Crowd-based Semantic Event Detection and Video Annotation for Sports Videos

Fabio Sulser Ivan Giangreco Heiko Schuldt
Department of Mathematics and Computer Science
University of Basel, Switzerland
{firstname.lastname}@unibas.ch

ABSTRACT

Recent developments in sport analytics have heightened the interest in collecting data on the behavior of individuals and of the entire team in sports events. Rather than using dedicated sensors for recording the data, the detection of semantic events reflecting a team's behavior and the subsequent annotation of video data is nowadays mostly performed by paid experts. In this paper, we present an approach to generating such annotations by leveraging the wisdom of the crowd. We present the CrowdSport application that allows to collect data for soccer games. It presents crowd workers short video snippets of soccer matches and allows them to annotate these snippets with event information. Finally, the various annotations collected from the crowd are automatically disambiguated and integrated into a coherent data set. To improve the quality of the data entered, we have implemented a rating system that assigns each worker a trustworthiness score denoting the confidence towards newly entered data. Using the DBSCAN clustering algorithm and the confidence score, the integration ensures that the generated event labels are of high quality, despite of the heterogeneity of the participating workers. These annotations finally serve as a basis for a video retrieval system that allows users to search for video sequences on the basis of a graphical specification of team behavior or motion of the individual player. Our evaluations of the crowd-based semantic event detection and video annotation using the Microworkers platform have shown the effectiveness of the approach and have led to results that are in most cases close to the ground truth and can successfully be used for various retrieval tasks.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

Keywords

Crowdsourcing; Sports; Multimedia retrieval; Video annotation

CrowdMM'14, November 7, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2660114.2660119.

1. INTRODUCTION

In line with the advent of big data and the general urge to collect data in various aspects of life, data-driven analytics has also come upon sports. Especially in team sports, such data are used not only for the single athlete to track his/her own performance and to identify possible improvements, but also for coaches to analyze the collective behavior of the entire team in order to map out strategies and give advise on their collective behavior.

For collecting these data, large sportswear companies have started the deployment of sensors that capture the positional information of players and the ball in the field. However, these data are often not publicly available. Furthermore, the use of sensors only gives limited information on the semantics of the game actions. To truly be able to analyze a team's behavior in a sports match or even in an entire tournament, a detailed annotation and labeling of semantic events together with precise temporal and spatial information (at which position on the field and in which moment in time did a particular event take place) is necessary.

In this paper, we present CrowdSport, a system that makes use of the wisdom of the crowd for annotating video material of soccer games. For this purpose, CrowdSport allows paid micro-workers to annotate video data with team, event and temporal as well as positional information. In the task, the workers will see a short video snippet of five seconds and are asked to identify semantic events and mark the team, the time, and the position on the field of the event happening. To ensure a high quality of the data, the workers are at first rated on the basis of an assessment, which results in a trustworthiness score for each user. Furthermore, each sequence is annotated multiple times by different workers. The various annotations provided by different users are integrated –using the trustworthiness ratings– to a single, coherent data set. In the course of further completion of tasks, the user rating is adjusted based on the integrated data in comparison to the entry of the user.

We use the CrowdSport application in combination with the Microworkers[1] platform. The semantic annotations obtained from the CrowdSport application are later on used as basis for sophisticated retrieval functionality leveraging the team's collective behavior and/or the motion of individuals that is offered to game analysts and coaches [1, 2]. For instance, the data gained from the CrowdSport application is used in [1] for retrieving video scenes for which the user specifies e.g., a location or a motion path.

[1] `http://www.microworkers.com`

The evaluations of the crowd-based semantic event detection and annotation using the Microworkers platform have shown the effectiveness of the CrowdSport approach and have led to results that are in most cases close to the ground truth which is given by manual annotations of expert users.

The remainder of this paper is structured as follows: Section 2 discusses related work. In Section 3 we present the CrowdSport approach in detail. Section 4 discusses the results of the evaluation. Finally, Section 5 concludes.

2. RELATED WORK

There exists a large body of research on the (semi-) automatic feature extraction on sport videos, in particular soccer videos. In [3], Hidden Markov Models are used for the detection and recognition of semantic events in soccer matches. This approach is used for events such as penalties, free kicks and corners. In [12], blob tracking and local temporal spatio-velocity transform is used for tracking the position of players on the field. [6] exploits various low-level image features to detect events and perform a summarization of the soccer video. [4] presents a system that infers in real-time events from positional sensor data.

The wisdom of the crowd has been used since many years for image annotation. The ESP game [15] is one of the most prominent examples. The main motivation of users to contribute to the ESP game is a combination of playfulness and contest as they compete with each other when annotating images or when labeling parts of images. [13] uses crowdsourcing for tracking motion and annotating human gestures. [10] present OCTAB, an online crowdsourcing tool for annotating behaviors within video scenes, and MM-Eval, a procedure to evaluate the validity of annotations coming from the crowd in comparison to annotations by experts. [18] use a web-based system to annotate objects, motion, activities, etc. in videos and build up a video database with rich annotations. Most recently, in [16], the authors present the results of a three years study on large scale video annotation using Amazon Mechanical Turk.

In the area of sports, crowdsourcing has been used to collect detailed data of a live soccer game [11]. The users can enter –while watching the game– data on the location of players, name the player that is in possession of the ball, or qualify ball passes. Rather than using the application for annotating the game in a post-hoc analysis by workers, it is meant to be used during the game by active watchers. The authors of [14] use live information from social media of the watching crowd for annotating a sports video to the end of creating interesting summarizations. In [17], crowdsourcing is used for tracking actors or objects, i.e., the players, the referee, or the ball, in key frames of a basketball video and interpolate the positions in between.

3. THE CROWDSPORT APPROACH

The CrowdSport web application provides a platform that allows on the one hand workers to perform the annotation task (on-line phase), on the other hand it takes over the task of integrating and cleaning the annotation data entered by all the workers (off-line phase). Figure 1 depicts the CrowdSport system, also in interaction with the Microworkers platform: A worker from Microworkers is directed from the published task to the CrowdSport application. When doing so, the worker carries an identifier, which allows the system to track the worker's performance over time and to systematically evaluate her on the basis of a ground truth consisting of already labeled data set (manually provided by a domain expert). In the CrowdSport application, the user is assigned a short video sequence of five seconds for annotation. The results collected from the user are stored for later off-line processing. In the off-line phase, after all tasks are completed for a specific video sequence, the data entered by multiple users is used in an extraction, transform, and load (ETL) process, in which the data are cleaned and integrated so that they eventually represent a concise set of events. Finally, the newly generated data are used for updating the trustworthiness score of the user by comparing the latter to her submission; depending on the user's performance the task is accepted and the payment freed for the worker.

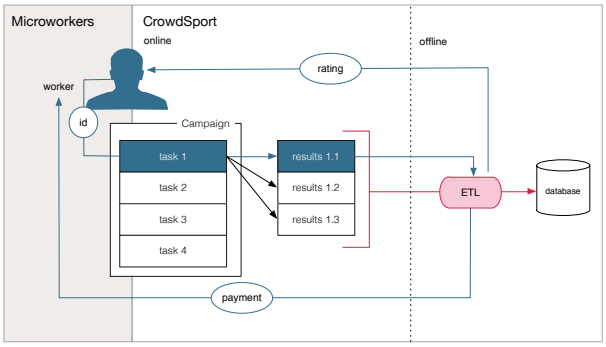

Figure 1: Overview of the CrowdSport system.

3.1 Task Structure

The task given to the worker requires her to annotate a short video snippet of five seconds. The beginning and the end of the snippet are assigned by the system and the user is only able to enter data to events within the time frame given to the worker. However, to get a more complete picture of the events happening and their local context, the user is able to move further back or forward within the video and watch previous scenes.

We require the user to enter the following information repeatedly for each event she has identified: In the first step, the user is asked to move within the video snippet in steps in the video frames of 0.1 seconds to the precise *moment in time* an event has happened. For the event, the user is asked to choose from the following possible semantic *event types*: goal, pass, tackle, shot on target, shot off target, interception, foul, out, corner kick, and/or penalty card. Moreover, the user is asked to denote the *team* that is involved in the event. Finally, we ask the user to mark the precise *location* (on the field) the event is taking place. All different event categories which are rated individually are subsumed as set event rating, ER, with $ER = \{\text{type}, \text{team}, \text{pos}, \text{time}\}$.

To avoid that a single user can annotate a full sequence only by herself, and ultimately to increase the quality of the data, each user is only allowed to perform one single task (i.e., annotating only a five second snippet) within a campaign. In our setting, a campaign constitutes about one minute of video.

Figure 2 shows a screenshot of the user interface to enter the data. Note that the GUI has been designed for interactive use and, thus, not all entry fields are visible.

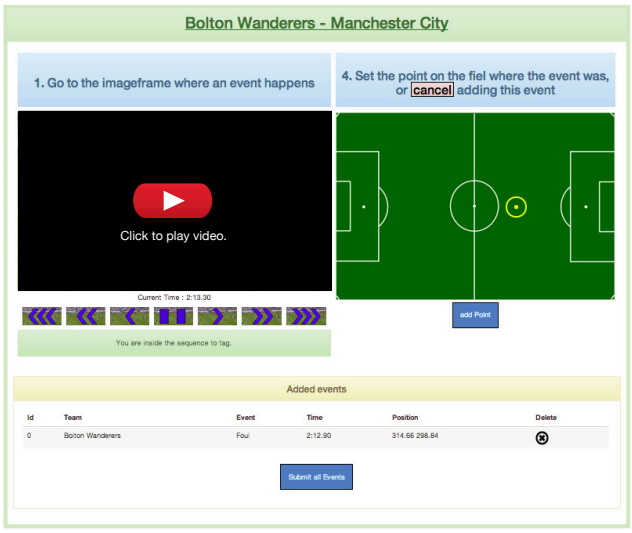

Figure 2: User interface of CrowdSport.

3.2 Data Quality

3.2.1 *User Rating*

In an attempt to increase the reliability of the data entered, we apply a rating system that assigns each user a confidence/trustworthiness score $uc \in \mathbb{R}$. The more negative the score, the more un-confident we are towards the user and her entries; the more positive the score, the higher our confidence.

This rating is first set on the basis of an initial assessment task and updated each time after the integration of the various entries and the calculation of the final events for a sequence (this will be described in Section 3.2.2). The confidence score serves two purposes: First, a confidence score that is too low (in our setting we use $uc \leq -1.5$ as threshold, i.e., accepting some untrustworthiness) leads to the banning of the user from all future tasks. Second, using the trustworthiness score, we give –when integrating and combining the contributions of multiple workers– more weight to trusted users and their entries [5].

When we assign an initial confidence score uc_{init} to a participant, the quality of the contribution of this user is assessed on the basis of an initial assessment task t_{init}. In this task, the user has to annotate a sequence that has already been labeled (similar to [9, 16]). The value of uc_{init} is calculated by the sum of the scores r_e for each single event e the user has entered to the initial task t_{init} as noted in Eq. 1.

The calculation of uc_{init} considers the entire set of events E_{init} specified by a user in the initial task. Each event $e \in E_{\text{init}}$ is then compared to the most similar event from the ground truth, denoted by argmax. Note that such assignment problem has in general a high asymptotic complexity and can only be solved in polynomial time. However, due to the short video sequences which are associated with a task, the actual number of events specified by a user is very small, and so the computational complexity of the assignment problem can be ignored in our case. Furthermore, the computation is performed at off-line time and does therefore not degrade the performance of the system.

If an event e entered by a user to the system cannot be matched to an event in the ground truth (and vice versa), we set the event rating $r_e = -0.5$. This means that with a threshold of $uc = -1.5$, a user can only enter three events that cannot be assigned to an event from the ground truth in the initial task (unless this is compensated by other, more positive ratings). Otherwise, she will be banned from the system and from entering new data.

$$uc_{\text{init}} = \sum_{e \in E_{\text{init}}} \text{argmax } r_e \tag{1}$$

In here, we rate each entry e, i.e., each event identified by a user, by the weighted sum of the rating we assign for the single elements asked from the worker. For an event e, these ratings are: $r_{\text{type}}(e)$ for the type of the event (e.g., shot on goal, interception, etc.), $r_{\text{team}}(e)$ for the team assigned to the event, $r_{\text{pos}}(e)$ for the position on the field associated with the event, and finally $r_{\text{time}}(e)$ for the timestamp on which the event takes place. Thus, r_e is defined as

$$r_e = \sum_{i \in ER} w_i r_i(e) \tag{2}$$

where ER denotes the four different event categories which are rated individually and which, in turn, correspond to the four questions asked to the user during a task. Moreover, w_i denotes the weight assigned to each element of ER, with $\sum_i w_i = 1$. In our implementation, we have equally weighted the ratings of type, position, team, and time.

In the following, we describe how the rating of each question is determined in CrowdSport. For questions in which the correctness of the answer can be assessed in a binary way (i.e., the associated team and the event type can be either correctly chosen or the answer is wrong), we define the rating as follows (in here, we exemplify this for event rating 'team' denoted as θ, but it is also true for 'type')

$$r_{\text{team}}(e) = \begin{cases} 1 & \text{if } \widetilde{\theta}(e) = \theta(e) \\ -1 & \text{if } \widetilde{\theta}(e) \neq \theta(e) \end{cases} \tag{3}$$

where $\widetilde{\theta}(e)$ denotes data entered by the user for event e, and $\theta(e)$ denotes data on e from the ground truth.

For the position rating $r_{\text{pos}}(e)$, we first calculate the Euclidean distance between the data entered ($\widetilde{\text{pos}} = (\tilde{x}, \tilde{y})$) by the worker and the data from the ground truth ($\text{pos} = (x, y)$) for an event e, i.e., $\delta(e) = \sqrt{(x - \widetilde{x})^2 + (y - \widetilde{y})^2}$.

To allow for small deviations δ_ϵ in the position data, we define the position rating $r_{\text{pos}}(e)$ as

$$r_{\text{pos}}(e) = 1 - \begin{cases} \frac{\delta(e)}{\delta_\epsilon} & \text{if } \delta(e) > \delta_\epsilon \\ (\frac{\delta(e)}{\delta_\epsilon})^2 & \text{if } \delta(e) \leq \delta_\epsilon \end{cases} \tag{4}$$

δ_ϵ is a threshold to allow for small deviations in the position. We set in our experiments $\delta_\epsilon = 2.5$m. With this, our rating is rather insensitive for distances $\delta(e) \leq \delta_\epsilon$, whereas for larger $\delta(e)$, the rating of e is adjusted more strongly.

For the difference in time (with the timestamp associated with event e being denoted by $\tau(e)$), we define $r_{\text{time}}(e)$ for an event e as follows:

$$r_{\text{time}}(e) = 1 - 2\,|\widetilde{\tau}(e) - \tau(e)| \tag{5}$$

which means that at a difference of 1 second to the ground truth (within the user interface this is 10 frames), the annotation is considered of low quality and is thus punished.

Using these definitions, the initial confidence score is calculated in the beginning on the basis of an assessment for which the ground truth exists ($uc = uc_{\text{init}}$). Moreover, the user confidence score is updated during the off-line phase when all tasks are completed and the data have been integrated (see Section 3.2.2). Using the newly calculated basis, uc is updated to uc^* using the rating from task t and in particular on the basis of the event set E_t that has been specified by the user for task t:

$$uc^* = uc + \sum_{e \in E_t} \text{argmax } r_e \tag{6}$$

3.2.2 Data Integration and Cleaning

To increase the quality of the data, a video sequence is labeled repeatedly by different users. Then, in an off-line processing phase, the data are integrated and cleaned with the goal to create a coherent data set. We use a sliding window approach when generating the video snippets so that we have each point in time labeled multiple times, but deliberately at different positions within the sequence to annotate.

For integrating and cleaning the data points entered, we consider each entry as a point in a four dimensional space. We use the DBSCAN [7] clustering algorithm to build clusters around the same event. DBSCAN has the advantage that it starts from an unknown distribution of the data and does not need to know the number of clusters present. Given a threshold ϵ, the algorithm adds all points that have a distance smaller than ϵ to the cluster. We have set the parameters so as to generate more false negatives (i.e., missing events), rather than false positives (i.e., wrong events).

Similar to [5], we use the score when combining the contributions of each worker to give more weight to trusted users. Given a cluster label for each point as resulting from the DBSCAN algorithm, the new position is calculated using the rating information at the moment of entering the data.

For the new values of the event categories, all events $e \in E_{\mathcal{C}}$ that are associated with the same cluster $\mathcal{C}$ are considered. The new position pos' for the integrated basis is calculated as

$$\text{pos}' = \frac{\sum_{e \in E_{\mathcal{C}}} \text{pos}(e)\ uc_e}{\sum_{e \in E_{\mathcal{C}}} uc_e} \tag{7}$$

where uc_e denotes the trustworthiness of the user that has contributed the specification of event e of $E_{\mathcal{C}}$. The time (time') is calculated in a similar way. The team (team') and the event types (type'), on the other hand, are determined by comparing the number of mentions and choosing the option that has most votes.

3.3 Implementation

We have implemented CrowdSport as a PHP and MySQL-based web application that uses Matlab scripts for the integration and cleaning step. The tasks have been published on the Microworkers platform in the category "search, click and engage" with an international target. We have defined an estimated completion time of 5 minutes per task, i.e., per video snippet of five seconds, and we have offered 0.20 US$ as payment for each completed task.

4. EVALUATION

4.1 Setting

For the evaluation, we have used an annotated sport video from a soccer match[2] of Manchester City (MC) vs. Bolton Wanderers (BW) from the English Premier League. We have chosen four scenes, which do neither contain any slow-motions nor replays, as workers only get to see a snippet and might not be able to distinguish between a replay and a real game scene. To generate the ground truth, we have manually and carefully re-annotated the four scenes, since the annotations provided in the data set have shown to be rather unprecise. We use one of the four scenes for the assessment step, the other three scenes are used in three campaigns that we have consecutively published after the previous campaign was completed (Test 1–3 in the following).

The assessment is a five second snippet that contains three events that the workers are expected to annotate. Users are in any case first redirected to the evaluation task within the Microworkers platform if the system detects that it has not yet been completed by the worker.

The test scenes are between 50 – 80 seconds in length. In the three tests, the scenes are split into 5 second snippets in a sliding window manner with a displacement of 0.5 seconds. For each snippet, we generate a task that we assign to a worker. Therefore, for each moment in time, we have data by ten different users (however, due to the displacement, for each user a particular point in time is at a different temporal position within the snippet). We then use the collected data for the integration and cleaning step.

The annotation campaigns were all completed within approximately 30 hours. On the demographics, using analysis data from Google Analytics, we can report that the largest group of participants, i.e., 71% of the users, were from Asia, with most users coming from Bangladesh, Nepal, and India.

4.2 Results

In the following, we report on the evaluation results from the assessment step and the three tests. Table 1 summarizes the results. It shows the acceptance rate denoting the percentage of the tasks that were accepted and received payment ($uc > -1.0$), and the banned rate, i.e., the percentage of users that were excluded after completing this task due to a rating that was too low ($uc \leq -1.5$). In the other cases, users neither get paid (and neither are their entries to the task used in the integration step), nor will they be excluded from further tasks.

The results in Table 1 show that our banning rate is rather high in the assessment step, considering that 47% of the workers, i.e., 332 workers in total, were not allowed to enter data to the true tests but were excluded after the assessment step already. A more in-depth analysis of the evaluation shows that about 40% of the events added by the workers in the assessment step were just random entries and did not correspond to any true event in the video sequence. Thus, the assessment step can be considered as a filter that excludes all users that have seemingly not been able to enter reasonable data. In particular, the assessment step banned all users that did not enter any data, i.e., missed all the three events that were supposed to be detected in the assessment.

[2]The annotations were available from the Manchester City Analytics Team, `http://www.mcfc.co.uk/the-club/mcfc-analytics`.

Table 1: Summary of evaluation results.

Description	Length	Workers	Accepted	Banned
Assessment	5s	709	53%	47%
Test 1	50.5s	101	83%	12%
Test 2	75s	150	79%	8%
Test 3	80s	160	86%	9%

Although the results in Table 1 could imply that, thanks to the assessment step, the wheat was separated from the chaff and, thus, the actual tests have very low banned rates (or, to put it differently, that most contributions have been accepted). However, although this is to some extent the case, one should note that in the tests the update of *uc* is based on the integrated and cleaned data entered by multiple users and not on a fully correct ground truth. Thus, the data of a single user will probably not heavily differ from the mean of the data of a cluster, since it was used as a basis to calculate the mean. This leads to a much more conservative value for the banning rate.

Consider, for example, the comparison of the data from the ground truth with the data calculated by our application in the integration step for Test 1 in Table 2. In particular, the table shows position, time, event, and team information that is available in the ground truth compared to what is computed using the DBSCAN algorithm – and also lists the difference between these two measures.

We have highlighted with a gray background the events that could not be matched to each other, i.e., in one case a single event that was calculated by the DBSCAN algorithm, but which is not available in the ground truth; in another case, there are seven events in the ground truth, but these were not calculated as events in CrowdSport. Note in this respect especially the events at timestamps 33:56, 34:22, and 34:41 for which two different events are present at the same time in the ground truth, denoting in all three cases an interception with a subsequent pass. In these cases, the DBSCAN algorithm was not able to create two distinct events with the data entered from the workers, but merged the event pairs, since the time and positional information is identical. Similarly, the DBSCAN algorithm merged the give-and-go (two consecutive passes) that happened at 34:08 and 34:09.

On the events that were detected by CrowdSport and that can be matched to an event from the ground truth, the table shows that the position is calculated with a precision that subsumes a difference between 0.78m up to 6.84m, i.e., about 7% of the whole field. On the other hand, the time of the event happening differed to the ground truth by only 0.3s at maximum.

Of course, the computation of the data in Table 2 heavily depends on parameters set for the DBSCAN algorithm. It should be subject to future work to come up with good parameter settings to decrease the number of both false positives and false negatives.

4.3 Lessons Learned

The results from the evaluation have shown that it is crucial to add assessment questions to the tasks to solve for the worker. The high rate of workers banned in the assessment step rises speculations about the quality of the results if all users were allowed to enter data to the actual tests. This is also suggested by [8], as they propose that the quality could be improved by hiding assessment questions within the task and pose those questions in varying intervals to the worker.

Nevertheless, from a deeper analysis of the entered data, we have realized that the bad quality of data entries was not only due to malicious workers, but in some cases also because of ambiguous and complex situations in the video scenes: For example, consider a scene in which a player intercepts the ball and passes it at the same time (or very shortly after) to another player of his team. In this case, the majority of the users entered only the interception, a minority entered only the pass, and only very few have correctly identified both events. Furthermore, exactly determining the position of an event has also posed problems, especially in close-up shots where no line of the field was visible in the video and workers did not have any visible aid to precisely locate the players on the field [11]. This has also been true for events such as goals, where it was not clear to the worker where to locate the ball, i.e., on the line, within the goal, etc.

5. CONCLUSION AND OUTLOOK

In this paper, we have presented the CrowdSport approach to crowd-based sports video annotation. Rather than applying complex event detection algorithms, we leverage the wisdom of the crowd to identify semantic events in soccer videos and to annotate the video with spatio-temporal information on these events. For the purpose of ensuring a high quality of the annotated data, we have applied a rating system that has shown in the evaluation to be very helpful to largely increase the quality of the data. The evaluation has shown that thanks to the various mechanisms we built into CrowdSport useful results in scenes that are not overly complex, e.g., that do not involve too many events at once, can be achieved with the objective that nearly 80% of the entries to the true tests could be accepted as good enough to the end of being used in sport video retrieval applications [1].

In our future work, we will further improve on the offline processing of the data, i.e., the data integration and cleaning step, to the end of improving the resulting data set. Furthermore, we will also experiment with videos from other team sports.

Acknowledgments

This work was partly supported by the Swiss National Science Foundation in the context of the Chist-Era project IMOTION (Intelligent Multi-Modal Augmented Video Motion Retrieval System), contract no. 20CH21_151571, and a Microworkers.com grant.

6. REFERENCES

[1] I. Al Kabary and H. Schuldt. Sportsense: Using motion queries to find scenes in sports videos. In *Proc. Int. Conf. on Information and Knowledge Management (CIKM 2013)*, pages 2489–2492, San Francisco, USA, 2013. ACM.

[2] I. Al Kabary and H. Schuldt. Enhancing sketch-based sport video retrieval by suggesting relevant motion paths. In *Proc. Int. Conf. on Research and Development in Information Retrieval (SIGIR 2014)*, pages 1227–1230, Gold Coast, Australia, 2014. ACM.

[3] J. Assfalg, M. Bertini, A. D. Bimbo, W. Nunziati, and P. Pala. Soccer highlights detection and recognition

Table 2: Results of integration in Test 1.

Ground truth					CrowdSport					Δ			
x	y	time	type	team	x	y	time	type	team	distance	time	type	team
293.6	274.4	33:54	pass	BW	287.8	263.0	33:54	pass	BW	1.96m	0.1s	-	-
457.1	347.5	33:56	intercept	MC	471.4	316.5	33:56	intercept	MC	5.25m	0.1s	-	-
457.1	347.5	33:56	pass	MC	*not detected by CrowdSport*								
427.4	275.6	33:58	pass	MC	435.3	260.3	33:58	pass	MC	2.64m	0.1s	-	-
491.8	209.5	34:02	pass	MC	497.0	207.4	34:02	pass	MC	0.88m	0.1s	-	-
not in ground truth					458.1	211.5	34:03	pass	MC				
437.3	262.8	34:04	pass	MC	442.5	248.4	34:04	pass	MC	2.35m	0.3s	-	-
408.8	319.8	34:06	pass	MC	415.5	317.8	34:06	pass	MC	1.11m	0.2s	-	-
398.9	417.3	34:08	pass	MC	419.7	359.6	34:08	pass	MC	9.41m	0.1s	-	-
284.9	319.8	34:09	pass	MC	*not detected by CrowdSport*								
131.3	300.8	34:18	pass	BW	127.5	285.8	34:18	pass	BW	2.37m	0.1s	-	-
462.1	224.0	34:22	intercept	MC	477.2	184.2	34:22	intercept	MC	6.53m	0.2s	-	-
462.1	224.0	34:22	pass	MC	*not detected by CrowdSport*								
209.4	267.7	34:37	intercept	BW	182.4	256.6	34:37	intercept	BW	4.60m	0.1s	-	-
204.4	259.5	34:37	pass	BW	212.8	241.8	34:38	pass	BW	3.01m	0.2s	-	-
234.1	235.9	34:39	pass	BW	*not detected by CrowdSport*								
249.0	279.3	34:40	pass	BW	257.2	235.3	34:40	pass	BW	6.86m	0.1s	-	-
298.6	246.3	34:41	intercept	MC	*not detected by CrowdSport*								
298.6	246.3	34:41	pass	MC									
239.1	248.7	34:43	pass	MC									

using HMMs. In *Proc. Int. Conf. on Multimedia and Expo (ICME 2002)*, pages 825–828, Lausanne, Switzerland, 2002. IEEE.

[4] M. Beetz, B. Kirchlechner, and M. Lames. Computerized Real-Time Analysis of Football Games. *IEEE Pervasive Computing*, 4(3):33–39, 2005.

[5] A. Doan and R. McCann. Building Data Integration Systems: A Mass Collaboration Approach. In *Proc. IJCAI W. on Information Integration on the Web (IIWeb 2003)*, pages 183–188, Acapulco, Mexico, 2003.

[6] A. Ekin, A. M. Tekalp, and R. Mehrotra. Automatic Soccer Video Analysis and Summarization. *IEEE Trans. Image Processing*, 12(7):796–807, July 2003.

[7] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proc. Int. Conf. on Knowledge Discovery and Data Mining (KDD 1996)*, pages 226–231, Portland, USA, 1996. AAAI.

[8] Q. Liu, A. T. Ihler, and M. Steyvers. Scoring Workers in Crowdsourcing: How Many Control Questions are Enough? In *Proc. Conf. on Neural Information Processing Systems (NIPS 2013)*, pages 1914–1922, Lake Tahoe, USA, 2013.

[9] R. McCann, W. Shen, and A. Doan. Matching Schemas in Online Communities: A Web 2.0 Approach. In *Proc. Int. Conf. on Data Engineering (ICDE 2008)*, pages 110–119, Cancún, México, 2008. IEEE.

[10] S. Park, G. Mohammadi, R. Artstein, and L.-P. Morency. Crowdsourcing micro-level multimedia annotations: The challenges of evaluation and interface. In *Proc. Multimedia 2012 W. on Crowdsourcing for Multimedia (CrowdMM 2012)*, pages 29–34, New York, USA, 2012. ACM.

[11] C. Perin, R. Vuillemot, and J.-D. Fekete. Real-Time Crowdsourcing of Detailed Soccer Data. In *What's the score? The 1st W. on Sports Data Visualization (VIS 2013)*, Atlanta, USA, 2013. IEEE.

[12] K. Sato and J. K. Aggarwal. Tracking Soccer Players Using Broadcast TV Images. In *Proc. Int. Conf. on Video and Signal Based Surveillance: Advanced Video and Signal Based Surveillance (AVSS 2005)*, pages 546–551, Como, Italy, 2005. IEEE.

[13] I. Spiro, G. Taylor, G. Williams, and C. Bregler. Hands by hand: Crowd-sourced motion tracking for gesture annotation. In *Proc. Conf. on Computer Vision and Pattern Recognition W. (CVPRW 2010)*, pages 17–24, San Francisco, USA, June 2010. IEEE.

[14] A. Tang and S. Boring. #EpicPlay: crowd-sourcing sports video highlights. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI 2012)*, pages 1569–1572, Austin, USA, 2012. ACM.

[15] L. von Ahn and L. Dabbish. Labeling Images with a Computer Game. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI 2004)*, pages 319–326, Vienna, Austria, 2004. ACM.

[16] C. Vondrick, D. Patterson, and D. Ramanan. Efficiently Scaling up Crowdsourced Video Annotation. *Int. J. of Computer Vision*, 101(1):184–204, Jan. 2013.

[17] C. Vondrick, D. Ramanan, and D. Patterson. Efficiently Scaling Up Video Annotation with Crowdsourced Marketplaces. In *Proc. Europ. Conf. on Computer Vision (ECCV 2010)*, pages 610–623, Heraklion, Crete, 2010. Springer.

[18] J. Yuen, B. C. Russell, C. Liu, and A. Torralba. Labelme video: Building a video database with human annotations. In *Proc. Int. Conf. on Computer Vision (ICCV 2009)*, pages 1451–1458, Kyoto, Japan, 2009. IEEE.