

VISUALIZATION OF USERS' ACTIVITIES IN A SPECIFIC ENVIRONMENT

Zdenek Mikovec
Ivo Maly
Pavel Slavik

Dept. of Computer Science and Engineering
Karlovo namesti 13, Czech Technical University, Prague
Prague, 121 35, CZECH REPUBLIC

Jan Curin

IBM Research, Voice Technologies and Systems
V Parku 4
148 00 Praha 4, CZECH REPUBLIC

ABSTRACT

Evaluation of user interface design is usually based on usability testing methods. In this paper we analyzed and described the user behavior in the form of the user model. The user model is built on a data set which was acquired through observation of user behavior. The criterion we use for the evaluation of the usability of interface design is the user cognitive load. We present different tools for visualization, simulation and easier analysis of the user model. Because each of the methods and tools presented use only part of the user model, in the end we present methods on how to use the complete user model to correctly describe user behavior. In order to understand such a complex model we need to have a proper visualization tool. The concept of such a visualization tool is presented at the end of this paper.

1 INTRODUCTION

Each program has a user interface through which it communicates with a user. The design and implementation of such a user interface is a rather complicated task. There are two classes of properties that should be taken into account. The first class contains the properties of the device the program runs on. The second class contains the properties of users that will use the program by means of a user interface. In order to investigate thoroughly the user activities many methods have been developed. These methods allow for creation of a user model that reflects the user activities investigated and also allows for future simulation of these activities. According to these activities it is possible to design a user interface that will conform to the user needs. There exist many definitions that explain what the user model is. One of the most commonly used is the following: The *user model* represents the computer model about the user's state of knowledge and it refers to the knowledge, intentions, goals, preferences, etc. of the user.

User models are created by means of collected data that reflect the user activity when the user performs some typical tasks with the program under development. These data are collected using various methods – mostly observation of the user or recording the user activities by means of various tools (like tracking position of the user, recording the individual interaction steps during particular task solving etc.).

The amount of data collected during the user activity tracking is very large. For proper understanding of the meaning of these data, visualization methods must be used. When properly visualizing these data it is possible to discover certain behavioral patterns otherwise unseen. This means that the use of visualization methods will contribute to the design of the high quality user interface.

The user model can be expressed in a formal way in various forms. The forms commonly used are represented by means of graphs where relations among various properties of the user activities are expressed (Tazari, Grimm, and Finke 2003). In our approach the user model describes user activities in temporal and spatial dimensions. This model should allow for computation of the user's region of interest (ROI) and thus determination of the cognitive load which is important for better program design.

This paper specifies various aspects of the use of visualization of the user model. There will be summarized possible approaches to visualization in particular stages of the user activity tracking and also benefits of the visualization will be shown. At the conclusion of this paper a possibility of integration of individual visualization methods under one umbrella will be discussed. Moreover we will discuss visualization issues in conjunction with other visualization methods by means of which we can get easier interpretation of our data. We have in particular in mind, the visualization of task models. In such a way we can get a synergy by means of which we can understand the data much better than by using standard visualization methods.

2 RELATED WORK

There are several tools that allow user activity visualization. We can divide them into several categories according to the visualization used and the usage of the tool.

User activities can be presented in a form of timelines (Harrison, Owen, and Baecker 1994), where the particular activities are recorded and displayed in sequential order by their timestamp. Another timeline visualization tool is Noldus Observer (Noldus 2007). It is a tool for analysis of data from a usability test or experiment. It is able to show the user interactions in form of video recording and timeline visualization. For the purpose of complex user model analysis those tools do not offer appropriate visualizations.

UbiREAL (Nishikawa et al. 2006) is a tool focused on visualization of home appliances status in consideration to their context of use. The main goal of the tool is to analyze the cooperation between several ubiquitous devices in one room. The tool allows simulation of application behavior in virtual environment, while being controlled by a real controller. A similar approach is used in the work of Holm et al. (Holm et al. 2002), where the 3D visualization is used for evaluation of user reaction in the virtual reality system. The data about a user are collected and presented in various graphical forms in the visualization. Luyten et al. (Luyten, Vandervelpen, and Coninx 2005) uses 3D visualization of the environment to analyze the ubiquitous applications. They present the virtual environment that is capable of showing how the tasks are distributed to applications in the environment. A problem of these tools is that they can't visualize user interactions.

Patterson et al. (2003) show how low level sensor information can be used for high level behavior description. Their method is automatic but is focused only on the position of the user and do not take into account the user's activities.

3 USER MODEL STRUCTURE

As mentioned above, the user model represents the computer model about the user's state of knowledge and it refers to the knowledge, intentions, goals, preferences, etc. of the user. The data collected for a user model definition can have various interpretations according to the goal the user interface designer has in mind. In our case we will interpret the collected data by means of identification of the cognitive load. The cognitive load is a very important issue in the process of designing a user interface. Briefly speaking: each user has at her/his disposal a certain mental capacity that can be used for solving some particular problem for which the program should be used. In the case when the user interface is not designed properly, a substantial part of her/his mental capacity is spent on the interaction with the program (trying to find a proper item

in a menu, looking for a mechanism to invoke help etc.). Our attention will be focused on visualization of data related to user activities that are linked up with cognitive load.

To show the complexity of the task investigated we will consider a relatively complex case where we assume the design of a user interface for mobile applications. Here the user activities have several aspects that should be treated:

- motion of the user in environment where s/he should perform interaction (e.g. in interior of a building)
- behavior of the user more in detail (like her/his gestures, sitting, standing up, communicating with other persons etc.)
- activities that represent user's interaction with a program (picking objects on the screen, selection from menus, entering data etc.)

In such a way we have introduced three distinct aspects of user activities that can be treated as three levels of detail of the user model (see Figure 1). These various levels allow the designer to investigate separately the activities from different points of view and thus can be handled separately.

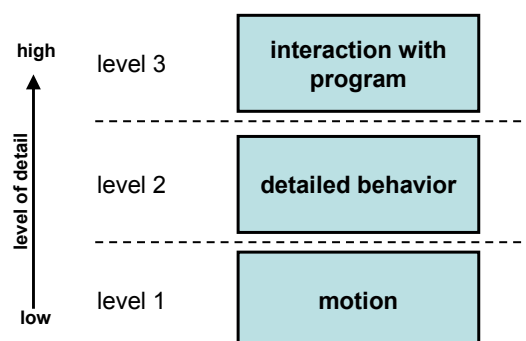


Figure 1: User's activities organized in three levels of the user model.

In our approach we have developed several tools by means of which we can visualize user activities and analyze individual issues from the three levels mentioned above. The tools already implemented are:

- tools for observing users interacting with mobile devices, introduced in Section 4,
- tool for observing history of user motion in specific environment, details are given in Section 5,
- tool for detail analysis and 3D visualization of users' behavior, described in Section 6.

Currently, each level of the user model contains

distinct information dedicated to each tool, which is suitable for different analysis but the knowledge of the whole model should be interconnected. In the next chapters we will present how to use each part for user cognitive load analysis.

4 INTERACTION WITH MOBILE DEVICES

Interaction between the user and mobile devices is the most detailed view of our user model. Analysis of the interaction allows one to consider how the cognitive load of the user changes during the work with the mobile device. We use usability testing methodology (Nielsen 1994) to collect and analyze this level of the user model. An advantage of this approach is that we can directly estimate the user cognitive load from the available data. The cognitive load is rising when the user is doing interactions which are not expected by the developer, or the user is not sure how to continue to finish her/his task, and so on. Of course, there is some basic level of cognitive load, which is minimal and represents the cognitive load that is necessary for processing the task.

4.1 Representation of User Model Data

This level of the user model process the user interaction data. The data can be in the form of user events (e.g. button presses) or text information (e.g. annotations about user interaction). These data are stored typically in form of a log file. Data for a user model can be acquired by means of a controlled experiment, e.g. usability testing or simulation of the user behavior.

Usability testing methods typically use the ability of the standard video recording tools, which do not provide the desired data format. They usually use both audio and video recordings and there are several possibilities that can be recorded. First of all, the screen of the device can be recorded. This provides us the information of what activity the user performed and when. The second possibility is to record activity on peripherals of the device (e.g. portable computer), like user moves over the keyboard or the whole device. The third possibility is to record the overall user activity. What is also very important is the audio recording. We usually record user voice and the sound from the application. User voice is important for a better understanding of user thoughts. We use the method called Thinking aloud, which is often used during the test. Using this method we encourage the user to tell us what s/he has in her/his mind. Such audio-video data must be transformed into the text form, typically as observer annotations. These annotations represent the facts about the user interaction, her/his ability to complete her/his tasks and so on.

Usability testing methods also allow for collection of the data from the program or from the device itself. For

example, we can store phone numbers the user was calling or we can store the numbers the user has typed on the keyboard.

4.2 Developed Tool

When all the data are collected and transformed they need to be analyzed. We developed a tool that allows analysis of the user model and estimation of the user cognitive load. The level of cognitive load is indirectly described by the annotations and user events. Annotations and events are presented in form of a timeline which is enriched by other models, in our case by a task model (see Figure 3). The timeline is presented as a sequence of annotations ordered according to the start time of the annotation. Each annotation belongs to a category that is presented by the color of the annotation. The color scale represents the severity of the annotated problem – red is more complicated for the user, white is easier for the user. The information from the author of the annotation is available as a tool tip text. Annotations are further segmented by the task, they are related to. Annotations for the first task are in the first box, annotation for the second task are in the second box and so on. Moreover, if the annotation is from the first subtask it is on the first line of the task box, when from the second subtask, it is on the second line and so on. This way it is much easier to identify when the annotations occurred. Task boxes are connected with the Task tree (as shown in Figure 3 on the right).

4.3 Usage and Validation of the Visualization

The amount of the cognitive load is primarily described by two factors. The first factor is the amount of user events. If more user events occur in a short time we assume that the user cognitive load was high because of the interaction with the mobile device. The second factor is the severity of the user annotation. If the severity of the annotation is high we assume that user cognitive load was high and the user had to solve the task solving problem. These factors are easy to find using our visual tool and the exact user activity is also available as tool tips of each annotation and event in the timeline.

The visualization tool also allows for other interesting transformations of the data. It is possible to show the annotations from more than one user. This way we can show user interactions, analyze the problematic parts and compare them with other users or with optimal simulated behavior. A complete design of the tool and its usage and validation was presented in Maly and Slavik (2007).

5 USER MOTION IN SPECIFIC ENVIRONMENT

In this paper we are concentrated on a specific environment, the mobile environment, and we should have some

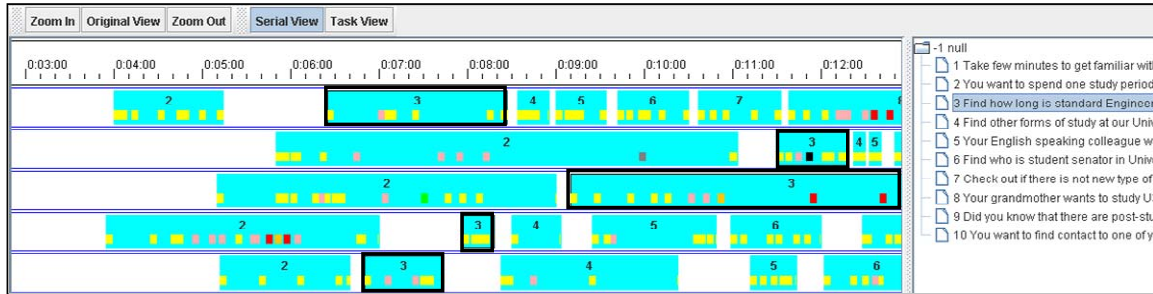


Figure 2: Sequential view of 5 tests.

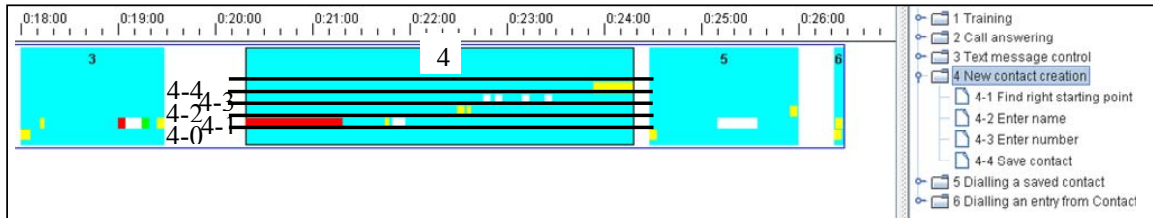


Figure 3: Binding between task model and tasks in timeline.

idea about the user motion in this environment. For the sake of simplicity we can imagine that a user walks in an exhibition hall, in a gallery, in a museum etc. When following her/his path we can discover some patterns (like concentration on specific areas – where e.g., are exhibited topics of her/his interest like some products, pictures etc.). In such a way it will be possible to discover attitudes of individual users (this may help to define groups of users that share common interests etc.). This kind of evaluation of the user behavior is based on visual analysis of the acquired data.

Let us define several important issues that are typical for visual analysis of data of the type mentioned. The basic activity is the navigation in the given environment (e.g. in an exhibition hall). This means that the user is trying to get from position A into position B. What is interesting is the investigation of a strategy for selection of the trajectory that has been chosen by the user. This information can give important information about the user background on which the selection of strategy was based. Another important issue is investigation of the speed of motion of the user. When the user moves slowly this fact might signal that the user attention has increased (there is something interesting to watch).

The derived movement activities like returning to the beginning of the trajectory (or some repetitive movement patterns etc.) could carry important information about the user. Also points on the trajectory where the user slows her/his movement could coincide with points where the user’s attention increased in the previous walkthrough. In the case where the user stopped at some point, the duration of the time period the user does not move is also important information. All these data might be used later for the region of interest (ROI) calculation and visualization and for cognitive load calculation and visualization.

5.1 Representation of User Model Data

For the analysis of the movement patterns the data must contain the positions of the user or users in a specified area. There are two types of information in the model. The first one is the description of the environment (room, equipment of the room, etc.) in particular the coordinate system, and the second one is the position stored in the form of a sequential log of the user position during the time interval. Currently, our tool supports the description of the single room or interconnection of multiple rooms.

5.2 Developed Tool

The tool for visualization of conference room history was written in C# programming language. The main features of the tool have been mentioned in the text above. In the main window of the tool there is a time range control for specifying the desired slice of history (see Figure 4). In the bottom of the main window there are two sliders by means of which it is possible to specify the time interval in which we would like to investigate the trajectories assigned to individual persons. By means of the upper slider we can adjust the time when the investigated time interval begins. The position of the lower slider defines the end of this time interval. In such a way we can dynamically adjust the interval of our interest. When the desired time interval is adjusted it is possible to run the animation by clicking the Play button. Another important functionality of the tool is the possibility to select an individual trajectory by clicking closely to the selected trajectory. The trajectory will be marked and the user can investigate specific motion of the tracked person. Besides observation of individual trajectories it is possible to investigate global

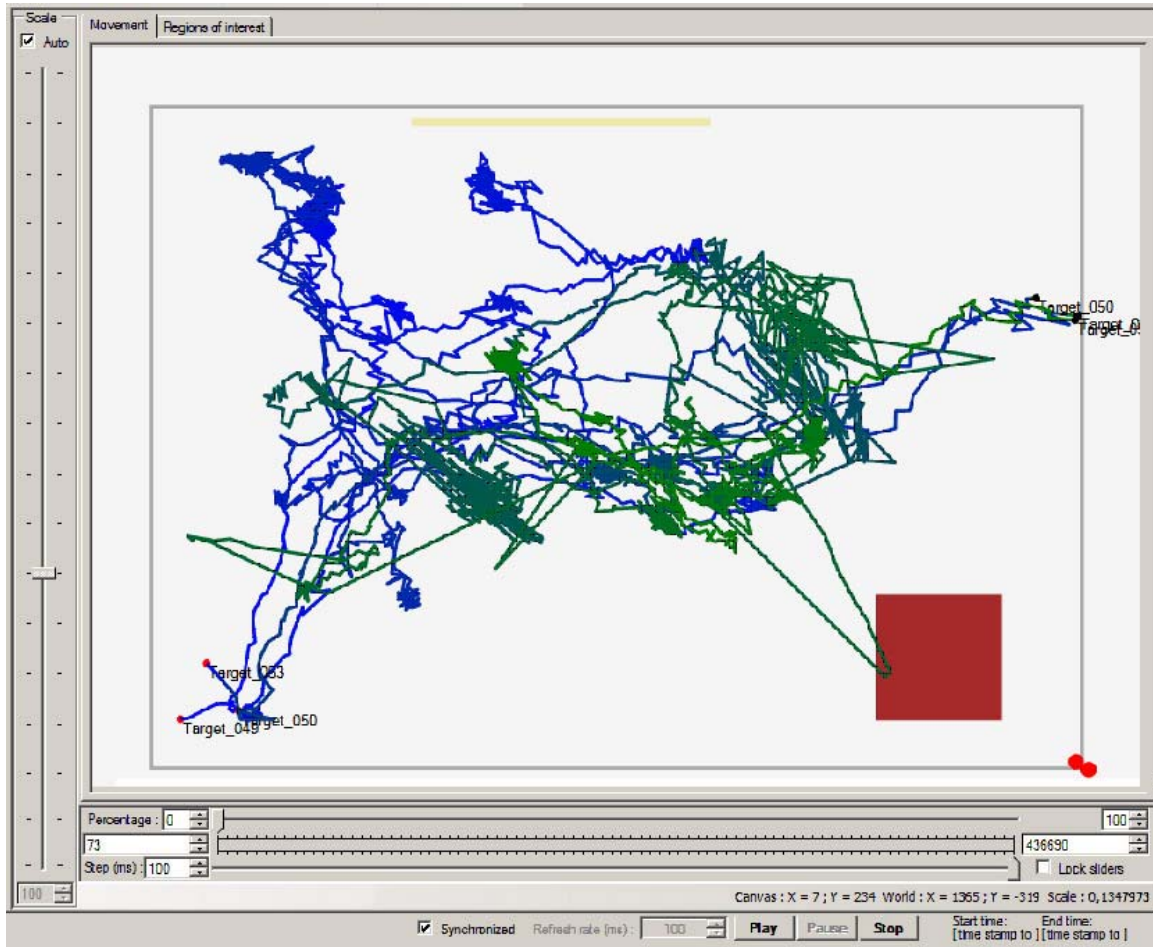


Figure 4: Visualization of history in the room.

properties of all trajectories investigated during one session. The information acquired has a form of tiled space with information where the average interest of all persons was concentrated. The various colors mark different levels of interest in the conference room under investigation (see Figure 5).

5.3 Usage and Validation of the Visualization

The main purpose of this tool is to analyze user cognitive load in temporal and spatial dimensions. For temporal dimension analysis we can select intervals where we study the user movement. We assume that a quickly moving user has low cognitive load and a slowly moving user has high cognitive load. This can be analyzed using the replay of the user movement where we can follow the development of trajectories throughout the time. For spatial dimension analysis we look for places where the user spent a large amount of time and we assume that user cognitive load is high at those places. For example spaces D6, D7, F4, F6 and G5 are considered to be places with high ROI and possibly also high cognitive load. Such information

can also be used for more detailed investigation of motion of individual persons in a particular part of an exhibition hall, gallery or museum.

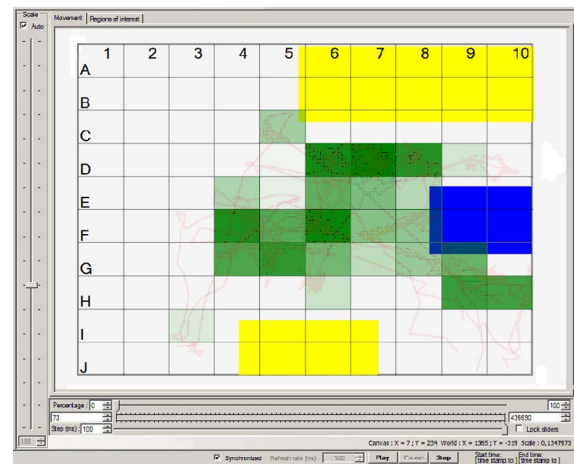


Figure 5: Levels of interest in the conference room.

6 USER DETAILED BEHAVIOR

This level of the user model is able to reveal user behavior and user interactions with other persons or with other objects. For example, we can describe the behavior of the user while following the lecture like, when s/he has high cognitive load and when the cognitive load has decreased. Observation of the cognitive load is not direct. We have to use the bridge between the behavior and the cognitive load. This bridge is the region of interest (ROI). The ROI shows what the current interests of the user are. The interest can be an object, a person or a particular activity. Estimation of the user ROI is a dynamic process, so it is changing in time. There are two classes of activities that influence the ROI. The first class contains activities like watching something or communicating with somebody, which increases the value of the ROI of the particular thing or person. The second class contains activities like rocking on the chair, sleeping, or throwing papers which decreases the value of the ROI of all things and persons. The cognitive load of the user is increasing when the ROI of the user is increasing.

6.1 Representation of User Model Data

Data about the user behavior are describing the user position with other information about her/his behavioral activity (e.g., laying, sitting or standing). There is also information about user direction s/he watches and information about additional user gestures, like hand, head and body movement.

The collection of all previously mentioned data is not trivial. While the position of the user is the easiest activity to detect, hand gesturing and body deformation detection and interpretation are very hard tasks. Detectors that collect data are of several types. They can collect data from one or more persons, they can collect data about one or more attributes and they can collect data with different precision. Generally, to obtain enough information about the user behavior it is necessary to use more detectors, which will produce more data in simultaneous streams, that need to be properly processed and synchronized before being analyzed.

Unfortunately, the situation in the detector industry does not allow us to detect all possible and necessary data for complete user behavior description. At the present time the detectors are not very precise and they are very expensive. Sometimes there is also the necessity of post processing of the data from detectors before being analyzed, because of low precision of parameter detection. To overcome this problem we developed a tool which is capable of generating data about user behavior based on created scenarios and it is capable of simulating output similar to different detector outputs. In this way we are

able to create data based for example, on video recording or even create artificial scenarios simulating a real situation. The focus is paid on editing functions like creation of user movements, control of user's actions and synchronization of interactions between multiple users. The tool stores the data in generic XML format, from which different detector's data can be generated for the purpose of simulation. Currently, we are able to generate position, state, body and head orientation of the user, user activities like speaking and user interaction with objects and other persons. Our future work is investigation on how to generate data for gestures and body movement.

6.2 Developed Tool

For analysis of the user detailed behavior model a new tool was developed. This tool visualizes the environment of the user in 3D using Java3D. The behavior of the user is visualized in a form of synchronous playback of all the data recorded by the detectors (see Figure 6). The recorded data are interpreted and visualized in a form of avatar behavior. The avatar is a virtual representation of the user in the artificial 3D environment. The visualization displays how the user and eventually other persons are behaving in the 3D environment. It is possible to filter the detectors inputs and for example focus on specific moments in the recording. It is also possible to control the data visualization, such as, to pause the visualization and navigate in the visualization.

Visualization of a ROI, which is presented in Figure 6, highlights the selected user by black color. Other objects or users in the scene (highlighted by black arrows), that are part of the selected user ROI are altered by means of their color which represents the ROI value. The green color means low ROI and the red color means high ROI. The cognitive load of the user is indicated by the yellow strip above the user avatar. The length of the strip corresponds to the user cognitive load.

6.3 Usage and Validation of the Visualization

The main goal of this visualization is to show relations between users in term of the ROI that are stored in the user data model. We can immediately see the cognitive load of the user and amount of interest in other objects in the environment. The data are more anonymous than video recording and we can immediately check their correctness (mainly continuity). In such a way we can also compare user behavior with desired behavior and analyze differences. The 3D visualization is part of a simulation and visualization tool SitCom – IBM Situation Composer for Smart Environments (Fleury, Curin, and Kleindienst 2007). This tool is used for analysis and on-line visualization of user behavior in smart spaces.

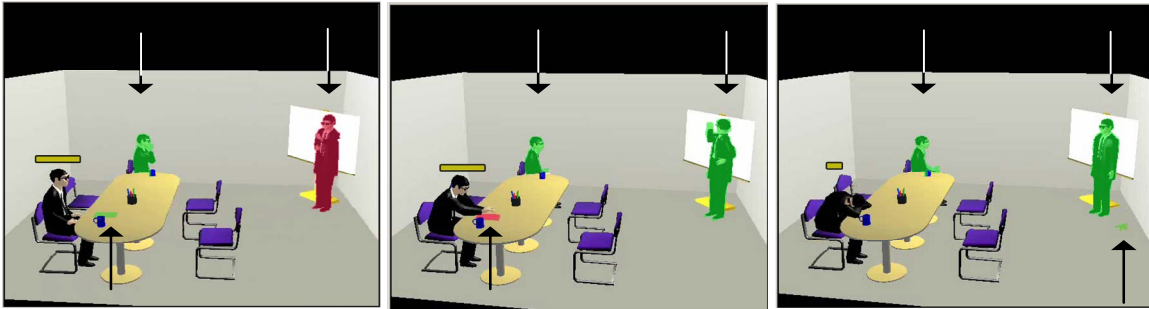


Figure 6: Demonstration of ROI visualization in user detailed behavior model.

7 SEMANTIC DRIVEN GLOBAL COGNITIVE LOAD AND ROI COMPUTATION

In previous chapters we have described methods for investigation of each of three levels of user activities in a separate way, without taking into account any influence among them. It is obvious that all three levels of activities are tightly related and that the interpretation of activities (like region-of-interest or level of cognitive load) gathered from one level only can lead to incorrect conclusion concerning the global user model (see Figure 7).

For example we have mentioned that when the user stops it can indicate that something interesting appeared in her/his neighborhood and we can conclude that this is the region of interest (ROI) of the user and that her/his cognitive load is increasing. If we look at the activities of the second level (user's detail behavior), we can for example observe, that the user is sitting in the chair and smoking a cigar. This activity will indicate, that the user is relaxing and not observing anything of her/his interest. We will estimate that the level of cognitive load is decreasing.

Another example can be interpretation of the user's motion slow down. This again can be an indicator of approaching a ROI, but if we analyze the third level of the user's activities (the interaction with a program) we can indicate for example reminder pop-up on the user's PDA which took the user's attention and made her/him slow down because of the cognitive load increase. Moreover the user can, depending on the kind of interaction with the PDA, stop moving at all (which in analysis of the first level of activities indicates a ROI) or can start unpredictable movement in a relatively small spot (in the case of a phone call for instance). This can lead to a wrong indication of the ROI and false identification of factors affecting the user's cognitive load.

We can see that to increase the accuracy of the interpretation of the user's activities, all three levels of user activities should be analyzed as a whole.

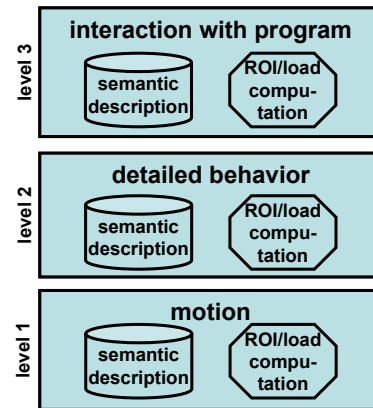


Figure 7: Interpretation of user's activities for each activity level separately.

7.1 Global Cognitive Load Computation

The computation of global cognitive load which will be based on all three levels of activities can be solved in several ways.

The simplest way is to define sum function, which will sum up the particular loads computed by algorithms defined for each activity level. This sum function can be refined by assigning weights to each particular load.

The second more complex way is to go one step deeper into the activities and to specify the behavior patterns across all activity levels. The cognitive load will then be assigned to these behavior patterns.

The third way is to utilize already existing algorithms for ROI computation on a particular activity level and to define algorithms for computation of a global ROI for all three activity levels and compute the cognitive load from this global ROI.

7.2 Global Region of Interest Computation

In previous chapters we have visualized the ROI for each activity level separately. To avoid problems described above we need to compute the global ROI on the basis of all three levels. This can be done by combination of ROI algorithms defined for each activity level.

These algorithms for combination of particular ROIs will be based on analysis of the semantic description common for all three levels of user activities. This common semantic description should be done in a formal way by means of an ontology language like OWL (OWL 2007). This is necessary for efficient implementation of semantic ROI algorithms (Mikovec et al. 2005).

The ROI values of particular ROI algorithms will be simply summed and normalized.

The sum function can be refined by assigning weights to each ROI algorithm based on the user activity analysis at all levels. Before the sum function is applied each result of a particular ROI algorithm will be multiplied by a coefficient (real number), which will increase or decrease the computed ROI. This weight coefficient represents the influence of the user's activity coming from a different activity level on a particular ROI algorithm. This can be implemented as rules, which define how the particular ROI algorithms will be influenced by user activity.

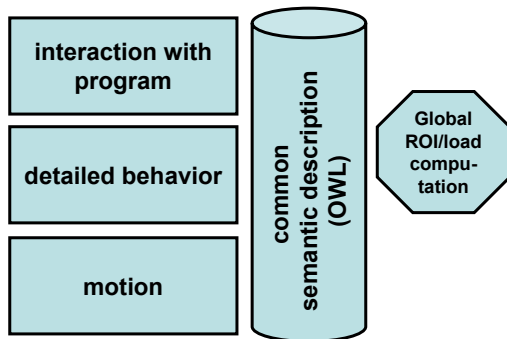


Figure 8: Global ROI and cognitive load computation based on all user activities.

An example can be the influence of the ROI algorithm for computing the ROI based on user motion speed by activity coming from the detailed behavior layer. The ROI value will be increased if the user is at the same time watching surrounding objects. If at the same time the user will be communicating with another person, the ROI will decrease.

The ROI value computed by the particular ROI algorithm can be decreased to zero level. An example can be the above described problem with user's motion slow down during exhibition visit, caused by received SMS instead of spotting interesting exhibit.

7.3 Visualization of Cognitive Load and Region of Interest

In previous sections we have introduced algorithms for computation of cognitive load and the ROI based on analysis of users activities of all three levels. Now we will present our approach to the visualization of these parameters to enable efficient creation of a user's model.

The first approach utilizes already existing solutions to particular activity levels (described in chapters above). The idea is to substitute their algorithms for cognitive load or ROI computation with the new ones (see previous sections). The user can than switch among three views on the user's activities (represented by three visualizations methods given above), but the cognitive load and ROI is computed with respect to all activities.

The second approach is to design a new visualization method which will introduce a cumulative view on the user's activities from all levels and at the same time allow one to zoom in to a particular activity level and analyze the user's activity in more detail. For the detail analysis of a particular activity the already existing solutions will be used.

The advantage of the first approach is the usage of tested visualization methods. The disadvantage can be worse orientation of the expert when switching between the views of user's activities. This should be resolved by the second approach, which on the other hand has not been implemented and tested yet.

8 CONCLUSION

We presented and described three levels of user activities, which allow the designer to analyze user behavior from different points of view. These levels form the user model and they can be treated as levels of detail of the user model.

We also presented tools for analysis of user models for the purpose of UI development. The tools work independently at each activity level and allow investigation of the user cognitive load, which is an important indicator for optimization of UI design. These three levels cover interaction with the mobile device, detailed behavior of the user and analysis of the history of user motion. We showed what data are used by each tool, how they are visualized and what information they can provide.

Success of the user analysis depends on sufficient amount and accuracy of the collected data of user activities. In the last chapter we presented a method on how to increase the accuracy of user model analysis by introducing methods for global cognitive load computation and region of interest (ROI) computation.

Our future work is investigation and evaluation of the global cognitive load and ROI computation and visualization. The whole tool will be used for complex user behavior usability studies. Currently, the user model is not extendable. We will investigate if it is possible to extend it or update it dynamically.

ACKNOWLEDGMENTS

This work has been sponsored by IBM within the frame of IBM-CTU Student research projects. It has also been

partly supported by the Ministry of Education, Youth and Sports of the Czech Republic under the research program LC-06008 (Center for Computer Graphics) and the research program 6840770014 (Research in the Area of the Prospective Information and Navigation Technologies).

REFERENCES

- Fischer, G. 2001. User modeling in human-computer interaction. *User Modeling and User-Adapted Interaction*, 11(1/2): 65–86.
- Fleury, P., J. Curin, and J. Kleindienst. 2007. SitCom- development platform for multimodal perceptual services. In *Proceedings of 3rd International Conference on Industrial Applications of Holonic and Multi-Agent Systems*. Regensburg, Germany.
- Harrison, B. L., R. Owen, and R. M. Baecker. 1994. Timelines: an interactive system for the collection of visualization of temporal data. In *Proceedings of Graphics Interface'94*, 141-148. Toronto: Canadian Info. Proc. Society.
- Holm, R., M. Priglinger, E. Stauder, J. Volkert, and R. Wagner. 2002. Automatic data acquisition and visualization for usability evaluation of virtual reality systems. In *Proceedings of Eurographics Short Presentations*, Saarbrücken, Germany.
- Luyten, K., C. Vandervelpen, and K. Coninx. 2005. Task modeling for ambient intelligent environments: design support for situated task executions. In *Proceedings of the 4th international Workshop on Task Models and Diagrams*, 127:87-94. New York: ACM Press.
- Maly I. and P. Slavik. 2007. Towards visual analysis of usability test logs using task models. In *Proceedings of 5th International Workshop on Task Models and Diagrams for Users Interface Design*, 16-23. Hasselt, Belgium: Springer.
- Mikovec Z., J. Jelinek, P. Guilcher, and P. Slavik. 2005. Domain ontology driven adaptation of graphical information. In *Proceedings of 11th International Conference on Human-Computer Interaction: The Management of Information - E-Business, The Web, and Mobile Computing (Volume 2)*. New Jersey: Lawrence Erlbaum Associates.
- Mori, G., F. Paterno, and C. Santoro. 2002. CTTE: support for developing and analyzing task models for interactive system design. *IEEE Transactions on Software Engineering* 28(8): 797-813.
- Nielsen, J. 1994. *Usability engineering*. Morgan Kaufmann.
- Nishikawa H., S. Yamamoto, M. Tamai, K. Nishigaki, T. Kitani, N. Shibata, K. Yasumoto, and M. Ito. 2006. Ubireal: realistic smartspace simulator for systematic testing. In *UbiComp, Lecture Notes in Computer Science* 4206:459–476. Springer.
- Noldus. 2007. Noldus observer. Available via www.noldus.com.
- OWL. 2007. Web ontology language (OWL). Available via www.w3.org/2004/OWL/, W3C.
- Patterson D., L. Liao, D. Fox, and H. Kautz. 2003. Inferring high-level behavior from low-level sensors. In *Proceedings of UbiComp*.
- Tazari, M. R., M. Grimm, and M. Finke. 2003. Modelling user context. In *Proceedings of 10th International Conference on Human-Computer Interaction*. New Jersey: Lawrence Erlbaum Associates.

AUTHOR BIOGRAPHIES

ZDENEK MIKOVEC is an assistant professor at the Dept. of Computer Science at Czech Technical University in Prague, Czech Republic. He is a member of several international research projects. He focuses on UI design in special environments and more specifically, he is investigating the interaction of mobile and visually impaired users with graphical information. His e-mail address is xmikovec@fel.cvut.cz.

IVO MALY is a PhD student at the Dept. of Computer Science at Czech Technical University in Prague, Czech Republic. His research is focused on techniques of UI design, more specifically he studies visualization of data from user interface usage and user interface tests in connection with UI models and application structure. His e-mail address is malyil@fel.cvut.cz.

PAVEL SLAVIK is a Full Professor of Computer Science at Czech Technical University in Prague, Czech Republic. His professional interests cover user interfaces and computer graphics (especially scientific visualization). He works as a member of the Computer Graphics Group at the Department of Computer Science and Engineering. He is a member of Eurographics and ACM. Every year he is a member of several program committees of conferences concerning user interfaces or computer graphics. His e-mail address is slavik@fel.cvut.cz.

JAN CURIN got his Ph.D. in Computational Linguistics from the Faculty of Mathematics and Physics, Charles University in Prague, in 2006. His research interest is in the field of human language technologies, human-computer interaction, and multimodal interfaces. He is currently in a research position at IBM, Czech Republic. His e-mail address is jan_curin@cz.ibm.com.