Multi-User Participation on Large-Screens – The Example of Collaborative Voting

Thomas Herrmann, Alexander Nolte, Marc Turnwald

Informations– und Technikmanagement, Ruhr-Universität Bochum

Abstract

The features of a prototype are presented to demonstrate how multi-user touch-interaction on a single interactive large screen can be practically employed for complex design tasks. The prototype refers to the field of collaborative modeling and supports the decision phase where relevant elements of a process have to be identified. Developing and using the prototype leads to a list of recommendations and proposals for improvement which serve as heuristics for the development of similar solutions where large interactive surfaces can be collaboratively used.

1 Introduction

Although large touchable displays became relevant for HCI-research during the last decade, it can be observed that introducing them for realistic applications does not advance at the same speed, as technical progress does. During recent years, development clearly aimed at detecting as many touches and gestures as possible on single surfaces. This trend is motivated by the idea of a more natural interaction for single users with virtual objects, like e.g. sizing or rotating photos. Additionally allowing multiple users to interact with onscreen-objects at the same time can be considered another trend (Morris et al. 2006).

Meanwhile the question arose for which kind of real-world scenarios, these new technologies and interfaces might become useful (Seow et al. 2009). While the aspects of personal information management, communication and entertainment are well examined for and occupied by small touchable displays like e.g. the Apple iPhone™ and iPad™, larger touchable displays like e.g. Smart Technologies SmartBoard™ or Microsoft Surface™ table, are oftentimes only used in information sharing and education. They have very little relevance in most people's everyday life, especially in people's business or work-life. According to our observations this is caused by a lack of appropriate applications that feature seamless integration of large touchable interfaces into existing workflows.

Based on our experiences with the collaborative development of graphical models in several cases (Herrmann 2009), the facilitation of touchable interfaces appears to be reasonable as it enables participants to directly interact with displayed materials, providing positive impact on the modeling process and thus the model itself (Rouwette et al. 2000). Especially when complex models are being developed, process owners, stakeholders and domain experts have to be integrated into the modeling process. This approach known as collaborative modeling (Renger et al. 2008) has been widely discussed in literature.

Furthermore a typical modeling application has many key features that make it affine for usage through touch interfaces. A modeling tool usually facilitates a modeling notation based on graphical representations of symbolic elements rather than complex textual representations. These elements are usually simple two-dimensional geometric objects like rectangles or circles, labeled with short textual descriptions. Complex situations can be described by adding elements of different types and captions as well as through bringing them in a spatial order on the screen or describing their relations to other model elements by lines or arrows. The creation and modification of graphical models like SeeMe (Herrmann 2006) contain two key affinities to touch screen interaction. First graphical objects can easily be manipulated by touch. Second graphical modeling limits the necessity of text input to rather short descriptions. Therefore switching to secondary input devices like keyboards or mice (Benko et al. 2009), or using virtual keyboards can be incorporated to a certain extent thus not disturbing the creative flow (Csikszentmihalyi 1997).

Additionally to modeling being an appropriate field of application even for single user touch interfaces, the size of the display comes into account, when models are developed collaboratively in collocated settings. During collocated collaborative modeling sessions, direct communication between the participants is very important. Furthermore it is feasible to provide manipulation techniques that can be used by every participant in parallel to ensure equal involvement. Having a look on earlier work (see e.g. Guimbretiére et al. 2001) reasonable tasks are presented in the area of handling multiple ideas for design but not under the condition of simultaneous multiple user collaboration.

In the remainder of this paper we will describe experiences with collocated modeling (section 2) and show why collaborative voting by touch can be used as a primal approach towards multi-user participation. Furthermore we will discuss its advantages and disadvantages compared to other technical solutions (section 3). In section 4 we will describe requirements for the implementation of a prototype. Afterwards we will show the interface that was developed and provide an analysis (section 5). Finally we will describe requirements for the development of interfaces for other activities in the context of collaborative modeling (section 6).

2 Previous Experiences with Collaborative Modeling on Interactive Large-Screens

Since 2007 we conducted several workshops (e.g. Herrmann & Nolte 2010) in a special special facilitation collaboratory called ModLab. Its centerpiece is a large, high-resolution interactive screen (4.80m x 1.20m; 4320x1050 pixels) consisting of three seamlessly integrated rear projection boards (see Figure 1). Touches are recognized through six cameras with overlapping angles ensuring seamless interaction over the whole width of the wall. For further interaction between audience and the displayed materials we provide a wifi network for capable devices like laptops or smart phones.

Figure 1:The ModLab – University of Bochum

During these workshops we use a self-developed graphical modeling tool supporting the development of socio-technical processes according to the modeling notation SeeMe[1]. The workshops are conducted according to the method of the socio-technical walkthrough (Herrmann 2009). During these workshops a moderator manages the communication while an additional modeler chauffeurs the modeling tool. We usually start these workshops by asking the participants which activities are conducted at a certain time, who conducts them and what is needed in order to finish them successfully. The moderator picks up these contributions and asks the modeler to enter them into the model, leaving all modifications on the displayed material solely to the modeler. The availability of touch interaction however enables the moderator to take over operations such as moving elements around or creating new elements, changing the modeler´s role to a typist for longer captions. Only complex operations especially those that require additional keys are still executed by the modeler.

[1] Please visit http://www.seeme-imtm.de for more information on the SeeMe Editor

Nevertheless having to rely on a moderator / modeler team remains a bottleneck of the modeling process, especially during early phases, where lots of new elements have to be created and labeled in a short amount of time. We also observed that the audience got bored more quickly, when the moderator interacted intensely with the large-screen, decreasing the audience´s immersion in the development process. We even observed participants standing up and joining the moderator at the large-screen to directly interact with the displayed materials. Similar observations are described by Hawkey et al. (Hawkey et al. 2005). So we started exploring new ways to involve participants more directly.

3 Collaborative Voting as a Primal Approach

Collaborative modeling is a key activity in system conceptualization and process design and therefore a particular field of interest for us. Renger et al. (Renger et al. 2008) developed a framework for collaborative modeling showing that group composition, interactive process and the underlying modeling method provide key influences to the overall model quality. In the course of this paper, we focus on improving the interactive process through the integration of multi-user cooperation on an interactive large screen.

The collaborative process can roughly be divided into divergent and convergent tasks. Divergent tasks cover aspects such as the generation of new ideas through e.g. brainstorming while convergent tasks focus on consensus finding, decision making or the preparation of such. Andersen and Richardson (Andersen & Richardson 1997) developed scripts for a set of key activities like brainstorming and voting. During several workshops where processes had to be newly developed it became obvious that our previous approach – the socio-technical walkthrough (STWT) – proved to be problematic due to its forced sequentiality. Therefore we integrated electronic brainstorming into our modeling tool in order to support parallel contributions by multiple participants (Herrmann & Nolte 2010). During brainstorming however usually a lot of contributions are produced, leaving the problem of coping with them. So we integrated a voting phase afterwards, in order to limit the options and to assess a direction for further process development.

According to our experiences voting however isn't the only activity in the context of collaborative modeling that could benefit from the facilitation of an interactive large screen. There are several other activities like collaborative sorting or letting different people develop different parts of a model at the same time. Multi user voting however appeared to be a reasonable first approach because it is relatively easy to implement and it provides some significant advantages over other technical voting solutions.

The first advantage is that the concept of placing votes by touch is easy to learn and to apply by the participants, because they can interact with items by directly touching them. Compared to other technical solutions like e.g. laserpointers (Wissen et. al 2001) there is no difference between motor space and visual space. The second advantage is that the number of parallel voting interactions is only limited by the space in front of the large-screen and not by the number of additional input devices. Although not yet empirically proven, it can be expected that the overall time required to prepare and execute a voting under the described circumstances will be shorter than any other current electronic solution. The third advantage it that direct interaction with a common collocated artifact can foster communication and can thereby improve consensus on the final outcome of the task as well as shorten the time to achieve a common result.

Beside its advantages, the collocated work on a common artifact without individual replication is also the most obvious disadvantage of the setting, because all participants have to share a single perspective. This restricts the size of a common artifact to the size of the shared display, because otherwise all participants have to come to an agreement which parts of the model is to be shown and when perspective is to be changed.

4 Requirements to Implement Collaborative Voting

In the course of this section we will describe requirements that have to be met by a voting system in a collaborative modeling environment under the facilitation of an interactive large screen. This list is in no way complete. It rather provides a first set of requirements that we derive from our findings in literature (see section 3) as well as our previous experiences in the joint creation of models (Herrmann 2009).

1. Parallel voting has to be supported

Parallel voting process should be preferred as sequential voting may lead to falsified results and possibly failed consensuses, because there will most likely be one or more dominant participants (Shaw 1981) especially when participants from different managerial levels are present. Furthermore working in parallel is reasonable as it proves to be up to ten times faster than working in a plenary group (Dennis et al. 1994).

2. Every participant has to be identifiable

In order to keep the voting process balanced each participant may use an equal number of votes, thus meaning that it has to be possible to identify them. It is not necessary to sustain the identification during post-processing as this compromises their anonymity and would possibly lead to the domination effect described previously. To ensure an equal number of votes, users only have to be distinguished in the moment of the interaction.

3. Participation has to be as intuitive as possible

In order get the participants involved into voting it has to be easy to learn and use. Furthermore user identification has to be easy to prepare and to apply. Errors are also likely to occur as we are dealing with a highly interactive task involving multiple people interacting on the same surface at the same time. Therefore every participant must have a possibility to remove erroneous votes when she needs to.

4. Different visualizations for different purposes have to be provided

There has to be a visual feedback in the moment of the interaction to ensure users that their vote has been recognized. For post-processing purposes, other visualizations are required that support the moderator during this task. An overview of the voting result when viewing the screen from larger distances has to be provided, as well as means for an accurate numerical analysis of voting results.

The described set of requirements led to the implementation of a prototypical solution that will be described in the following section.

5 Prototype / Implementation

A panel on the side of the SeeMe-Editor contains the controls to set up a voting (see Figure 2 left), containing controls to activate voting as well as to set up visualizations and multi-user options. The panel also contains a list of elements that have received votes, sorted by the number of votes they received. In the following we will describe it´s functionality in detail.

General Operation (Requirements 1, 2 and 4)

Voting can be activated through a single button in the voting panel (see top left corner of Figure 2). The tool distinguishes between multi-user-anonymous-mode and multi-user-distinctive-mode. We start by explaining the anonymous mode, explaining the multi-user-distinctive mode afterwards.

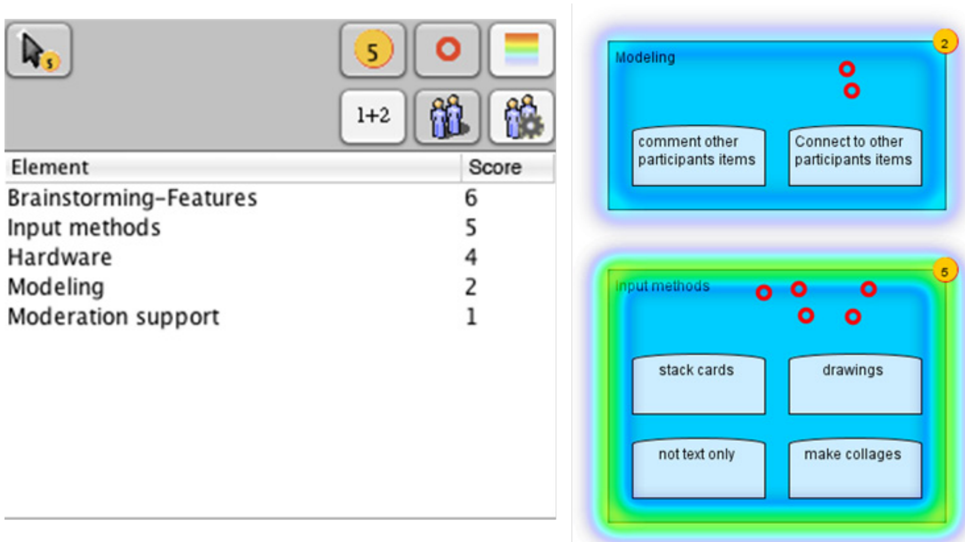

Figure 2: Voting user interface (left) and different visualizations (right)

Multi-User-Anonymous mode

In order to make voting as simple as possible we restricted interactions to short touches (taps) for the voting task. We also provide an error-detection mechanism to enable multi-user voting on single-touch interfaces. Multi-user voting on single-touch surfaces might sound paradox at first but as the voting activity is restricted to a very short interaction (a single tap) it is actually possible to enable multi-user voting on single-touch interfaces. When two users tap at different positions at the same time an error-detection mechanism measures the distance between the detected Tap Down and Tap Released Events. If these positions differ significantly, both votes are rejected and a small error-message is displayed at both-positions for a short period of time. According to our observations this provides sufficient feedback for users to become aware of the conflict and to repeat their action.

Multi-User-Distinctive mode (Requirements 1, 2 and 3)

In multi-user-distinctive mode, voting and user identification require a two-stage interaction. The first tap defines the position of the intended vote as well as the position of the user identification menu. The second tap – inside the user identification menu – relates the vote to the user. The identification menu is designed as a clock face with a maximum of 12 user signatures consisting of a single letter distributed around it (see Figure 3). The signatures are given easily distinguishable colors. The design of the menu was specifically chosen to ensure a simple setup with good memorability and thus low learning effort. Corresponding to the number of voting participants, several of these menus can be displayed at the same time.

Upon the beginning of the voting session, every user is assigned a signature with a corresponding letter on the clock face. The bottom right button inside the voting panel (see left side of Figure 2) provides access to a multi-user setup window. To match the group size, some of the user signatures may also be deactivated completely which leads to e.g. only four letters remaining on the clock face (see right side of Figure 3).

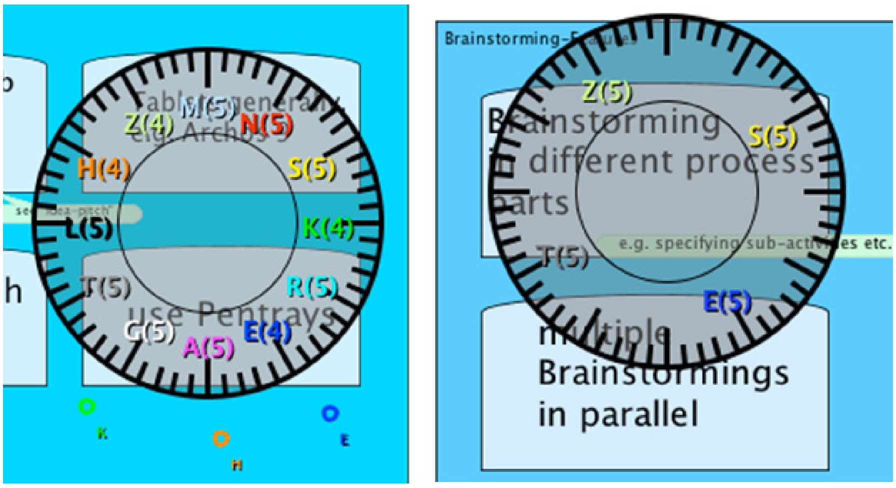

Figure 3: User identification menu for 12 participants (left) and 4 participants (right)

The user distinction mechanism also allows the assignment of individual voting quotas. When a user places a vote, the user's voting quota is decreased by one vote. When the allowed number of votes of a specific user reaches zero, the corresponding signature is no longer available on the clock face, whereby this user may not place further votes.

Voting visualization (Requirement 4)

There are several different requirements for the visualization of voting results leading to three different visualization techniques. Each visualization techniques may be activated separately thus allowing their combination (see Figure 2).

Direct feedback to confirm the success of a voting action is provided by a *dot* on the position of the intended vote. In multi-user-anonymous-mode these dots are colored red, while in multi-user-distinctive-mode their color is chosen corresponding to the user´s color (see Figure 3). Each dot provides a direct feedback to a users tap and thus enables her to recognize if her tap was correct. In multi-user-distinctive mode, the user's signature is also shown at the bottom-right of each dot in the users color (see left side of Figure 3). Additionally, these dots can be tapped in order to remove the corresponding vote.

In order to provide means for easy quantitative analysis, the *Badges* visualization shows small badges in the upper right corner of each element (see right side of Figure 2), containing the current voting count of the element as a numerical value. As stated before, the list of voted elements in the voting panel also provides means for a quantitative analysis (see left side of Figure 2). To provide a quick overview from a distance, we implemented another visualization technique called *Heat maps*, which creates a glowing aura around voted elements, similar to the images produced by infrared cameras (see right side of Figure 2). The brightness, color and intensity of them are related to the amount of votes that the corresponding element received. This enables the users to quickly distinguish between elements that are *hot* and elements that remain *cold*.

6 Lessons Learned and Conclusion

This paper describes the integration of the collaborative, simultaneous usage of an interactive large screen with the practical task of voting. Through our observations and feedback of workshop participants we found a number of requirements that have to be fulfilled when people work together on voting tasks (see section 4).

In contrast to the usage of interactive tabletops, these requirements are partially to the fact that the large screen offers much more opportunities for every participant, to work on different areas of the screen and to move between them. We consider the presented solution as a prototype which still leaves room for improvement, e.g.:

- that the participants can see a list of their own individual votes
- that they can carry out an undo without the need to find the dot (see right side of Figure 2 and left side of Figure 3) which is assigned to their vote

- that an option can be activated with which the dots will disappear after a short feedback has been given to the user or after she has given the next vote. This will help to reduce the influence of the ongoing voting results on other users.

We do not suggest that the current prototype represents a mature solution for voting in the context of the collaborative modeling. However, we consider it as an instrument for further research in this area as it helps to answer questions by experiments and case studies such as:

- How do the voting results differ if the participants give their votes in solitude using their own laptop without observing the behavior of others or communicating with them?
- Which differences can be observed if the votes won´t stay visible after the interaction?
- Does the work on a common artifact intensify the communication between participants and are there any further effects?
- How important is it – and under which conditions – that the users belonging to a touch are identified. How does the identification influence the outcome of the task?

We have only covered one task where multi-user interaction might be feasible in the context of collaborative modeling and creative identification of relevant items of process models. We suggest to develop more applications demonstrating how collaborative, simultaneous work with interactive surfaces can be employed and which allow for further detailed research within practical case studies.

References

Andersen, D. & Richardson, G. (1997). Scripts for group model building. *System Dynamics Review 13* 2, 107-129.

Benko, H., Morris, M., Brush, A. & Wilson, A. (2009). Insights on Interactive Tabletops: A Survey of Researchers and Developers. *Technical Report*

Csikszentmihalyi, M. (1997). *Creativity: Flow and the psychology of discovery and invention*. HarperPerennial: NewYork, USA.

Dennis, A., Hayes, G. & Daniels Jr, R. (1994). Re-engineering business process modeling. In: *Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences*. IEEE. S. 244-253.

Guimbretière, F., Stone, M.C. & Winograd, T. (2001). Fluid interaction with high-resolution wall-size displays. In: *Proceedings of the 14th annual ACM symposium on User interface software and technology*. ACM Press. S. 21-30.

Hawkey, K., Kellar, M., Reilly, D.F., Whalen, T. & Inkpen, K.M. (2005). The proximity factor: impact of distance on co-located collaboration. In: *Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*. ACM Press. S. 31-40

Herrmann, T. (2006). SeeMe in a nutshell - the semi-structured, socio-technical Modeling Method.

Herrmann, T. (2009). Systems Design with the Socio-Technical Walkthrough. In Whitworth, B. and de Moor, A. (Hrsg.): *Handbook of Research on Socio-Technical Design and Social Networking Systems*. Information Science Publishing. S. 336-351.

Herrmann, T. & Nolte, A. (2010). The Integration of Collaborative Process Modeling and Electronic Brainstorming in Co-Located Meetings. In Kolfschoten, G., Herrmann T. & Lukosch S. (Hrsg.): *CRIWG 2010, LNCS* 6257. Heidelberg: Springer. S. 145-160.

Morris, M. R., Huang, A., Paepcke, A. & Winograd, T. (2006). Cooperative gestures: multi-user gestural interactions for co-located groupware. In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM Press. S. 1201-1210.

Renger, M., Kolfschoten, G. & De Vreede, G. (2008). Challenges in collaborative modeling: a literature review and research agenda. *International Journal of Simulation and Process Modelling 4* 3, 248-263.

Rouwette, E., Vennix, J. & Thijssen, C. (2000). Group model building: A decision room approach. *Simulation & Gaming 31* 3, 359-379.

Seow, S.C., Wixon, D.R., MacKenzie, S., Jacucci, G., Morrison, A. & Wilson, A. (2009). Multitouch and surface computing. In *Extended Abstracts CHI 2009*. ACM Press. S. 4767-4770.

Shaw, M. (1981). *Group dynamics: The psychology of small group behavior*, McGraw-Hill College.