EXCITE: EXploring Collaborative Interaction in Tracked Environments

Nicolai Marquardt[1], Frederico Schardong[2], Anthony Tang[2]

[1]University College London, Gower Street, London, UK
nicolai.marquardt@acm.org
[2]University of Calgary, 2500 University Drive NW, Calgary, Canada
frede.sch@gmail.com, tonyt@ucalgary.ca

Abstract. A central issue in designing collaborative multi-surface environments is evaluating the interaction techniques, tools, and applications that we design. We often analyse data from studies using inductive video analysis, but the volume of data makes this a time-consuming process. We designed EXCITE, which gives analysts the ability to analyse studies by quickly querying aspects of people's interactions with applications and devices around them using a declarative programmatic syntax. These queries provide simple, immediate visual access to matching incidents in the interaction stream, video data, and motion-capture data. The query language filters the volume of data that needs to be reviewed based on criteria such as application events, and proxemics events, such as distance or orientation between people and devices. This general approach allows analysts to provisionally develop theories about the use of multi-surface environments, and to evaluate them rapidly through video-based evidence.

Keywords: Interaction analysis, collaborative interaction, tracked environments

1 Introduction

One important concern in designing and building multi-surface environments is ensuring that the tools and interaction techniques meet the collaboration needs of people in the environment. Researchers conduct studies of collaborative activity to understand the effect of interaction techniques and applications [9], and one of the main challenges is analysing these studies. Typically, there are many interactions in a study of collaborative behaviour in a multi-surface environment: collaborators are working with one another, or alone; they may be making use of tablets, tables, or large displays; they may be studying something on a tablet before looking up or exploring data on a different display. Yet, to determine which factors are affecting what behaviours is time-consuming. Proper analysis of this data involves time-consuming transcription and annotation of video data recorded of these studies to understand the interactions between the moving entities (e.g. [4]).

To support this analytic task, we designed EXCITE, a tool that can ease the burden of video and tracking data analysis for multi-surface environments (Fig. 1). EXCITE

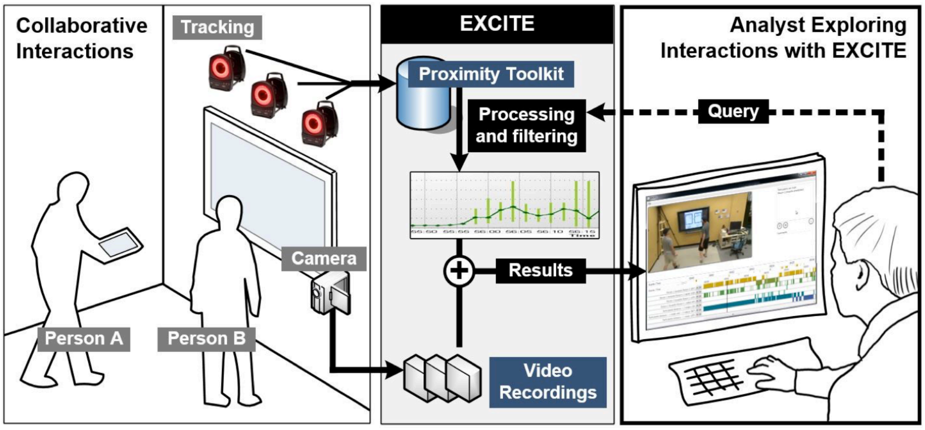

Fig. 1. EXCITE overview: Capturing video, tracking data, and application events (center) in collaborative environments (left), and providing querying interface and visualizations to allow analysis of group interactions (right).

leverages proxemic information—such as people's and devices' distance and orientation, captured with the Proximity Toolkit [5]—simultaneously with video data and event data from the tools, and allows video analysts to generate queries on the data as a whole. These queries are annotated on a timeline, allowing the analyst to scrub through synchronised video capture feeds to validate and further annotate study sessions. Using our tool allows the analyst to compare the incidence of various events with one another, and see these in captured video of behaviour from study sessions.

Figure 1 illustrates the core features of EXCITE. It captures and synchronizes spatial tracking data (e.g., people's and devices position, orientation, and movements) with video streams and supports interactive queries that filter these data streams. As illustrated in Figure 2, these queries result in an annotated timeline of study sessions, where EXCITE provides a visual interface with event-scrubbers allowing an analyst to skip between occurrences of events, and even compound constructions of events. The queries themselves are constructed through a mix of declarative/imperative semantics. Because these can be constructed and visualised quickly, an analyst can rapidly explore and iteratively test different hypotheses. This kind of approach supports the style of inductive analysis that is used in video analysis [4,6,8,9].

2 Related Work

We review tool support for facilitating qualitative analysis of interactions, in particular (1) studying interactions with video analysis, (2) evaluating prototype hardware, and (3) investigating interactions in multi-surface environments.

To facilitate the laborious task of analysing video data, tools have been developed for easier video review, motion and frame-by-frame analysis, or adding of annotations [2]. Recent systems began augmenting video data with visualizations of captured sensor information to facilitate the analysis of interactions. For example, VACA provides a simultaneous review of video and sensor data [1]. It uses a synchronised timeline and side-by-side playback of video and additional captured sensor data. The person analysing the video can then use the provided sensor data as additional cues for finding relevant parts of the recorded interactions in the video stream.

Further specialised tools have been designed for investigating interactions with novel hardware prototypes. D.tools introduced a statechart-based analysis tool linking videos of test sessions, interaction states and other events in combined visualizations and a common event timeline [3]. ChronoViz uses multiple streams of time-based data for studying paper-based digital pen annotations [10]. Our design of EXCITE is inspired by this work, translating a similar visual inspection tool of video+sensor data to ubiquitous computing (ubicomp) interactions.

More recently, tools facilitating the analysis of multi-person and/or multi-device interactions have emerged. With pure capture of people's actions with multi-device software, VICPAM provides timeline visualizations of groupware interactions [7]. VisTACO [8] records interaction sequences of multiple remote-located people with digital tabletops, providing touch trace visualizations, and allowing insights into the spatial and temporal distribution of tabletop interaction [9]. Panoramic is an evaluation tool for ubicomp applications, providing historical sensor data visualizations [11]. Finally, the Proximity Toolkit allows recording and playback of tracked interaction sequences of multi-entity relationships (multiple people and devices) [5].

3 Design Rationale

In general, the prior work provides great insight into the challenges of analysing collaborative interaction, and fit closely with our own experiences with this task. The challenges of studying collaborative behaviour with technology are well documented—briefly, that in contrast to studying one user interacting with a computer (where there is only one relationship to observe), the multi-device, multi-person nature of multi-surface collaboration means there are many more relationships to be studying, observing, and to be made sense of. Researchers have generally found this challenging to do using traditional field notes, and so relying on video to enable replaying and reviewing of the interactions is common. Nevertheless, the task remains time-consuming: the common method of using an inductive, qualitative analytic approach means that a researcher needs to be able to generate hypotheses, and explore this to see whether it seems to be happening across the captured video [4].

As we have seen, sensor data can help, but only to a limited extent—sensor data provides cues into the data, but at very low levels, this sensor data can be misleading, cuing us to the wrong events in the video stream entirely (e.g. [8]). For instance, just because two study participants in a collaborative task are nearby one another, does not mean they are working with one another. But, if we also see application events at this time that indicate cross-device information transfer is happening, then we can be more certain they are working together. The main analytic challenge—making sense of what is happening (or what happened)—remains.

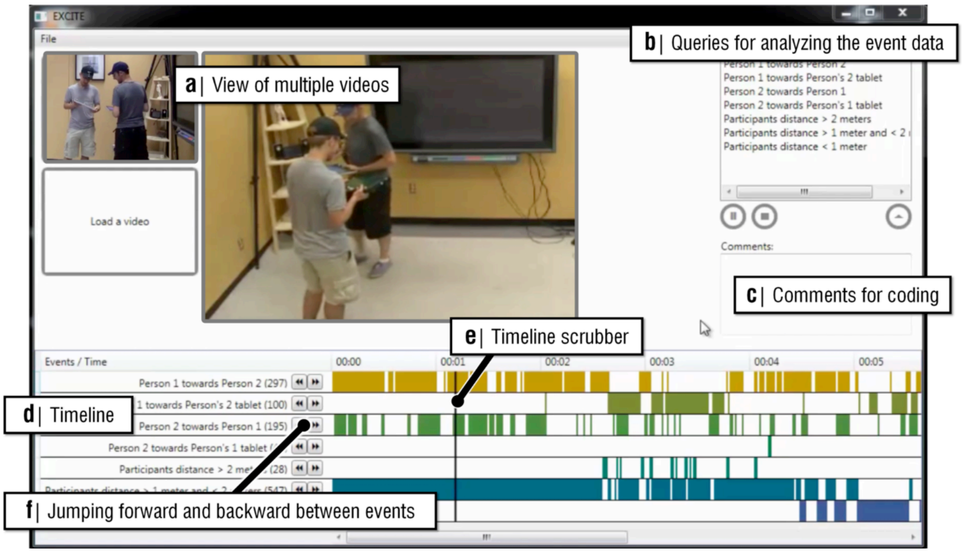

Fig. 2. User interface of EXCITE (see text for descriptions of interface elements).

The rationale for designing EXCITE begins from this argument. To support analysis of multi-person interactions in multi-device environments, we saw that we needed not only to integrate multiple streams of data (i.e. video data, sensor data, and also application data), but also to provide means for analysts to provisionally query the data. These queries, generated by the analyst, and executed by the system, would then allow the analyst to focus efforts on the higher-level analytic task (e.g. is collaboration happening), rather than on the low-level task of simply finding video evidence to support or refute hypotheses. Thus, beyond functioning just as a filtering mechanism, the queries allow the analyst to engage in true exploration of the data on analytic, rather than on sensor-value terms.

4 EXCITE: Overview and Design

We contribute EXCITE, a tool that allows rapid review of captured tracking data of interaction sessions in multi-surface environments. EXCITE facilitates the analysis of group interaction tasks in ubicomp environments, by providing an expressive querying interface and appropriate event visualizations. It unifies the access to multiple data sources (including up to four video streams and spatial tracking data).

The user interface of EXCITE consists of three major elements. First, the views of one or multiple recorded video streams of the interaction sequence (Figure 2a). A user can add up to four video files to the viewer that can be rearranged and resized. Second, Figure 2b shows the list of all currently entered queries for analysing the interaction. Figure 2c also shows a text box for adding comments during video coding. Finally, Figure 2d illustrates the navigation timeline, which includes a list of all queries, including a visualization along the x-axis of the timeline indicating when the conditions of the query are met. The timeline includes a temporal navigation scrubber (2e), and allows scrubbing the timeline to navigate forward or backwards. Alternatively, the analyst can use the navigation buttons (2f) to jump forward and backward between events.

Internally EXCITE connects to the Proximity Toolkit [5] for capturing the tracking information of people's and devices position and orientation. The Proximity Toolkit uses depth-sensing cameras to track people's location, and a high end motion capturing system with infrared-reflective markers to track devices. EXCITE records the proxemic information provided by the toolkit—that is, distance, orientation, identities, movement, and location—and stores this data in an internal data structure to perform the queries.

5 Query Language

The query language is the core toolset provided by EXCITE for analysing the recorded video and sensor data of the performed interaction. The queries allow filtering and analysing the captured data for quickly finding particular events that happened during interaction.

5.1 Structure and Composition of Queries

Each query is composed out of (1) one or two presence/application-event identifiers, (2) a function to compare or a property to check, and (3) a condition. An analyst can add as many individual or compound queries for the event stream as they need; any new query is added as a new horizontal parallel event stream in the timeline view (Figure 2d and 3).

The *presence identifiers* directly correspond to the identification names of entities tracked in the Proximity Toolkit. These can be identifiers for people such as 'Person1' or their name such as 'Taylor', or for devices such as 'Smartboard' (for the large interactive surface) and 'Tablet' (for an interactive tablet computer). EXCITE also handles *application event identifiers*, where the system can read log files generated by applications (e.g. Tablet1.TouchDown).

The next components are the *functions*. They can compare values between two entities or check a property of a single entity. The available functions (with Boolean or Integer return values) for comparing two presence identifiers are:

- **Distance (Integer):** distance between entities (in mm).
- **Velocity_difference (Integer):** the velocity difference between two tracked entities.
- **Orientation_difference (Integer):** the difference in orientation angles of the default pointer in the proximity toolkit (in degrees).
- **Towards (Boolean):** true if an entity is pointing towards another entity (orientation angles divided in two sections at +90 to -90).
- **Pointing_at (Boolean):** true if an entity's previously defined pointing vector (e.g., the normal vector of a screen) is directly pointing at another entity.

- **Touching/Colliding (Boolean):** true if two entities are either touching or their bounding volumes are colliding (below a set fixed threshold).
- **Parallel/Perpendicular (Boolean):** true if default pointing vectors are parallel or perpendicular.

Each function can be combined with the names of *two tracked entities* and a *condition* to compare to. Valid operators for the comparison are: <=, <, >, >=, ==, and !=. For example, the following query checks if the distance between two people is smaller than 1m (1000mm):

```
person1.distance(person2) < 1000
```

As another example, a query can check if a person is facing towards the large display (i.e., the smartboard) or facing away (using the *towards* function):

```
person1.towards(smartboard)
```

Individual properties of an entity include the entity's 3D coordinates (X, Y, Z values) as well as orientation and velocity values. Again, the properties can be combined with a condition to filter only the events of interest.

5.2 Compound Queries and Parallel Event Streams

Combining multiple conditions into compound queries gives analysts a powerful tool for refining the hypothesis investigate with EXCITE. Multiple statements can be combined with logical operators (&& for *and*, || for *or*, ! for *not*), and compound queries can be composed of any number queries and logical operators. The following example's query only returns results if both concatenated individual queries are valid: first, if the distance between two people is smaller than 1m, and second, if 'person1' is facing towards the large display.

```
person1.distance(person2) < 1000 && person1.towards(smartboard)
```

Because the queries can be constructed and visualised quickly, the analyst can rapidly generate, explore and refine different hypotheses. This kind of approach supports the inductive analytic methods used by video analysts.

6 Analysis Walkthrough Case Study

We now show a case study example to demonstrate how an analyst can apply EXCITE in practice when analysing group interactions with interactive surfaces.

Larry has designed a simple multi-surface system that allows people to share information to others via two interaction techniques (inspired by [6]): (1) *portals*—if tablets are held close to one another, information can be swiped from one display to the next, and (2) *hold-to-mirror*—if the tablet is turned to face toward the large display, then information on the tablet is transferred onto the large display. Larry is interested in how people share information in his study.

After running his study, Larry loads the logged data streams from his application, which populates on the timeline as two separate tracks: one for "portal" transfers, and one for "hold-to-mirror" transfers (Fig. 3a). As Larry goes through the synchronised

Fig. 3. Analysis generated through queries to the EXCITE tool.

video data, he notices that the "hold-to-mirror" events match up with times when participants are not close together. He generates the following query:

```
person1.distance(person2) > 4000
→ When are the people far away from each other (more than 4m)?
```

This does not seem to return many query results—participants stayed within 4m of each other most of the time. He *refines* the query with a smaller window:

```
person1.distance(person2) > 3000
→ When are the people further than 3m from one another?
```

Based on the new track (Fig. 3b), he is able to see that indeed, every one of the "hold-to-mirror" transfers happens when participants are not standing too close to one another (Fig. 3b). Out of curiosity, Larry refines his query again to see whether close distances correspond with "portal" transfers:

```
tablet1.orientation_difference(tablet2) < 15 && person1.distance(person2) < 1000
→ When are tablets oriented in the same direction, AND people are close together?
```

The results of this compound query do correspond with most of the "portal" transfers, yet the window in which this is happening seems quite big (Fig. 3c). As Larry goes each incident by inspecting the video, Larry realises that participants are actually sharing information by showing each other their tablets rather than strictly using the "portal" tool! He constructs a final query to capture this (Fig. 3d):

```
person1.distance(tablet2)< 500 || person2.distance(tablet1)< 500
→ When are people close to the other person's tablet?
```

While the result of this query is not perfect, Larry can use the results to cue him to parts of the video where one participant might be sharing information with another by simply showing the tablet to another person. Without EXCITE, Larry would be left to review the video data, perhaps using a manual annotation tool, and not arrive at his final theory until much later in the process.

7 Conclusion

Opportunities and Limitations. While EXCITE is a flexible tool facilitating analysis of collaborative interactions, it is a proof-of-concept system. The current lexical power is limited to the query language outlined in Section 5, though there is ample opportunity to extend this: for instance, by allowing analysts to script altogether new semantics (e.g. walking, turn-around). These functions, which might take into account proxemic variables over time, could then be used for the filtering of data. Similarly, the UI can be improved to allow for dynamic filtering—for instance, rather than entering a specific value for a velocity or orientation, one would be able to manipulate a slider. The current implementation also depends on the Proximity Toolkit, which mainly functions with the high-end VICON and OptiTrack motion capturing systems. While in principle the Proximity Toolkit can be extended to other position/orientation capture systems, this dependency limits the current application of EXCITE.

Summary and conclusion. We contributed the design of a novel tool for analysing interaction sequences in multi-person and multi-device environments. The EXCITE tool and its query language facilitate the rapid inspecting of time-synced video and captured motion-tracking data—and support finding answers to enquiries about participants' use of ubicomp gestural interactions with tablets, walls, and tabletops. In our own work, we have begun actively exploring the use of EXCITE to support our analysis, and are looking for ways of both improving the power of the query language, as well as simplifying the syntax. We are interested in also exploring how spatial semantics (e.g. [8]) can be integrated into textual queries of the data. We demonstrated the potential of these kinds of analysis with our walkthrough, and believe that the design of EXCITE will be valuable for future studies investigating people's interactions in ubicomp ecologies (EXCITE is available for download at: http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Toolkits/EXCITE).

8 References

1. Burr, B. VACA: A Tool for Qualitative Video Analysis. *CHI '06 Ext. Abs.,* ACM (2006).
2. Dasiopoulou, S., et al. A survey of semantic image and video annotation tools. Knowledge-driven Multim. Information Extraction and Ontology Evol., Springer, 2011.
3. Hartmann, B., Klemmer, S., et al. Reflective Physical Prototyping Through Integrated Design, Test, and Analysis. *Proc. of UIST*, ACM (2006).
4. Jordan, B. and Henderson, A. Interaction Analysis: Foundations and Practice. *Journal of the Learning Sciences 4*, 1 (1995), 39–103.
5. Marquardt, N., Diaz-Marino, R., et al. The Proximity Toolkit: Prototyping Proxemic Interactions in Ubiquitous Computing Ecologies. *Proc. of UIST '11,* ACM (2011).
6. Marquardt, N., Hinckley, K., and Greenberg, S. Cross-Device Interaction via Micro-mobility and F-formations. *Proc. of UIST '12*, ACM (2012), 13–22.
7. Moghaddam, R. and Bailey, B. VICPAM: A Visualization Tool for Examining Interaction Data in Multiple Display Environments. *Proc. of HIMI,* Springer, 2011, 278–287.
8. Tang, A., Pahud, M., Carpendale, S., and Buxton, B. VisTACO: Visualizing Tabletop Collaboration. *Proc. of ITS '10*, ACM (2010), 29–38.
9. Tang, A., et al. Three's Company: Understanding Communication Channels in Three-way Distributed Collaboration. *Proc. of CSCW*, ACM (2010), 271–280.
10. Weibel, N., Fouse, A., Hutchins, E., and Hollan, J.D. Supporting an integrated paper-digital workflow for observational research. *IUI*, ACM (2011).
11. Welbourne, E., et al. Specification and Verification of Complex Location Events with Panoramic. *Pervasive Computing*. Springer, 2010, 57–75.