline. Users were recruited through email lists; the only condition for participation was having followed baseball for a minimum of one year. The age of the users varied from 19 to 39, with the majority being in their 20's. Although most were not involved with baseball professionally (most of our users were students and researchers) they all professed to having a deep understanding of baseball. In the screening questionnaire, the users reported having a knowledge of baseball of 8.12 +/- 1.35, on a scale of 1 to 10.

In this section, we analyze the tracking results with respect to the tracking error and the annotation time. We also perform a qualitative analysis of the system and present the results of a likert scale questionnaire we applied to the users after they performed the plays annotation.

Analysis of Plays

In this section, we present the ten plays that were used for the generation of tracking data using HistoryTracker and Baseline. All plays were used for both annotation methods, with each user annotating the same play only once. Figure 8 shows the ten plays using Baseball Timeline [25], a spatio-temporal visualization that represents the position of the players with respect to how close they are to the bases, as well as ball possessions, throws and hits. In this visualization, player position is represented in the Y axis and time, in the X axis.

[Play 1] The first play is from the fourth inning of the Philadelphia Phillies versus the Atlanta Braves, June 6, 2017. Batter Maikel Franco grounds the ball softly towards first base. First baseman Matt Adams catches the ball, runs to first base and the batter is out. This is a relatively simple play.

[Play 2] This play is from the third inning of the Chicago White Sox vs Detroit Tigers game, June 27, 2015. Alexei Ramirez hits a ground ball, second baseman Ian Kinsler catches it, throws to first baseman Miguel Cabrera and the batter is out.

[Play 3] Ninth inning of the Texas Rangers versus the Colorado Rockies, August 9, 2016. There are runners on first and second base. Batter Gerardo Parra grounds the ball out, where it is caught by second baseman Rougned Odor who throws it to first baseman Mitch Moreland to get the batter out. Runners at first and second advance one base.

[Play 4] Third inning of New York Yankees versus the Minnesota Twins, July 26, 2015. Baseman Mark Teixeira flies the ball out to right fielder Torii Hunter and reaches first base. The runner on second base reaches third base.

[Play 5] Sixth inning of New York Yankees versus Baltimore Orioles, June 13, 2015. Batter Didi Gregorius hits a ground ball. Second baseman Ryan Flaherty catches the ball, throws it to shortstop J. Hardy. The Runner at first is out at 2nd base. Batter saves first base.

[Play 6] is from the fourth inning of Chicago Cubs versus the Houston Astros, September 10, 2016. This is a very unique play. With Jose Altuve batting, runner at first Alex Bregman steals second base. Catcher throws the ball the the second baseman.

[Play 7] Houston Astros versus New York Yankees, third inning, October 17, 2017. Batter Austin Romine grounds the ball, which is caught by third baseman Alex Bregman who throws it to first baseman Yuli Gurriel. Runner at first Todd Frazier reaches second base.

[Play 8] is from the Seattle Mariners versus the New York Yankees, sixth inning, August 27, 2017. Batter Starlin Castro hits a line drive towards the center of the outfield, where it is caught by center fielder Guillermo Heredia. Batter reaches first base.

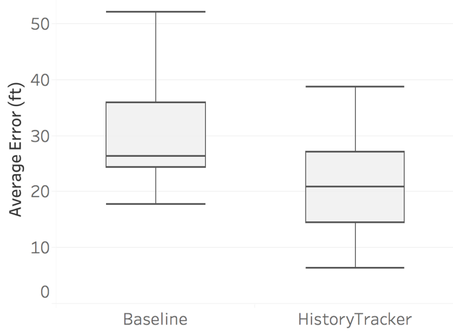


Figure 6: Error comparison between HistoryTracker and manual tracking from scratch (Baseline).

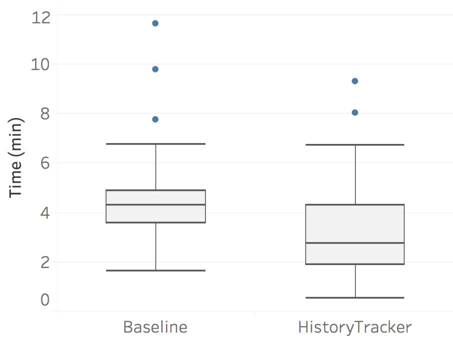


Figure 7: Time comparison between HistoryTracker and manual tracking from scratch (Baseline).

[Play 9] First inning of the Pittsburgh Pirates versus the Colorado Rockies, September 24, 2015. Batter Starling Marte grounds the ball towards left fielder Rafael Ynoa. Runner at second Jaff Decker reaches home plate. Runner at first Andrew McCutchen reaches second base while the batter reaches first base.

[Play 10] The final play is from the seventh inning of the San Diego Padres versus the Baltimore Orioles, June 22, 2016. Batter Yangervis Solarte hits a soft line drive towards right fielder Mark Trumbo, whose fielding error allows second baseman Matt Kemp to reach home plate. The batter reaches 1st base.

Quantitative Analysis

In this section, we compare the quality and annotation time of the tracking data produced by HistoryTracker and the tracking data acquired with the Baseline manual annotation tool. In order to compare the quality of the annotations, we computed the average Euclidean distance of the annotations to the ground truth produced by MLB Statcast, by averaging the distances between the annotated position and the ground truth position at every sampled time step. Denoting the set of play elements (players and ball) as E and the set of times over which the positions are sampled as T, we have the error of the annotated trajectory $\hat{x}$ with respect to ground truth x as:

$$Error = \frac{\sum\limits_{e \in E} \sum\limits_{t \in T} ||x_t^e - \hat{x}_t^e||}{|E| \times |T|}$$

The comparison between error results of HistoryTracker and Baseline are shown in Figure 6. HistoryTracker performs significantly better than the Baseline, exhibiting about 20% lower median error, with the same amount of spread. Furthermore, about 40% of the tracking data generated by History Tracker has a lower error than the lowest error generated by the Baseline.

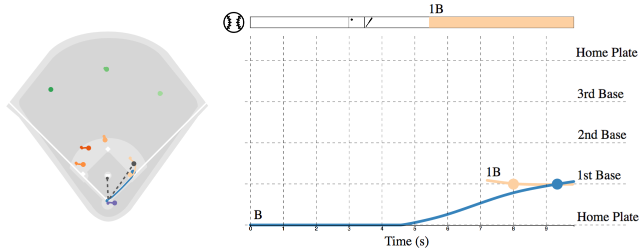
We also compared the play annotation time of HistoryTracker and Baseline. Figure 7 shows the time taken to complete the annotations of every play with both tools. The median time taken to annotate using the HistoryTracker was about 1.5 minutes less than with the Baseline. Overall, our user study indicates that warm-starting play annotations make users more efficient and accurate.

User Feedback

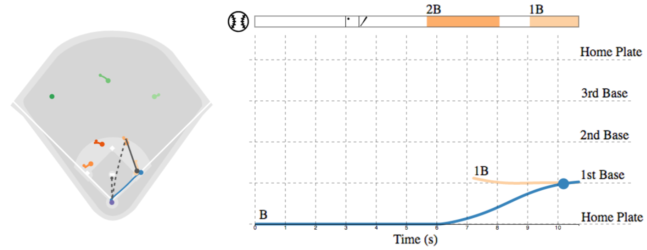
After the users performed the play annotations with the Baseline and HistoryTracker, they were asked to fill a likert scale questionnaire, which contained statements regarding their perception of both systems. The users could rank the statements from 1 (strongly disagree) to 5 (strongly agree). The statements and the answers from all the users are presented in Table 1.

Our first two statements evaluated how confident users were about their annotations, regarding both HistoryTracker and Baseline. We can see that the users are equally confident on using both systems. However, the third statement shows that all users perceived HistoryTracker to produce the annotations in a shorter amount of time, compared to Baseline. Regarding ease of use, the fourth and fifth statements show that the users thought both systems were equally easy to understand and use. Therefore, from a perceptual standpoint, using HistoryTracker makes the annotation process faster, with no loss in the perception of difficulty or quality of annotations.

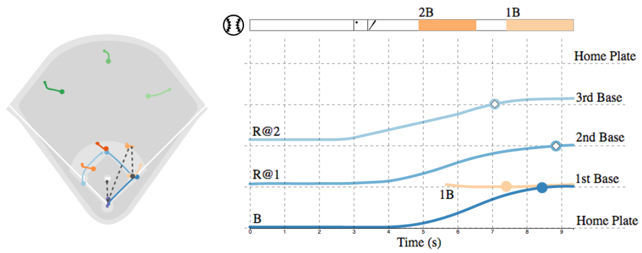
Overall, the HistoryTracker was well received by our users. Two annotators noted that HistoryTracker offered a good initial approximation of the plays. Another user mentioned that they spent less time annotating common movements, such as pitching and the batter running to first base, because a lot of the work was filled in already.



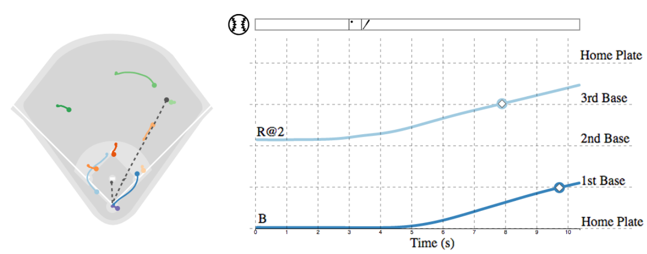
(a) [Play 1] Ground ball to first baseman.



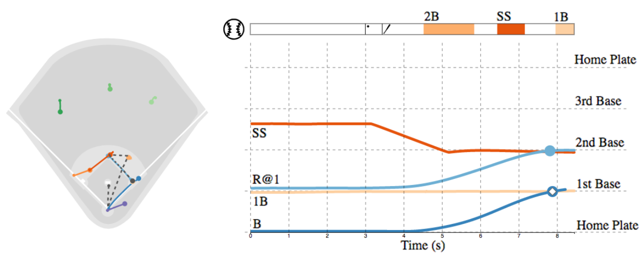
(b) [Play 2] Ground ball to second and first basemen.



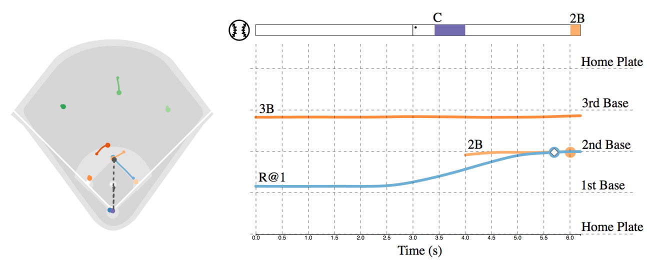
(c) [Play 3] Runners reach second and third bases.



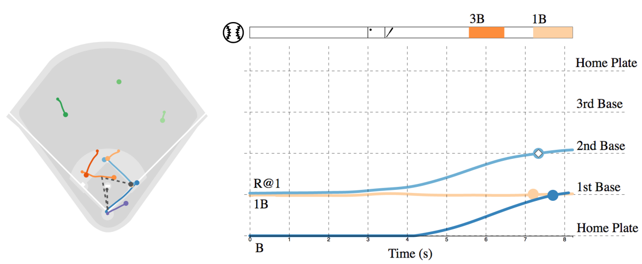
(d) [Play 4] Batter and runner at second advance one base.



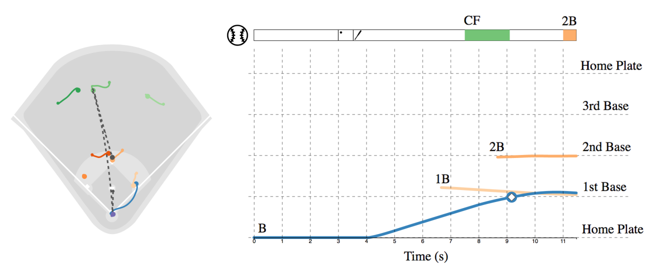
(e) [Play 5] Runner at first out. Batter advances one base.



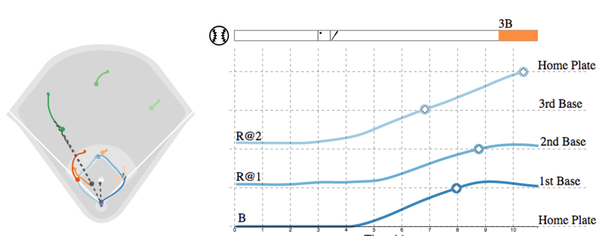
(f) [Play 6] Runner at first steals second base.



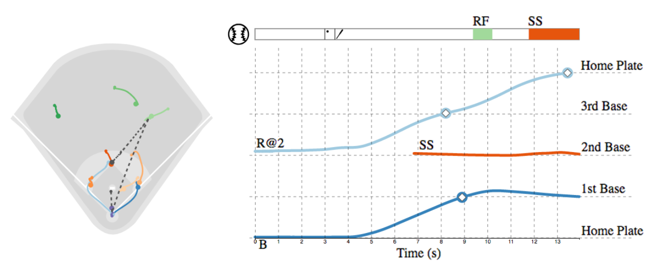
(g) [Play 7] Batter out. Runner at first steals second.



(h) [Play 8] Line drive to center fielder.



(i) [Play 9] Three players advance bases. Runner at second scores a run.



(j) [Play 10] Batter advances to first base. Runner at second scores a run.

Figure 8: Graphical representation of the evaluation plays using Baseball Timeline [25]. The X axis represents time, in seconds. The Y axis represents the position of the player relative to the bases. When a player reaches a base safely, he is marked with a ◇.

	Strong. Disagree	Disagree	Neutral	Agree	Strong. Agree
1) I feel confident about the tracking data I produced using HistoryTracker.				6 (75%)	2 (25%)
2) I feel confident about the tracking data I produced using Baseline.			2 (25%)	5 (62.5%)	1 (12.5%)
3) I feel like HistoryTracker allowed me to annotate baseball plays in a faster manner, in comparison to Baseline.				3 (37.5%)	5 (62.5%)
4) I feel like HistoryTracker is easy to understand and use.			1 (12.5%)	3 (37.5%)	4 (50%)
5) I feel like Baseline is easy to understand and use.			2 (25%)	3 (37.5%)	3 (37.5%)

Table 1: User perception of the system, using a 5 point likert scale.

6 CONCLUSION

It is widely accepted that sports tracking data has been revolutionizing sports analytics with its unprecedented level of detail. Instead of relying on derived statistics, experts can use that data to "reconstruct reality" and create their own statistics or analysis without prior constraints [12]. Moreover, tracking data can be used for training "simulation engines", that can predict game developments and enable new hypothesis to be tested [33]. Unfortunately, this data can be expensive to acquire, either requiring multi-million dollar investments in infrastructure and services (e.g., MLBAM Statcast or the NFL's Zebra tracking system), or systems that use manual annotators, which tends to be tedious and require many passes to generate high-quality data.

In this paper, we propose to use knowledge already acquired to lower the cost of future data acquisition. This is intuitively a very simple idea. We present HistoryTracker, a tool that takes broadcast video from baseball games, and with much lower level of user input, is able to generate high-quality tracking data. The system automates many of the tedious tasks by leveraging information retrieval techniques on a corpus of previously acquired tracking data. We presented a tool tailored for baseball, but we believe HistoryTracker could be extended for other domains. Extending our tool for other sports, such as soccer and basketball, is straightforward: we only need a set of events to describe plays and some historical tracking data. One can imagine applying the warm-starting procedure to non-sports domains as well. For example, we can use historical information to help annotating semantic image segmentation datasets [1]. Pixel-wise image annotation is a time consuming task, so it would greatly benefit from our warm-starting methodology. As future work, we would like to investigate how to use historical data to initialize image annotation tasks.

Furthermore, we believe systems such as ours can be used in novel applications, for instance, annotating historical video collections can potentially be used for generating statistics for comparing how players performance changes over time; or the system can be used for college or high-school video collections, enabling parents (or coaches) to track the performance of players as they mature.

There are many opportunities for improvements. An obvious extension of HistoryTracker would be to make it into a crowdsourcing tool that could potentially be used during live events. Among the challenges, we would need to research the best way to integrate multiple people's input, including potentially providing an intelligent interface that would update as others make edits to the play. Supporting multiple sports is also another obvious extension. We would also like to explore introducing more intelligence into the system as to further simplify the role of the user.

ACKNOWLEDGMENTS

We would like to thank the CHI 2019 reviewers for their thoughtful comments and efforts towards improving our manuscript. We thank Neel Dey, Jean-Daniel Fekete, Raoni Lourenco, Oded Nov, and the members of the NYU Sports Analytics team for help and support throughout this project, and the AVIZ research team, INRIA Saclay, who hosted C. T. Silva during this work. MLB Advanced Media provided partial funding for the project and access to data. This research was partially supported by Labex DigiCosme (project ANR-11-LABEX-0045-DIGICOSME) operated by ANR as part of the program "Investissement d'Avenir" Idex Paris-Saclay (ANR-11-IDEX-0003-02); NSF awards CNS-1229185, CCF-1533564, CNS-1544753, CNS-1730396, and CNS-1828576; the Moore-Sloan Data Science Environment at NYU; J. P. Ono and C. T. Silva are partially supported by the DARPA D3M program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. This work was performed while J. P. Ono and C. T. Silva were NYU Provost's Global Research Institute (GRI) fellows at NYU Paris.

REFERENCES

[1] Alexander Klaser. 2010. LEAR - Image Annotation Tool. https://lear.inrialpes.fr/people/klaeser/software_image_annotation

[2] R. Arthur. 2016. MLB's Hit-Tracking Tool Misses A Lot Of Hits. https://fivethirtyeight.com/features/mlbs-hit-tracking-tool-misses-a-lot-of-hits/

[3] Major League Baseball. 2016. *Official baseball rules*. http://mlb.mlb.com/mlb/official_info/official_rules/official_rules.jsp

[4] A. Bialkowski, P. Lucey, P. Carr, Y. Yue, and I. Matthews. 2014. Win at Home and Draw Away: Automatic Formation Analysis Highlighting the Differences in Home and Away Team Behaviors. *MIT Sloan Sports Analytics Conference* 28.

[5] S. Bock and G. Widmer. 2013. Maximum Filter Vibrato Suppression for Onset Detection. In *Proc. of the 16th Int. Conf. on Digital Audio Effects (DAFx). Maynooth, Ireland (Sept 2013)*. 7.

[6] G. C. Bogdanis, V. Ziagos, M. Anastasiadis, and M. Maridaki. 2007. Effects of two different short-term training programs on the physical and technical abilities of adolescent basketball players. *Journal of Science and Medicine in Sport* 10, 2 (April 2007), 79–88.

[7] D. Cervone, A. D'Amour, L. Bornn, and K. Goldsberry. 2014. POINTWISE: Predicting points and valuing decisions in real time with NBA Optical Tracking Data. *MIT Sloan Sports Analytics Conference* 28.

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

[9] D. H. S. Chung, M. L. Parry, I. W. Griffiths, R. S. Laramee, R. Bown, P. A. Legg, and M. Chen. 2016. Knowledge-assisted ranking: A visual analytic application for sports event data. *IEEE Computer Graphics and Applications* 36, 3 (2016), 72–82.

[10] ChyronHego. 2016. TRACAB Optical Tracking. http://chyronhego.com/sports-data/tracab

[11] J. Cross and D. Sylvan. 2015. Modeling spatial batting ability using a known covariance matrix. *Journal of Quantitative Analysis in Sports* 11, 3 (2015), 155–167.

[12] C. Dietrich, D. Koop, H. T. Vo, and C. T. Silva. 2014. Baseball4D: A tool for baseball game reconstruction amp; visualization. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*. 23–32.

[13] ESPN. 2012. Player tracking transforming NBA analytics. http://www.espn.com/blog/playbook/tech/post/_/id/492/492

[14] M. Fleischman and D. Roy. 2007. Unsupervised Content-based Indexing of Sports Video. In *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval (MIR '07)*. ACM, New York, NY, USA, 87–94.

[15] K. Goldsberry. 2012. Courtvision: New visual and spatial analytics for the NBA. In *MIT Sloan Sports Analytics Conference*.

[16] P. R. Kamble, A. G. Keskar, and K. M. Bhurchandi. 2017. Ball tracking in sports: a survey. *Artificial Intelligence Review* (Oct. 2017).

[17] M. Lage, J. P. Ono, D. Cervone, J. Chiang, C. Dietrich, and C. T. Silva. 2016. StatCast Dashboard: Exploration of Spatiotemporal Baseball Data. *IEEE Computer Graphics and Applications* 36, 5 (Sept. 2016), 28–37.

[18] H. M. Le, P. Carr, Y. Yue, and P. Lucey. 2017. Data-Driven Ghosting using Deep Imitation Learning. *MIT Sloan Sports Analytics Conference*, 15.

[19] B. McFee, C. Raffel, D. Liang, D. Ellis, M. McVicar, E. Battenberg, and O. Nieto. 2015. librosa: Audio and Music Signal Analysis in Python. In *Proceedings of the 14th python in science conference*. 8.

[20] A. McIntyre, J. Brooks, J. Guttag, and J. Wiens. 2016. Recognizing and Analyzing Ball Screen Defense in the NBA. *MIT Sloan Sports Analytics Conference*, 10.

[21] P. E. Meltzer and R. Marazzi. 2013. *So You Think You Know Baseball? A Fan's Guide to the Official Rules* (1 edition ed.). W. W. Norton & Company, New York, NY.

[22] K. P. Murphy. 2012. *Machine learning: a probabilistic perspective*. MIT Press, Cambridge, MA.

[23] NFL. 2018. Glossary | NFL Next Gen Stats. https://nextgenstats.nfl.com/glossary

[24] S. Nylander, J. Tholander, F. Mueller, and J. Marshall. 2014. HCI and Sports. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 115–118.

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

[26] T. Page. 2015. Applications of Wearable Technology in Elite Sports. *Journal on Mobile Applications and Technologies* 2, 1 (April 2015), 1–15.

[27] C. Perin, R. Vuillemot, and J. D. Fekete. 2013. Real-Time Crowdsourcing of Detailed Soccer Data. In *What's the score? The 1st Workshop on Sports Data Visualization*.

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

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

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

[31] G. Pingali, A. Opalach, Y. Jean, and I. Carlbom. 2001. Visualization of Sports Using Motion Trajectories: Providing Insights into Performance, Style, and Strategy. In *Proceedings of the Conference on Visualization '01 (VIS '01)*. IEEE Computer Society, Washington, DC, USA, 75–82.

[32] C. B. Santiago, A. Sousa, M. L. Estriga, L. P. Reis, and M. Lames. 2010. Survey on team tracking techniques applied to sports. In *2010 International Conference on Autonomous and Intelligent Systems, AIS 2010*. 1–6.

[33] T. Seidl, A. Cherukumudi, A. Hartnett, P. Carr, and P. Lucey. 2018. Bhostgusters: Realtime Interactive Play Sketching with Synthesized NBA Defenses. *MIT Sloan Sports Analytics Conference*, 13.

[34] L. Sha, P. Lucey, Y. Yue, X. Wei, J. Hobbs, C. Rohlf, and S. Sridharan. 2018. Interactive Sports Analytics: An Intelligent Interface for Utilizing Trajectories for Interactive Sports Play Retrieval and Analytics. *ACM Transactions on Computer-Human Interaction* 25, 2 (April 2018), 1–32.

[35] M. Spencer, C. Rechichi, S. Lawrence, B. Dawson, D. Bishop, and C. Goodman. 2005. Time-motion analysis of elite field hockey during several games in succession: a tournament scenario. *Journal of Science and Medicine in Sport* (2005), 10.

[36] Sportvision. 2016. PITCHf/x. http://www.sportvision.com/baseball/pitchfx

[37] STATS. 2016. SportVU Player Tracking | STATS SportVU Tracking Cameras. http://www.stats.com/sportvu/sportvu-basketball-media/

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

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

[40] A. Tang and S. Boring. 2012. # EpicPlay: Crowd-sourcing sports video highlights. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1569–1572.

[41] R. Theron and L. Casares. 2010. Visual Analysis of Time-Motion in Basketball Games. In *International Symposium on Smart Graphics*, Vol. 6133. 196–207.

[42] USAToday. 2015. Data deluge: MLB rolls out Statcast analytics on Tuesday. https://www.usatoday.com/story/sports/mlb/2015/04/20/data-deluge-mlb-rolls-out-statcast-analytics-on-tuesday/26097841/

[43] G. Van Oorschot, M. Van Erp, and C. Dijkshoorn. 2012. Automatic extraction of soccer game events from twitter. *Proceedings of the Workhop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2012)* 902 (2012), 21–30.

[44] C. Vondrick, D. Ramanan, and D. Patterson. 2010. Efficiently Scaling Up Video Annotation with Crowdsourced Marketplaces. In *European Conference on Computer Vision*. Springer, 610–623.

[45] W. Zhou, A. Vellaikal, and C. Kuo. 2000. Rule-based Video Classification System for Basketball Video Indexing. In *Proceedings of the 2000 ACM Workshops on Multimedia (MULTIMEDIA '00)*. ACM, New York, NY, USA, 213–216.