A Resilient Interaction Concept for Process Management on Tabletops for Cyber-Physical Systems

Ronny Seiger, Susann Struwe, Sandra Matthes, and Thomas Schlegel

Institute of Software and Multimedia-Technology, Technische Universität Dresden,
Dresden, D-01062, Germany,

{ronny.seiger,susann.struwe,thomas.schlegel}@tu-dresden.de,
sandra.matthes@mailbox.tu-dresden.de

Abstract. Using tabletop devices for controlling complex applications is rather uncommon as interaction on tabletops is still very error-prone. Especially in everyday life settings, there is a multitude of sources of distractions and input errors. The tabletop-based management of cyber-physical systems influencing both the virtual and the real world introduces new challenges with respect to reliability and fault tolerance of the user interface. In this paper, we present an interaction concept for complex management applications on tabletops introducing more resilient gestures for different stages of the interaction. The interaction concept provides more reliable interactions than the basic set of touch gestures does and proves suitable for more complex applications. A user study conducted as part of the work confirms the concept's increased accuracy without influencing the overall usability. As an example, we present a process management application for cyber-physical systems implementing the interaction concept.

Keywords: Process Management, Interaction Concept for Tabletops, Multimodal Interaction, Security and Reliability.

1 Introduction

In his visionary paper “The Computer for the 21st Century” Mark Weiser presented the idea of pervasive computing systems and devices being almost invisible to the users [1]. With the introduction of tabletop computers, these *ubiquitous systems* have reached a new level of seamless integration into the user's everyday life. A tabletop does not only play the role of an input device for digital applications but can also be treated like a piece of furniture positioned in central places within a house, offering space for interaction and for depositing physical objects [2]. However, with the development of applications for the interaction with large screen tabletop devices in everyday life new challenges arise: users might be inattentive; objects may lay on the input surface; clothing may touch the surface; multiple inputs may occur at once; environmental factors may distort the interaction. All of these factors could lead to input errors and thus

to unintended behavior [3]. Hence, current applications usually provide only a limited set of functionality for controlling relatively simple digital objects and systems in order to prevent input errors [3].

Cyber-physical systems (CPS) are a further step towards ubiquitous systems. They combine digital (cyber) and real-world (physical) systems and objects to a new degree [4]. By integrating sensors and actuators with data processing units, embedded computers, and cloud services, a close connection between real objects and virtual objects is created. The emergence of *smart spaces* (smart homes, smart factories, smart offices) shows the increasing importance and spreading of CPS throughout all areas of everyday life. Controlling complex CPS involves not only manipulating virtual objects and processes but also influencing the real world. Therefore, control devices and user interfaces for CPS have to be designed with these new properties in mind. Particularly, input errors and unintended actions have to be avoided as they may have far-reaching consequences in both the cyber and the physical world. This is in contrast to the properties of tabletops, as they are very likely to produce input errors, especially when being used in interfering environments.

We therefore developed an interaction concept enabling more complex applications on tabletops, preventing unintended control actions and thereby allowing more critical operations. As an example, we picked a process management system comprising the visualization, management, and control of processes in a smart home environment. Processes allow the execution of services across application and device boundaries within complex systems in order to facilitate the automation of repeating sequences of activities. Therefore, they play an important role in adaptive and autonomous cyber-physical systems.

The paper is structured as follows: Section 2 discusses challenges that arise when using tabletops for managing CPS applications. Section 3 discusses related research. Section 4 presents our interaction concept for CPS applications on tabletops. Section 5 introduces our process management system. Section 6 evaluates and discusses our interaction concept and management application. Section 7 concludes the paper and shows starting points for future work.

2 Challenges

Tabletops are a new type of interaction device, which provides users with a large multi-touch screen usually mounted horizontally on a stand. A tabletop enables multiple users to interact with its user interface in parallel. Several researchers envision tabletops and tabletop-like devices to become more integrated into people's everyday lives in the near future [1,5]. In a smart home setting, an interactive tabletop could replace the classic coffee table and provide the residents with additional means for infotainment, communication, and control/management tasks.

However, when using a tabletop both as a piece of furniture and as a digital interaction device new challenges arise that have to be dealt with, especially in the context of cyber-physical systems:

- **N:1 Interaction:** In contrast to common touch-based input devices, e.g., tablets and smartphones, tabletops are usually not bound to and operated by only one user. Due to the large screen and interaction surface, several users are able to interact with the tabletop’s user interface in parallel. This can lead to users influencing and interfering with each other [6].
- **Fixed Horizontal Screen:** Being mounted in a fixed position, the tabletop device provides no means for an adapting to the users current position. Therefore, users have to adjust their interaction habits to the properties of the tabletop. For example, the large screen makes it necessary to touch the UI elements holding the finger in a preferably steep angle in order to avoid additional contacts with the input surface by other fingers, the wrist, or a sleeve. Moreover, the interaction distances are likely to be increased compared to smaller screen devices.
- **Distracted Users:** Especially in everyday environments, users are not solely focussed on interacting with the input device. In addition to being in a hurry, a multitude of other sources can distract the user’s attention from the tabletop’s current user interface [7].
- **Environmental Distortions:** Depending on the technology implemented for input recognition on the tabletop, environmental and physical properties can have influence on the accuracy of the touch and gesture recognizers. For example, optical systems based on infrared light are very sensitive towards sunlight and are likely to mistake light beams for intended touch input. Physical objects lying on the tabletop’s input surface may also trigger additional actions or distort the current interaction.

All of these factors have significant influence on the accuracy and usability of the applications developed for the tabletop. These issues can lead to unintended behavior triggered by imprecise or unintentional interactions [3]. This can be a major deal-breaker, especially when interacting with cyber-physical systems, as CPS also affect real world objects and processes [4]. Erroneous input, e.g., additional input events or wrong parameters, can trigger safety-critical actions or even cause damage to objects or people. The task of managing processes in cyber-physical systems increases the importance of correct and reliable user input as processes are intended to support the users by automating repeating activities. Tabletops do not seem to be suitable for the critical tasks of managing and controlling cyber-physical systems. As described in this section, there are several sources of input errors and issues that need to be dealt with when interacting with a tabletop in complex environments [7].

3 Related Work

Our intention is to develop a touch-based interaction concept providing reliable interactions for complex applications. Lambeck et al. [8] argue that traditional interaction and visualization concepts known from desktop applications (WIMP) can not be completely transferred to complex touch-based applications.

The authors present new interaction concepts and paradigms to enable the use of complex applications on tabletops.

In order to reduce frequent input errors and falsely triggered actions on tabletops as observed in [3,7], Lepinski et al. [9] and Moscovich [10] propose to apply marking menus and sliding gestures. From their observations, these interaction elements extend the basic touch gestures [11] and enable more reliable interactions. Providing visual feedback about the state of an interaction supports the user with preventing input errors. ShadowGuide [12] and Escape [13] are two systems employing techniques for the direct visualization of gestures and using advanced UI components for improving touch-based user interfaces.

In [14] Antle et al. developed an interaction prototype for a collaborative learning game on a tabletop device. User studies conducted with the help of this application showed that dragging and dropping gestures are suitable and accepted for collaboration scenarios. In [6] a comprehensive system for tabletop interaction is presented, which applies small sequences of interactions instead of single touch gestures. By using dragging and sliding actions as part of the interaction sequences, the authors achieve an observable reduction of input errors.

As the aforementioned concepts facilitate the design of a fault-tolerant interaction concept for tabletop devices, we will integrate the ideas of interaction sequences comprising sliding and dragging gestures, visual feedback, and customized UI components for touch-based applications into our interaction concept. In [8,15] management systems from the areas of production and disaster control are presented. We will base our process management system applying our interaction concept on these works.

When looking at current applications implemented on tabletop devices, we find that they often only provide relatively simple user interfaces in order to offer a good user experience. Showroom and game-like applications are the predominant use of tabletops nowadays. However, with the advance of tabletops and smart home technologies, we believe that tabletops will also be used for controlling more complex applications involving real-world objects and processes as parts of cyber-physical systems. Current tabletop interaction concepts and user interfaces are not designed with the properties of CPS in mind and are therefore not able to completely meet the requirements for secure and reliable interactions. Therefore, we developed a resilient interaction concept to be used for controlling complex CPS applications on tabletops.

4 An Interaction Concept for Management Applications

With respect to complex management applications, we find that menus and lists to select items and operations from are key elements for interacting with digital objects and information. These applications often provide a multitude of information and operations, which can be best represented in the form of lists and menus. However, the selection of list elements from a complex list is a very error-prone action on tabletops and often leads to an incorrect selection of an entry [9]. Performing an actual control operation in CPS could trigger

critical real-world actions and processes. Hence, the interaction gestures used for selection and execution of an operation on a tabletop device have to be designed to tolerate possible input errors. We developed a multi-stage selection and confirmation procedure to be applied in management applications in order to reduce input errors.

4.1 Setup Stage (Non-critical Actions)

The *Setup* stage consists of non-critical actions for switching between views, configuring parameters, and browsing information. These actions do not manipulate data nor do they lead to an immediate change in a system's or an application's state. They can be reverted easily by simply returning to the previous state or reconfiguring the current selections using simple interactions. Therefore, unintended input does not lead to critical reactions nor does it increase the user's frustration level.

During the Setup stage, commonly known touch gestures can be used because input errors can be tolerated during this stage. In order to browse information and configure parameters, we apply basic touch gestures (e. g., tap, drag, spread, pinch, and slide [11]). As no immediate action is triggered, errors resulting from unintended input can be corrected easily by re-applying these basic gestures.

4.2 Selection Stage (Semi-critical Actions)

The *Selection* stage comprises the selection of menu items available with respect to a digital object. These items can be operations to be performed immediately on the object or additional information to be displayed (e. g., in a newly opened window). These semi-critical actions lead to a direct alteration of the digital (**cyber**) system's or object's state as well as to a change of the current information visualization. During this stage, input errors lead to the selection of a

Fig. 1. State machine controlling the LinMark-Select gesture

wrong list item and thus to a wrong operation to be performed. Reverting the false selection usually is more complex and involves a combination of interactions, which raises the user’s frustration level. Therefore, input errors should be avoided during this stage.

Using simple touch gestures (cf. Setup stage) for this activity can be very error-prone on tabletops as single touch events on the wrong area of the UI can trigger a false selection. Therefore, we introduce a sequence of composite gestures called “LinMark-Select” [9]. This sequence consists of a tap for initiating the selection process, a dragging action for the selection of an item, and a lifting movement for confirmation of the current selection (cf. Fig. 2a). An underlying state machine controls the states of this complex gesture (cf. Fig. 1) [6] and the user interface component provides visual feedback about the action’s current state [12] and selected item (cf. Fig. 2a). The user can easily switch between available list items while holding the finger down as well as revert/cancel the selection procedure.

Fig. 2. LinMark-Select (a) and Safe-Select (b) gestures

With a state machine controlling the LinMark-Select procedure, erroneous input and incorrect touch events are ignored automatically. Only valid input events lead to a change and therefore to a progress of the selection sequence. Moving the finger in the direction of the desired list item is more intuitive and more robust [13] than a single imprecise tap on the list item [14], as with the LinMark-Select gesture the movement direction is the main indicator for the desired menu entry. The state machine allows corrections of the current selection and the cancellation of the complex gesture by simply moving the finger out of the active area. When performing this selection procedure, the user’s attention is drawn towards the list of available options, which should lead to a more precise and correct selection.

4.3 Confirmation Stage (Critical Actions)

The *Confirmation* stage consists of confirming the current configuration of a selected action and thereby triggering its immediate execution. This stage is intended to be the last stage before critical actions having effects on real world (**physical**) objects and systems will be performed or a system’s state is changed significantly. As these actions will influence both the cyber and the physical world, input errors have to be avoided in any case. Reverting wrong selections

and false configurations usually means a considerable effort for the user to undo the triggered actions. Therefore, errors lead to a high level of frustration and may also cause damage to real-world objects.

Using simple touch gestures during this stage can be very error-prone as unintended touch events can lead to a false confirmation of an action to be performed. Therefore, we introduce the “Safe-Select” gesture for confirmation. This swipe gesture is often applied for unlocking a smartphone from the idle state. Performing this gesture involves sliding a finger along a control bar (cf. Fig. 2b). The user receives visual feedback about the current state of the action [12] and the operation will only be executed if the checkbox has been completely moved from left to right.

This confirmation gesture requires the user to pay attention to the active UI area and ignores every touch event outside the control bar. Only a complete slide of the checkbox along the control bar confirms the selected action to be triggered. This interaction is more robust than simple touch gestures for confirmation [10]. It can be canceled easily by moving the finger out of the control bar area or by sliding it back from right to left.

5 Process Management Application on Tabletops

Using our interaction concept, we developed a process management application for cyber-physical systems on a SUR40 tabletop with Microsoft®PixelSense™.

5.1 Process Information

In order to interact with processes in a smart home environment, the information related to available processes, running process instances, and the process execution engine’s state first need to be visualized. In accordance with Shneiderman’s Visual Information Seeking Mantra [16], we first present an overview of the complete process management system (cf. Fig. 3). Here, the user is able to zoom into and filter information concerning the *System State*, *Process Models*, and *Running Instances*. After selecting a specific view or process object, the application presents additional details regarding the selection.

System State Information. The *System State* view gives a concise overview of the process engine’s current state including a log of recent events and errors that have occurred during process execution.

Process Model Information. The *Process Model* view presents a list of categorized processes (process models) available for execution. An additional property panel containing details can be opened for every process model. These details include general information (e.g., name, type, and input/output parameters), a log for process instances executed according to the particular model, and a graphical, scalable representation of the process model. Multiple property panels can be opened in parallel and moved freely on the UI by dragging gestures, allowing simultaneous and collaborative interaction among multiple users.

Process Instance Information. The *Running Instances* view displays an overview of current and recently executed processes (process instances). Similar to the process model view, a property panel can be opened with respect to every process instance. This panel presents general information including the current progress of an instance, data values, a log for this particular instance, and a graphical representation of the corresponding process model.

Fig. 3. Overview screen of the management application

5.2 Process Control

Setup Stage. In the context of our application, the *Setup Stage* applying only simple touch gestures comprises browsing through processes and process information, switching between different views, and setting parameters for the instantiation of processes. As these actions do not have immediate effects on the system or processes, simple touch gestures are sufficient for executing these setup actions.

Selection Stage. During the *Selection Stage*, an entry from a list of available menu entries is selected using the LinMark-Select gesture. We use this gesture for selecting operations to be performed on particular process models and instances. With every process, we provide a list of possible actions, e.g., opening a new window for starting an instance, changing its state, and opening a property panel (cf. Fig. 4a). These actions do have an immediate effect on the system's state and presentation of information and are therefore selected using the more resilient LinMark-Select gesture.

Confirmation Stage. The *Confirmation Stage* leads to the immediate execution of a real world process or a significant change in the system’s state. Therefore, we apply the more robust Safe-Select gesture for confirming critical actions to be immediately triggered after the gesture. In our application, changing the state of a process instance (start, stop, pause, cancel) is confirmed using the Safe-Select gesture (cf. Fig. 4b) as well as the addition, deletion, and modification of process models. During the confirmation stage, only the current confirmation window can be interacted with allowing only one user to confirm or cancel the execution of the selected action.

Fig. 4. The LinMark-(a) and Safe-(b) Select gestures used for process management

6 Evaluation and Discussion

An evaluation with respect to Nielsen’s usability heuristics [17] shows good results regarding the usability of our interaction concept and management application. Nine out of ten of Nielsen’s heuristics can be evaluated as “good” or even “very good”. As managing processes in a smart home environment is yet not very common, users first have to get used to the terminology we applied in the process control system. The management system is currently in the stage of a prototype not including any help or documentation to support the user, which is the only major point of criticism with respect to the usability heuristics.

6.1 User Study

In order to evaluate our concept and application on a qualitative level, we conducted a small user study among ten computer science students and university employees in the age group from 20 to 39. All of the participants were already experienced in using touch devices and familiar with basic touch gestures. We chose this group of test subjects because they are open to interacting with new

technologies and they are likely to understand the concepts behind the process management and CPS topics more quickly.

The user study was divided into two main tasks. First, the subjects had to use a file management software on the tabletop to open and copy some files using basic touch gestures, e.g., tap, long tap, double tap, and drag. Second, the subjects had to interact with our process management application to start and stop some processes as well as retrieve additional information regarding particular processes. The second task involved using our interaction concept including the LinMark-Select and Safe-Select gestures.

Using only basic touch gestures during task one, the participants struggled a lot with input errors triggered by clothing and other parts of the hand/arm when selecting items from a complex list as well as when opening files and the context menu on the SUR40 tabletop. The majority agreed that using only these basic gestures is very unreliable and cumbersome, which increases the effort of interacting with a complex application on a tabletop device. Despite a higher initial learning effort, the users confirmed the reduction of input errors and unintended behavior without decreasing the overall interaction efficiency for our interaction concept as they got accustomed to the newly introduced gestures quickly. All in all, our test subjects can envisage using our application for managing their smart homes on a tabletop.

6.2 Discussion

The interaction concept presented in the previous sections is not meant to be a complete set of interaction gestures necessary for a complex management application. It rather introduces new gestures/interaction sequences as well as user interface components that can be applied complementary to an existing interaction concept. Based on related work, existing concepts and gestures are put into a new context with respect to our application scenario: process management in CPS. The LinMark-Select and Safe-Select gestures should be considered when designing new interaction concepts for safety-critical applications as they reduce input errors and accidental actions. At the same time, acceptance and efficiency of the interactions are preserved as shown by our user study.

In order to further evaluate the management application for a broader target audience, the rather complex application has to be simplified with respect to its wording and its relatively high amount of text should be enhanced by graphics and icons. Elements from gamification can be used to introduce users to our interaction concept and to simplify access to the management application.

Although we started to design the interaction concept taking into account the high error rates of our optical tabletop device, several of the aforementioned sources for input errors also exist for other types of input recognition systems (e.g., capacitive touchscreens) and tabletops in general. When controlling critical actions in cyber-physical systems, there is a general need for failure tolerant user interfaces including multi-stage selection and confirmation procedures in order to prevent unwanted actions to be executed.

7 Conclusion and Future Work

In this work, we presented an interaction concept for selecting and executing safety-critical actions in cyber-physical systems (CPS). Tabletop devices will become more and more integrated into people's everyday lives as described by Mark Weiser's vision of ubiquitous systems. Due to a multitude of sources of input errors and distractions, there is a need for a fault-tolerant user interface especially for tabletops. We introduced a set of new gestures and UI elements dividing the interaction with complex management applications into three stages of different criticality for setup, selection, and execution actions. The new interaction concept can be applied to complex applications on touch-based devices in general preventing the user from adapting himself/herself to the properties of the device. With the newly introduced LinMark-Select and Safe-Select gestures we are able to reduce input errors without implementing additional error recognition and filter algorithms. User tests confirmed the reduction of input errors and the feasibility of our approach.

As part of our future work, we plan to test our interaction concept in a real-life smart home setting. Currently, the process management prototype is not connected to an actual process execution engine. The integration of both systems will be part of our cyber-physical systems research project. An extensive user study involving the execution of automated processes in a smart home and their control by a tabletop device will be conducted in the project's final stage in order to evaluate the overall usability and test the functionality of our integrated CPS infrastructure. We also plan to extend the management application to also comprise the management of smart home sensors, actuators, live camera streams, and service robots resulting in a comprehensive management application ready to be used in an actual smart home.

Acknowledgments. This research has been partially funded within the VICCI project under the grant number 100098171 by the European Social Fund (ESF) and the German Federal State of Saxony.

References

1. Weiser, M.: The computer for the 21st century. *Scientific American* 265(3), 66–75 (1991)
2. Shen, C., Ryall, K., Forlines, C., Esenther, A., Vernier, F.D., Everitt, K., Wu, M., Wigdor, D., Morris, M.R., Hancock, M., Tse, E.: Informing the design of direct-touch tabletops. *IEEE Comput. Graph. Appl.* 26(5), 36–46 (2006)
3. Ryall, K., Forlines, C., Shen, C., Morris, M.R., Everitt, K.: Experiences with and observations of direct-touch tabletops. In: *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems, TABLETOP 2006*, pp. 89–96. IEEE Computer Society, Washington, DC (2006)
4. Lee, E.: Cyber physical systems: Design challenges. In: *2008 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*, pp. 363–369 (2008)

5. Milne, A.J.: Entering the interaction age: Implementing a future vision for campus learning spaces. *Educause Review* 42(1), 12–31 (2007)
6. Block, F., Wigdor, D., Phillips, B.C., Horn, M.S., Shen, C.: Flowblocks: A multi-touch ui for crowd interaction. In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, UIST 2012*, pp. 497–508. ACM, New York (2012)
7. Hornecker, E.: 'i don't understand it either, but it is cool' - visitor interactions with a multi-touch table in a museum. In: *3rd IEEE Intern. Workshop on Horizontal Interactive Human Computer Systems, TABLETOP*, pp. 113–120 (October 2008)
8. Lambeck, C., Kammer, D., Weyprecht, P., Groh, R.: Bridging the gap: advances in interaction design for enterprise applications in production scenarios. In: *Proceedings of the Conference on Advanced Visual Interfaces 2012, Capri Island, Italy (2012)*
9. Lepinski, G.J., Grossman, T., Fitzmaurice, G.: The design and evaluation of multitouch marking menus. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2010*, pp. 2233–2242. ACM, New York (2010)
10. Moscovich, T.: Contact area interaction with sliding widgets. In: *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, UIST 2009*, pp. 13–22. ACM, New York (2009)
11. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: User-defined gestures for surface computing. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2009*, pp. 1083–1092. ACM, New York (2009)
12. Freeman, D., Benko, H., Morris, M.R., Wigdor, D.: Shadowguides: Visualizations for in-situ learning of multi-touch and whole-hand gestures (2009)
13. Yatani, K., Partridge, K., Bern, M., Newman, M.W.: Escape: A target selection technique using visually-cued gestures. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2008*, pp. 285–294. ACM, New York (2008)
14. Antle, A.N., Bevans, A., Tanenbaum, J., Seaborn, K., Wang, S.: Futura: Design for collaborative learning and game play on a multi-touch digital tabletop. In: *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction, TEI 2011*, pp. 93–100. ACM, New York (2011)
15. Nebe, K., Klompmaier, F., Jung, H., Fischer, H.: Exploiting new interaction techniques for disaster control management using multitouch-, tangible- and pen-based-interaction. In: Jacko, J.A. (ed.) *Human-Computer Interaction, Part II, HCI 2011*. LNCS, vol. 6762, pp. 100–109. Springer, Heidelberg (2011)
16. Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: *Proceedings of the IEEE Symposium on Visual Languages*, pp. 336–343 (1996)
17. Nielsen, J.: *Ten usability heuristics* (2005)