Tool-based Task Modelling of Medical Human Machine Interfaces

Sebastian Feuerstack[1]

Cognitive Analysis Tool, HMI Design, Task Analysis, Medical Applications.

Abstract

We present an extension to CogTool to model tasks for clinical HMIs. CogTool focuses on analysis of graphical desktop user interfaces based on a fixed palette of interface widgets. With our approach the use of custom instruments can be modeled and translated to a cognitive architecture's language. Although based on wire-framing, we can model physicians moving around to control a complex, distributed HMI, such as it is often used in clinical applications. We describe the modelling of an x-ray acquisition procedure that has been performed together with the device manufacturer and are currently preparing a study to evaluate how the modeling process of the tool influences HMI design decisions.

Introduction

Operator's task performance is usually evaluated by systematically studying operators controlling a system prototype in a setup that matches as close as possible the reality. Such an approach promises high-quality evaluation results in terms of data precision, but requires a realistic prototype and subjects that represent the targeted audience. The number of participants, the complexity of the tasks to evaluate and the amount of design alternatives to test is limited by costs and time, which often results in an evaluation performed at the very end of the design, with view subjects, very basic tasks and with a final prototype.

The idea of cognitive modeling is to simulate operators based on psychological and physiological plausible models. These models predict operator behavior and do not require a new setup of a study for each new version of a system. The quality of prediction models depends on the degree of model validity. Model-based predictions, gained by simulation runs can be generated much faster. Therefore, the amount of design variants and the complexity of tasks that can be evaluated are much higher. Additionally, these predications can be generated for each design cycle as an additional source of information indicating the efficiency of an HMI.

Cognitive modeling is still for experts. They translate what they learn from earlier studies with domain experts to a virtual cognitive operator model in order to predict behavior as best as possible. Often it is questionable if validated models of a specific setup can be directly transferred to a new use case. Thus, the cognitive models often focus on applying established, but basic models such as KLM, GOMS, and Fits law for instance.

CogTool [John et al. (2004)] is a cognitive modeling tool that predicts human performance for storyboarded graphical user interfaces. In the past years it has been applied in real-world software development processes with promising results. A recent empirical study indicates that novice modelers significantly made more correct recommendation to improve a user interface than without tool-support [Hong and St Amant (2014)].

So far HMIs cannot be modeled for cognitive analysis with CogTool. The storyboarding of CogTool is limited to graphical interfaces with a recurring set of fixed widgets. HMI happens

---

[1] Oldenburger Forschungsinstitut für Informatik (OFFIS), Oldenburg

in a 3 dimensional world, where people move and look around. Physiological motor actions take much more time and are often beyond moving a mouse in a fixed, sitting position.

We present ongoing work in advancing the tool-based task performance prediction by extending the analysis scope from desktop user interfaces to HMIs such as they are used in clinical environments. This tool-supported process promises on the one hand to significantly ease and fasten the generation of human behavior predictions since and on the other hand it represents a pragmatics approach since predictions cannot be compared absolutely but only relatively by comparing different HMI design alternatives.

Approach

Medical HMIs are designed for their specific application. Manufacturers mostly implement their own interfaces and ergonomic guidelines. Therefore, we propose cognitive instrument usage blocks that replace the predefined WIMP widgets of CogTool. These building blocks separate expertise in modelling activities: Cognitive modelers design these blocks that apply the functionality of a specific cognitive architecture and specify how an instrument is controlled and observed in terms of psychological or physiological actions.

Domain experts on the other hand can use these blocks without any expertise in cognitive modeling as the syntax and semantic foundation for the storyboarding of tasks and processes. The storyboarding is performed by a demonstration of tasks in an environment consisting of interconnected photos that are annotated with instrument positions.

Modelling Health HMIs

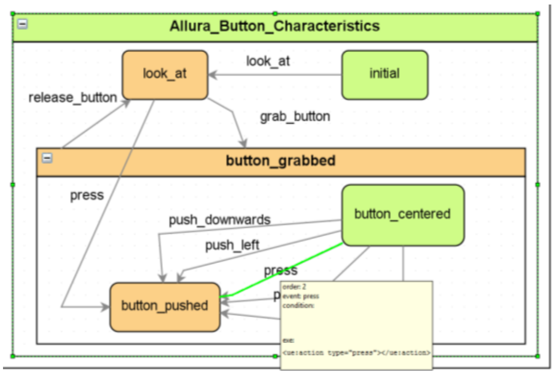


Fig.1: An instrument block specifying a button that can be pressed but also pushed in four directions. The transition names reflect the commands that the tool user can apply on the button. Within a transition a translation of an operator's action into a concrete set of cognitive actions is specified.

Instrument Block Modeling

Instruments are modeled from a cognitive perspective:

- how the instrument is thought to be used
- how the instrument's functionality is perceived/monitored by the operator

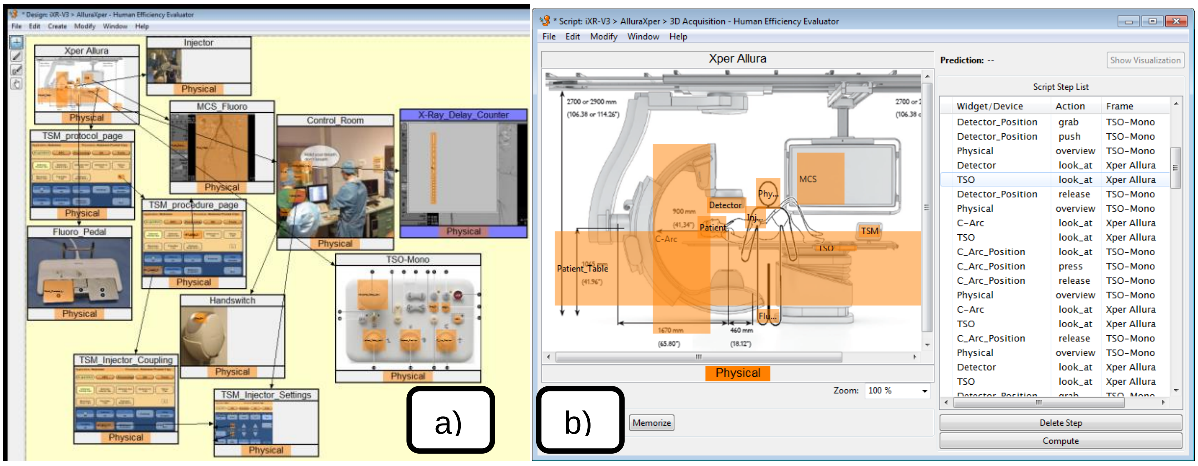
For cognitive modelers we use state charts to specify the cognitive instrument usage blocks. A state chart refers to a set of executable actions that are executed by state transitions. An action can trigger psycho-motor processes such as head, eye or body movements or mental processes, like remembering and recalling information. Figure 1 depicts a specific button of a medical equipment manufacturer that can be pressed and additionally pushed to several directions. Transitions between the states define the possible physical manipulations of the button but also establish usage constraints. Thus, the button of figure 1 is not suitable for blind operation: the operator needs always to look at it to be able to press or grab and push it into one direction thereafter.

Which processes can be used within the transitions depends on the concrete cognitive architecture used. CogTool applies ACT-R [Anderson et al. (2004)] and the vocabulary available is comprised of ACT-Simple [Salvucci and Lee (2003)]. In our case we apply CASCaS, a cognitive architecture that has been validated for HMIs in the Aeronautics [Luedtke et al. (2009)] and Automotive domain [Wortelen et al. (2013)].

Based on earlier works we already have a standard set of instrument blocks from other application domains defined that could be re-used for the health HMIs. This includes hardware buttons, pedals or indictor LEDs. Only two further instrument blocks were added: One to reflect the press and bush buttons (c.f. fig 1) and one that specifies the supported adjustments of the patient table.

Environment Modeling

Complex medical HMIs are distributed in an environment like an x-ray room. The way HMI are distributed around the working place can have a huge impact on the operators' task performance. But also visual and mental workload can be affected for instance if information need to be collected from various places and stored in mind to be recalled later within a complex task.



Script Step List

Widget/Device	Action	Frame
Detector_Position	grab	TSO-Mono
Detector_Position	push	TSO-Mono
Physical	overview	TSO-Mono
Detector	look_at	Xper Allura
TSO	look_at	Xper Allura
Detector_Position	release	TSO-Mono
Physical	overview	TSO-Mono
C-Arc	look_at	Xper Allura
TSO	look_at	Xper Allura
C_Arc_Position	look_at	TSO-Mono
C_Arc_Position	press	TSO-Mono
C_Arc_Position	release	TSO-Mono
Physical	overview	TSO-Mono
C-Arc	look_at	Xper Allura
TSO	look_at	Xper Allura
C_Arc_Position	release	TSO-Mono
Physical	overview	TSO-Mono
Detector	look_at	Xper Allura
TSO	look_at	Xper Allura

Fig.2: a) The environment model of an x-ray system. A root frame (top left) illustrates in physical exact dimensions the location of the various sub-systems with an HMI interface. Specific subsystems interactions can then be "wire-framed". Transitions between frames are "look at", "walk to" or "run to" actions. b) An excerpt of the storyboarding script (right) that is generated by demonstrating a procedure on an interactive wire-framed prototype (left).

Figure 2a) shows the environment modeling composed of a root frame representing the initial position of the physician in front of an x-ray machine that is composed of several instruments and displays to be observed before and during x-ray acquisition. The interactions can be detailed. For instance the interaction with a touch-display is wire-framed to illustrate the control options over several different screens. In the same way room changes (to trigger the x-ray) or a focus on a specific instrument (e.g. the pedal under the patients table) is specified. The entire HMI for a standard x-ray procedure that is controlled by one physician was captured by 12 wired frames. A total of 34 instrument annotations were required. The first version of a new HMI was composed of 23 frames with 23 instrument annotations to demonstrate the same procedure. The description of each environment model did not take longer than 10 minutes each. Since the overall hardware setup remained the same, several of the frames of the current HMI could be copied for the new HMI environment model.

Storyboarding

By storyboarding the procedure of an x-ray acquisition can be demonstrated. The demonstration is started with showing the root frame like depicted in figure 2b. Each annotated instrument (indicated by the orange boxes) can be clicked to perform an action. A *look_at*, *run_to*, or *walk_to* action usually ends up in a new frame displaying the new situation. Other actions include for instance *check_indicator_led* or *monitor* a device (for instance to observe the moving c-arm of the x-ray). The availability of actions to be performed on a specific instrument depends on the state chart of the corresponding instrument block. The sequence of demonstrated actions is recorded (c.f. right part of fig. 2b). The x-ray procedure has around 120 basic actions and could be recorded in around 15 minutes with the tool.

Current State and next Steps

We enhanced CogTool to support modeling interaction with HMIs. In a clinical application use case that was performed together with three human factors experts from a medical instruments manufacturer we noticed that the modeling process with the tool motivated the experts to re-think some of their design decisions. Also it was mentioned that "the structured process of modeling and recording the procedure in a sequential way made them better understand their own use case".

Currently, we are finalizing the tool so that we can generate task performance time predictions for both HMI variants of the x-ray acquisition procedure. In a study that we are currently preparing, we intend to investigate in two aspects:

1. Do task performance comparisons with the tool influence design decisions for clinical HMIs and can these decisions be explained by knowledge gained by the tool usage?
2. Does the tool-supported modelling process have an impact on HMI design decisions and can these decisions be explained by knowledge gained by the tool usage?

Literature

[Anderson et al. (2004)] John R Anderson, Daniel Bothell, Michael D Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. An integrated theory of the mind. *Psychological review*, 111 (4): 1036, 2004.

[Hong and St Amant (2014)] Kyung Wha Hong and Robert St Amant. Novice use of a predictive human performance modeling tool to produce ui recommendations. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 2251–2254. ACM, 2014.

[John et al. (2004)] Bonnie E. John, Konstantine Prevas, Dario D. Salvucci, and Ken Koedinger. Predictive human performance modeling made easy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 455–462, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8.

[Luedtke et al. (2009)] Andreas Luedtke, Lars Weber, Jan-Patrick Osterloh, and Bertram Wortelen. Modeling pilot and driver behavior for human error simulation. In VincentG. Duffy, editor, *Digital Human Modeling*, volume 5620 of *Lecture Notes in Computer Science*, pages 403–412. Springer Berlin Heidelberg, 2009. ISBN 978-3-642-02808-3.

[Salvucci and Lee (2003)] Dario D Salvucci and Frank J Lee. Simple cognitive modeling in a complex cognitive architecture. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 265–272. ACM, 2003.

[Wortelen et al. (2013)] Bertram Wortelen, Martin Baumann, and Andreas Lüdtke. Dynamic simulation and prediction of drivers' attention distribution. *Transportation research part F: traffic psychology and behaviour*, 21: 278–294, 2013.