Virtual prototypes in usability testing

Kari Kuutti¹, Katja Battarbee², Simo Säde², Tuuli Mattelmäki², Turkka Keinonen², Topias Teirikko³, Anne-Mari Tornberg⁴

¹University of Oulu, Department of Information Processing Science
   E-mail: kari.kuutti@oulu.fi

²University of Art and Design Helsinki, Smart Products Research Group
   E-mail: katja.battarbee@uiah.fi

³University of Lapland, Department of Industrial Design
   E-mail: tteirikk@levi.urova.fi

⁴Umeå University Designhögskolan, Interaction Design
   E-mail: anipoi99@student.umu.se

Abstract
This paper describes a study of a three-dimensional virtual prototype intended for usability testing and concept validation over the internet. A virtual reality modeling language (VRML)-model of a fictive mobile phone concept with simulated interaction functionality was tested for usability following two approaches. The first group performed a traditional task-based usability test, the other performed a free exploration test. The approaches were compared e.g. by asking the participants to draw their mental model of the product's user interface.

The research aimed at supporting de-centralised product development. The goal was to expand the possibilities of user-centred design methods by utilising the internet. The results indicate that 3D virtual reality prototypes can be used for remote usability testing and design evaluation. The limitations and requirements for successful prototyping and testing are discussed. The task-based and free exploration testing approaches are also compared. Moreover, broader issues on usability testing using virtual reality (VR) models compared to physical models are discussed. For example how users experienced the product from the VR model compared to a physical appearance model.

1. Introduction

The study described in this paper is part of VIRPI, a larger research project on virtual prototyping. The project focuses on the area of mobile telecommunications and Internet based electronic commerce services. It belongs to the national programme to improve the efficiency of new product development, co-funded by the Technical Development Centre of Finland (TEKES), industries, universities and research institutes in Finland. From the start of the VIRPI-project there has been a goal to evaluate the possibilities to use a virtual model in usability testing. The motivation for this is obvious: if it is possible to test usability based on a virtual model instead of a real prototype, it may be possible to push testing earlier into the design process, where it is easier and cheaper to correct any potential errors. Initially the aim was to use an available high-end VRP environment with 3D glasses and the haptic interface for testing. During the first year, the project gained a lot of experience of what could be done with such a high-end environment, and the interest of the project and the partner companies turned more and more towards low-end solutions distributed over the Internet. Although such low-end prototypes lose some important features with respect to the realism of the models, it was felt, that the potential benefits of the ability to distribute and test virtual prototypes easily over networks was more important. This led to a new orientation for this part of the project. A high-end prototype at only one place can be tested using similar methods as in normal usability testing. Data collection, for example, is naturally facilitated by a person who runs the tests. This is not possible in remote testing as a log system is needed to collect the information. This was problematic because such a log system did not exist, and building one would have been a new major task not taken into account in the project plan or resource allocation. A decision was made not to do real remote testing, but only simulate it to collect initial information to support future log system building. The simulation took place in a series of testing experiments explained more accurately in the subsequent sections.

Normally, usability testing is done in situations where both testers and users are at the same time in the same place. Users perform given tasks with the system, and testers observe their behaviour to find problems in the user interface. Often, the test situation is videotaped to help later analysing of the session. In remote usability testing, the persons running a test are distanced, spatially
and/or temporally, from the users of the system to be tested [1].

The internationalisation of markets and companies has created a need to evaluate and test products during the design process in a geographically wide area with users belonging to different cultures. Products may have several different potential user groups, or companies have product development in several places. At the same time the development of networks has opened possibilities for distributed activity and also for remote usability testing. It is possible, e.g., to use shared desktop applications, video conferences, questionnaires located on the Internet to collect subjective experiences of users, for example, so that performing some function automatically opens a reply screen. Many remote testing settings rely on recording or transmitting both video and audio of the test situation [2]. Another possibility would be to collect data automatically from actual interactions with the product. Users have also been trained to recognise usability problems and to report them to product developers. For example, when developing a method called as the “user-reported critical incident” [1], developers had made a special button in the application under development and testing so that users could easily generate a problem report and send that to developers whenever problems were encountered. In addition to that the system recorded screen events and sent with the report a clip of the screen action. Another approach [3] used intelligent agents to monitor users actions and compare them to the developers’ expected use sequences. When a mismatch arised, the system allowed the user to communicate with the developers synchronously or unsynchronously.

Castillo et al. [1] list the following issues that are different between a remote and traditional testing situations and those that must be considered.

- Is the product used in a real-life setting or is the setting artificial?
- Timing of users’ activity related to testers’ activity.
- Location of users’ activity related to testers’ activity.
- Who identifies problems: user, tester or both?
- Amount of interaction between testers and users.
- What is the form of collected data (video, audio, field notes, a log file, problem reports, answers to questionnaires)?
- What is the technology necessary for data collection?
- What are the costs of data collection? Costs of analysing it?
- Usefulness and quality of collected data.

This list is based on the mainstream usability testing tradition, where the main emphasis has been in testing computer programs. Because of the goals of the project VIRPI, this must be expanded to contain issues related to embedded systems or smart products, which contain both the physical and software-based user interfaces. For their evaluation the following issues have to be considered:

- What are the dimensions the model to be tested shows to the user -- physicality, 3-dimensionality, interactivity?

How many issues is it possible to evaluate during one session? Is the result a holistic evaluation of the product, or are only the usability aspects evaluated?

How does the lack of the real use environment influence the results?

2. Experiments

The usability tests of virtual prototypes were conducted in cooperation with University of Arts and Design in Helsinki, University of Oulu, University of Lapland and the CCC/Cybelius company. Because of technical reasons, the tests were not real remote tests, but the remoteness was simulated by the test organisation. The collection of use log data was replaced by videotaping the use sessions. Videotaping collects much richer data than a user log file. However, this was considered necessary, because there was no advance information what were the features of the model and use actions that should be recorded into the log file. By making a richer record, the research group was able to identify the important issues as requirements for a future log file recording system. In addition to usability testing, it was assessed how well the virtual prototype could be used to evaluate the design of the product and its features.

2.1 Prototype

![Figure 1. "Bugsy"-phone, physical mock-up](image)

A small-sized "Bugsy"-phone – a concept design made as a design exercise at the University of Lapland – was selected as the product to be tested. A physical appearance model was built (Figure 1). Also, A 3D CAD model of the phone was made by using the Alias modelling program (Figure 2). The user interface of the phone, a www-testing environment and on-line help for it were
specified by the University of Arts and Design. The Cybelius Authorizer-program was used to change the Alias VRML-model into a virtual prototype, which was then connected to the www-test environment developed by the University of Oulu (Fig 5.2). The resulting system enabled free movement, scaling and rotation of the model in a browser window (Cosmo Player) and the functionality of the phone could be used by clicking corresponding parts of the model using a mouse.

Figure 2. VRML model of "Bugsy"-phone

Information about the user’s operations with the prototype on the web page could be gathered from three sources: the log of manipulation of the VRML model, the log of action on the web page, and user’s comments or answers to questions on the web page.

2.1.1 Manipulation of the model. A log of the user’s operations with a virtual model could be recorded automatically. Depending on the aims of the evaluation, the log could be analysed in a number of ways. If the users had been provided with explicit tasks, the sequences revealed by the log could be compared to the optimal task sequences. Any systematic deviations from the optimal sequence could indicate usability defects. Though the overall time of performance is likely not to be a relevant criteria in decentralised use where the rate of interaction may be biased by many factors, systematic delays in the subject’s sequences can be interpreted as possible usability problems during the particular phases of interaction. If the task flow is interrupted at the same phase repeatedly, a usability defect can again be expected. Finally, the log could reveal whether the given task as a whole had been completed successfully.

If subjects were allowed to explore a virtual model without explicit tasks, other kinds of conclusions could be made. Some functions of the device may get more attention from some users than from others. These may have been more interesting from some users point of view. On the other hand, the neglected features may have not been made apparent enough for the users to find them. Whatever the reason for not exploring some specific functions may be, the information is important for product development. The reasons may have to be obtained with additional studies.

2.1.2 Action on the web page. The www environment of product presentation provides possibilities to show rich product related information to the user. Out of the several possibilities online help and online tailoring are among the most interesting. Sound, animation, context sensitivity, good quality display, and use of multiple windows enhance the possibilities.

Online help could be used diagnostically for product evaluation. In a similar way, as the sequences with the product are recorded, the usage of the online help could be recorded and the two synchronised. The reasons for usability defects could be determined from the way the subjects search for help. Even though the user’s help-seeking-behaviour may not be effective or appropriate, it would reveal the expectation concerning the reasons for failure, which, as such, is valuable information for the designers.

2.1.3 User’s subjective feedback. Recording the use and help logs is aimed at evaluating usability from the user’s behaviour. In a traditional usability test, the user’s simultaneous verbal protocol is an important source of analytical qualitative information. One of the most obvious defects of decentralised evaluation is the missing of this direct information. However, several possibilities to deal with the problem can be considered.

First, the contact between the user and the evaluator can be established by video phone. The evaluator would be able to see and hear the subject. An approximately similar situation could be achieved with voice telephone contact, if the user’s and evaluator’s virtual models could be synchronised. These solutions may enable relatively natural communication between the user and the evaluator, at least for people who are familiar and used to the still quite unusual technical environment.

The solutions above require that the test evaluator is working simultaneously with the user – though at a different location. To increase the cost efficiency of the evaluation by asynchronisation, solutions for subjective user response gathering are needed. The user should be able to express feelings about the interaction in general and point out specific notices during the interaction, preferably in a way that they are synchronised with the log data. Simultaneous verbal protocols, i.e., thinking aloud, is not considered to disturb too seriously the primary task. The same should be the aim in developing the online usability feedback tool. Speech recognition might be the solution in the future. At the moment, a feedback window where the user can write short comments could be visible at all times and connected to the use log. Also, on-line questionnaires could be placed on the web page to gather user feedback after the test session.
2.2 Tests

The first testing simulated traditional usability testing. The testers were in the same room as the users. They introduced users to the environment and then gave tasks to them, one by one. The users had altogether five tasks to perform. They were asked to think aloud when doing the tasks, and the sessions were videotaped. The tasks were relatively simple and straightforward:

1. Turn the phone on.
2. Make a call to “Anna”.
3. Change the alarm melody into number 2.
4. Read the last text message.
5. Close the phone.

After performing the tasks, paper and pen were given to users and they were asked to draw a picture of the phone and to describe its functionality. This was also videotaped. Finally, a questionnaire was filled and the physical mock-up model was shown to the user at this last phase. Three users participated in this phase.

Based on the experiences from the first testing, two new tests were planned and performed, one for task-based and one for free exploration. Their aim was to simulate the remote usability testing situation better. The first test resembled closely the previous, traditional testing. The only difference was that the users had to do the tasks without any external help (Figure 3).

Figure 3. Usability testing, unguided user session.

A training phase to the Cosmo Player environment was also added. There, the model to be explored was a model of a current Nokia mobile phone. Users had the possibility to explore the environment until they felt confident. Then, tasks were given to them on paper and, after that, the test proceeded like the first type, only without a tester being present in the room. Four users did this test, none of them being the same as those who took part in the previous test.

According to the list of issues given above, based on Castillo et al. (1993), the modelling and evaluation method in the VIRPI project was as follows:

- Product is not in real use, the users may be both real and/or representative.
- Time of evaluation is asynchronous.
- Users are located in their own environment.
- Problems are identified by both the user and the evaluator.
- Tasks may include ones predefined by the evaluator or the user’s own tasks.
- There is no direct interaction between the users and evaluators.
- Data is collected by data logging, user’s comments and surveys.
- Data collection requires computers, net connection, interactive VRML model, www site, and software for log gathering and analysis.
- Once the technology is in hand, the data collection is relatively inexpensive, but the cost to analyse the data may be high.
- The usefulness of data suffers from the need to analyse the log and from the lack of interaction between the user and the evaluator, when compared with traditional laboratory testing.
- The usefulness of data suffers from the need to analyse the log and from the lack of interaction between the user and the evaluator, when compared with traditional laboratory testing.
- The model represents the interaction and the three-dimensionality, but misses the physical and haptic attributes.
- Evaluation takes place at the computer, so for most products the environment and context is not real.
- In addition to usability evaluation, it is possible to evaluate the quality of design, though in a limited manner only.

Free exploration was tested by giving first a similar Cosmo Player training session as in previous test and then the users were given a free exploration scenario (“check, how the phone works”). After that, the test proceeded like the previous one. Four users also did this test, none of which had participated in any previous tests.

3. Findings

3.1 General

A clear result of the experiments was that virtual prototypes can be indeed used in recognising usability problems, like problems in the logic of functioning, confusing positions of input/output (I/O) devices, etc. However, the use of virtual prototypes for testing is not straightforward, but it introduces a new set of concerns, that must be taken into account.

First, there is in every case a clear difference in the experienced realism between the virtual model and the real
thing. The worse the quality of the model, the more possible is it that the performance degrades from what the result would have been with a real thing. For example, during the tests, users, every now and then, tried to push “wings” of the Bugsy-phone that are part of the phone body, as if they were moving buttons. Such a mistake would have been very improbable with a real model. The conclusion was that the visual quality of the virtual model was not good enough to prevent this type of error. This lack of realism makes it also more difficult to evaluate any other product features for a virtual prototype.

Second, the usability of the testing environment itself is crucial. In our case the testing environment consisted of four levels: the www browser, the testing environment proper with the help and the comment windows, CosmoPlayer in one of the environment’s windows and finally, the model itself. So there were actually a set of three separate programs to be used around the prototype. These were never designed as a whole and there were obvious idiosyncrasies and usability problems with them. The major problem was the CosmoPlayer that was used to show and manipulate the model in the virtual place. CosmoPlayer was initially aimed to navigate in the virtual reality environment and its controls had been designed accordingly. They are not optimal for virtual prototype manipulation purposes. For example, in virtual reality, one can go very close to an object, so that it, for example, fills the whole field of visibility. When manipulating a hand-held product, this behaviour was unnatural as people do not usually take anything closer than where the eye can be focused, say 25 cm. It is, however, impossible to limit movements this way in CosmoPlayer. This led to situations where some of the testers did testing with a greatly enlarged model – so large that it did not fit into a screen anymore and some behavioural hints were lost because of that.

This was a test environment, put rapidly together with limited resources, so these kinds of problems were quite natural. The experience was that the testing environment should itself be designed as a whole, and rigorously tested for its usability in virtual prototype testing. If something like CosmoPlayer is used, it is necessary to train users to it by using some other model of preferably familiar product for training purposes. For example, in later tests, a model of a Nokia mobile phone was first used for training before the actual tests with the Bugsy-phone. This clearly reduced problems with the environment.

3.2. Comparison between different types of testing

In the following, the benefits and drawbacks of virtual models and remote testing compared with traditional models and usability testing are discussed. The importance of each benefit and defect depends on the situation and goals on hand.

<table>
<thead>
<tr>
<th>Table 1: The benefits of virtual models and remote evaluation:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Interactive 3D model</strong></td>
</tr>
<tr>
<td><strong>Photorealistic images</strong></td>
</tr>
<tr>
<td><strong>Remote use</strong></td>
</tr>
<tr>
<td><strong>Remote design evaluation</strong></td>
</tr>
<tr>
<td><strong>Large number test subjects</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2: The drawbacks of virtual models and remote evaluation:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No physical qualities</strong></td>
</tr>
<tr>
<td><strong>Limited capability to evaluate ergonomics and certain dimensions of usability</strong></td>
</tr>
<tr>
<td><strong>Indirect manipulation</strong></td>
</tr>
<tr>
<td><strong>Lack of interaction between the users and evaluators</strong></td>
</tr>
</tbody>
</table>
It is not a question of the qualities of a virtual model and a physical model. The main approach, nowadays, in prototyping smart products for usability and design evaluation is to represent the interaction with an on-screen two-dimensional software prototype and the physical product with static mock-up or appearance model. The comparison (Tables 1 and 2) is between:

- an asynchronous remote usability test with a 3-dimensional interactive on-screen prototype which can be operated with a mouse, and
- a traditional laboratory based usability test using a design mock-up and a 2-dimensional interactive on-screen prototype which can be operated with a mouse.

4. Further research and development

If and when virtual prototypes are used in remote usability testing, a log system to collect the information about the use is essential. This will be necessarily more complicated than a typical program collecting user actions with a GUI. At this moment, the differences include:

There are several programs in parallel use (browser + added functionality, CosmoPlayer and its navigation and manipulation tools, and the virtual prototype itself), and the log system must follow each of them.

It is not sufficient to record the use of CosmoPlayers controls without knowing why they are being used (what was the initial position of the virtual model). It may be necessary to make a complete record of all navigations and manipulations that can be rerun afterwards during the analysis.

In order to get meaningful data out from the use of the virtual prototype itself, it is necessary to be able to connect at least any click and perhaps also all movements to those parts of the model that are pointed at. This means that the log system must have access to the VRML model, and that “visible entities” can be separated even if they actually are part of the same logical entity (for example, in our example the log file should recognize when the “right wing”or “left wing” is clicked and make the separation, despite the fact that both are parts of the same body and they look like just pushable buttons.

This level of required granularity hints that the best way to collect data might be a full replayable record of a session. This has at least two drawbacks. First, the amount of data collected from each session will be large. Second, it is difficult to automate the interpretation or even to preprocess the data for interpretation, but each of the sessions must be replayed and watched by a human interpreter. Such a log system would be complex to build and relatively heavy to use. Further research and experimenting is necessary to find out the limits and potential more accurately.

The quality of the virtual model should be high enough to cause no more additional confusion. At least with small devices, like phones, the rendering quality of the image should be at the photographic level. Also the models themselves should be very accurate, taking into account, for example, thickness of materials, etc.

The testing environment is one source of problems that may degrade the value of the testing data. The version used in the test was a rapidly put-together research prototype, with many rough edges. For a serious test, the environment should be designed as a whole and subjected under rigorous usability tests itself to iron out the rough edges. CosmoPlayer is another source of unnecessary confusion with its different basic orientations to what would be actually needed. The best solution, although resource-wise difficult to achieve, would be to design a special virtual model manipulation tool for the purpose.

5. Summary

Virtual prototypes can be used productively in usability testing. The quality of the virtual model and the testing environment are crucial for success of the testing, and the testing situations must be carefully designed. Task-based remote testing seems to be suitable for testing the logics and other usability issues, while a free exploration seems to provide better feeling of the design and sharper impressions. A log system for remote usability testing is necessary. It will necessarily be quite complex to build and the interpretation of data collected will be difficult, in particular in a free exploration situation. Building of such a system from scratch is beyond the capabilities of a normal tester. A testing support system consisting of a log collector and tools for analysing the log files would be needed and this might be potential area for a new software product.

6. Acknowledgements

The research was conducted when the last author was working in University of Lapland, Department of Industrial Design.

7. References