

A Web-Based Interactive Virtual Environment for Mobile Phone Customization

Peijun Wang¹

e-mail: wangpi@hotmail.com

Robert Bjärnemo

e-mail: robert.bjarnemo@mkon.lth.se

Damien Motte

e-mail: damien.motte@mkon.lth.se

Division of Machine Design, Department of Design Sciences, Lund University, Lund Institute of Technology, P.O. Box 118, Lund, SE 221 00, Sweden

With globalization, more and more companies are becoming multisited. Efficient and timely collaboration and communication among the development team members become critical for product success. This paper proposes a web-based virtual environment for mobile phone customization, named VMPDS (Virtual Mobile Phone Design Space). The features of this system include the combination of web technology and VR (Virtual Reality) technology, the ability to deal with and extend the product model database, adequate interaction for three-dimensional product model customization, and usefulness for the examination and evaluation of product concept design. The system is implemented by using standard open technologies to assure accessibility and platform independence. [DOI: 10.1115/1.1818685]

Keywords: Virtual Reality, VRML, Concept Design, Mobile Phone

1 Introduction

In order to stay competitive and profitable, many mobile phone companies are becoming multisited, operating with partners and customers on a global market. The fact is that the increased competition is driving manufacturing away from mass production, where the manufacturer tells the customer what to buy, and toward mass customization, where the customer decides what the manufacturer will make [1]. In this case, effective collaboration and communication among designers and customers play a crucial role for the success of mobile phone products.

Many factors influence the success of mobile phone products, such as technological advance, product price, attractive design, rich functionality, marketing, etc. Customer preference is one of the most important factors. As a result, collecting and identifying customer needs and understanding their preferences are essential tasks in the development of new mobile phone products. However, it is demanding and challenging for designers to transfer customer-implicit preferences into product design specifications. Chuang et al. investigated perceptual factors underlying user preferences about the product form of mobile phones [2]. In this paper we propose a web-based interactive three-dimensional (3D) virtual mobile phone design environment, called VMPDS. It is

mainly based on the technologies of VRML (Virtual Reality Modeling Language) [3], Java, EAI (External Authoring Interface) [4], and the Internet. There have been some applications using these technologies [5–12]. However, most of them were real-time, walk-through simulations. Compared to these existing systems, the main innovations of VMPDS are: (i) the product is accurately modeled using CAD software, not just a simple, rough or conceptual representation; (ii) this system has scalability due to the combination with a product database; (iii) the dynamic I/O operation on VRML scenes is achieved; and (iv) VMPDS is a complete system, not only a 3D visualization. In VMPDS, users are involved in the product design activities and can submit their preferences on mobile phone style to the database, which will serve as the product design benchmark to help designers adopt a proper design for the intended end users. The network communication is accomplished by a browser/server structure using HTTP protocol. A Java-enabled web browser together with a VRML plug-in is the only requirement for the execution of VMPDS on an ordinary desktop PC.

2 Development Tools and System Architecture

One important criterion that influences the choice of development tools is that we want this application to use as low bandwidth and hardware sources as possible in order to be accessible even through the dial-up manner. To achieve this goal, three open standards—VRML, Java, and JavaScript—and EAI, are employed. VRML is used to create the 3D virtual environment. Both Java and JavaScript are used in the URL fields of Script nodes. Two approaches are used to connect VRML files with external Java programs, as shown in Fig. 1. One is SAI (Script Authoring Interface), which uses JavaScript to communicate with Script nodes of VRML files. Another is EAI, which is an application interface for Java. We use EAI as a channel to connect the VRML world with HTML pages through Java applets that control the VRML scene dynamically.

Without using special hardware, such as high performance graphic acceleration cards, VMPDS allows any user to enter it by visiting its homepage. There are four subsystems in VMPDS: user registration, virtual mobile phone design space, product library, and product specification.

3 User Interface

In order to be accessible for various users, some of whom have no significant computer expertise, an intuitive user interface is proposed in VMPDS. The I/O operations are performed through the Java applet buttons named *Clear*, *Save*, *Import*, and *Evaluating*, etc. In addition, as a virtual-reality application, VMPDS permits a virtual phone call by clicking the digital and functional keys. The LCD screen of the mobile phone model will display the input content together with sound response. Figure 2 shows the user interface.

3.1 Function Analysis of Control Panel. The control panel, on the left of the window, consists of two parts, giving the user access to the full functionality of the system. One is immersed in the VRML scene and accesses a collection of buttons. Another is some Java applet buttons and pull-down lists, as shown in Fig. 2. We divide mobile phone products into a few basic classes according to their essential appearance and characteristics. With regard to a certain class, users can choose various styles through selecting items from the pull-down lists to get different combinations of the cover, the LCD screen, the back, etc. The basic functionality of the control panel includes viewing from among predefined viewpoints, playing rotation animation or stopping animation at any time, resetting the scene to the default state and eliminating all the modifications, switching the product styles (Fig. 3 shows a snapshot of switching to another style when the cover3, LCD3, back1, and slot2 are selected), selecting colors of phone components, and customizing 3D design environment, etc.

¹Also at Southwest Jiaotong University, Department of Mechanical Engineering, Chengdu, 610031, China; e-mail: wpjd@263.net

Contributed by the Engineering Simulation and Visualization Committee for publication in the JOURNAL OF COMPUTING AND INFORMATION SCIENCE IN ENGINEERING. Manuscript received May 23, 2003; Revised September 8, 2004. Associate Editor: S. Szykman.

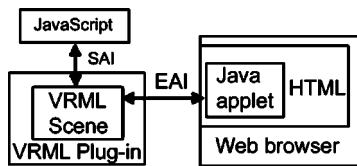


Fig. 1 Communication channels

3.2 Mobile Phone Model Visualization and Fine-Control Sliders. In order to make effective interactions and modifications, mobile phone models consist of ten parts: upper body (cover), back, digital keys (from 0 to 9), functional keys (including #, *, etc.), antenna, battery, brand slot, LCD screen, text, and lid. All of these parts can be modified individually, corresponding to user input. In addition, VMPDS allows users to detach and assemble the model by using the mouse to drag the cover. Figure 4 shows the upper body detached from the model body. In this way users can have a close look at the element and the interior structure separately.

Moreover, for every product style, users can customize their preferred appearance attributes quantitatively using four groups of control sliders on the right of the window. By dragging the sliders up and down, the colors can be changed. Users can play with the transparency, shininess, and ambient intensity to customize a desirable product. Every modification is displayed at the top of the corresponding slider in the form of data ranging from 0.00 to 1.00. Fine-control sliders provide users with more optimization capability in an intuitive and quantitative way.

3.3 Input/Output Operation. VMPDS features extensibility and scalability. The following three operations are designed for this purpose. After clicking the *Clear* button, the user removes the whole model from the scene. The *Import* button is linked to a HTML interface that is embedded with a Java applet connected to the model database. Furthermore, the user can save the custom-



Fig. 2 User interface



Fig. 3 Snapshot of switching to another style



Fig. 4 Snapshot of the detaching operation

ized model by clicking the *Save* button, which is also linked to the HTML interface. Because VRML does not provide functions of direct data input and output, it is difficult to save the modified 3D model in real time. In VMPDS, the task of input/output is performed with the help of Java applets and EAI. Figure 5 describes the saving interface. If the *OK* button is pressed, all the customized information will be sent to the database in the server. Otherwise, the clicking on the *CANCEL* button leads to returning to the 3D virtual environment for further modification. Of course, only the designers and selected users, who are carefully selected and given passwords for access to the database, have permission to work with the database. The user registration system is designed not only for security checks, but also for collecting user information.

In addition, VMPDS is developed for analyzing, evaluating, and communicating. As shown in Fig. 2, users can submit their evaluations by clicking the *Evaluating* button and jumping to the HTML file where they can input their comments and rate the product's conceptual design. After some analyzing and statistical processing, the company can have good insight into customer preferences. Moreover, by clicking the *Specification* button, users will access the product specifications where product introductions, prices, component materials and weights, sizes, technological features, as well as the company website are given in the forms of text and 2D images. Finally, we intend to provide an interactive tool for real-time video information through a "Meeting Room" to aid the communication between designers and users. This part of the work is underway.

4 Exchange and Optimization of 3D Model Data

There are two main methods for creating VRML 3D models. One is to directly use the geometry nodes defined inside VRML. The resultant file is very compact. But it is difficult to create realistic and complex models using only these primitive nodes. In



Fig. 5 Snapshot of the saving interface

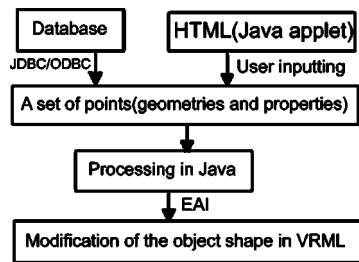


Fig. 6 Quantitative modification process

VMPDS the second method is used. The mobile phone model is created accurately using computer-aided design (CAD) software packages, then the CAD model is exported to VRML format. Designers and users can create their own models by using any kind of CAD software. We predefine three datum planes and alignment axes as well as size units. The back part of the mobile phone is selected as the base part, and other elements are integrated into this base. Following these rules will make new components fit VMPDS and achieve easy assembly. Considering the diversity of user software platforms, it is necessary to store the 3D model both in VR format and other compatible and neutral CAD formats, like STEP and IGES, to ensure quick, easy, reliable, and flexible sharing and exchanging of product data information among design teams.

In our case, the models are built using Pro/Engineer (Parametric Technology Corp.) and then exported to 3DS Max® (Autodesk Int.), where they are edited and optimized further. The converted VRML files from CAD software packages tend to be very big. The main reason is that VRML uses polygons instead of nonlinear free-formed surfaces, such as NURBS, to describe 3D objects. The more accurate the model, the more polygons have to be used. As a result, a compromise between realistic fidelity and rendering speed has to be made. Since the mobile phone is a consumer product, attractive appearance is crucial for the success of product design. Thus, a high-quality 3D model is necessary. On the other hand, VMPDS aims at being available on ordinary desktop PCs through the Internet without using special high-performance hardware. In this case, we optimize the 3D model through such methods as using some software to reduce the polygon number, simplifying less important parts, using as few texture maps as possible, compressing the VRML file, etc.

The possibility of changing the geometric shapes dynamically and quantitatively depends on the complexity of the object. For simple geometries, like box, cone, sphere, etc., and their combinations, the interactive modification is easy. For geometries that can be described using some mathematical equations and other approaches, the quantitative changing is also possible. Figure 6 explains the quantitative modification process. But for complex geometries that are converted from CAD software, the quantitative modification is difficult and time consuming. The first reason is that the converted object in VRML consists of huge numbers of

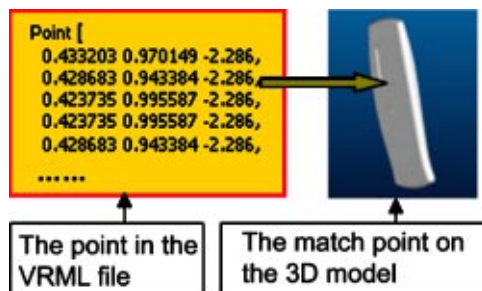


Fig. 7 Match relation between the points

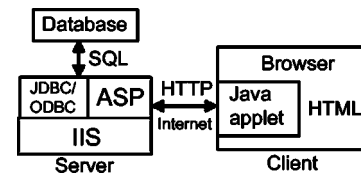


Fig. 8 Network and database structure

points inside a set of IndexedFaceSet nodes, and the spatial logical relations among these points are nonlinear. Another reason is that, as shown in Fig. 7, it is difficult to find the match between the point on the 3D model and its corresponding point in the VRML file. As a result, every basic element in VMPDS, such as the back, the LCD screen, the cover, etc., is considered as the minimum unit for modification.

5 Network and Database Structure

In VMPDS, a browser/server architecture and the standard HTTP protocol are adopted, as shown in Fig. 8. Although such a structure could be a bottleneck for networks with low bandwidth, we intend to use this solution because it is a comparatively simple approach and because the whole size of VMPDS is controlled as little as possible.

In the product database, models are arranged according to their specialties. Two product libraries exist in VMPDS. One is the basic product library maintained by designers and accessible only for designers and lead users. Another is the public product library that is a subset of the basic product library and accessible for any user. The second library stores the models that will be evaluated and customized by customers.

6 Conclusions

This paper presents a web-based interactive virtual mobile phone design system called VMPDS. VMPDS provides intuitive and effective means for designers and unsophisticated users to manipulate, modify, evaluate, comment on mobile phone product conceptual designs, and communicate with each other. VMPDS can be applied in the earlier and more creative phase of the design process rather than a walkthrough of final design, offering a potential for large cost savings. Moreover, using VMPDS to identify customer needs has obvious advantages over traditional interviews or other media. With VMPDS, companies can have more information to go on when developing new products, and the design team can get optimal design solutions from a large number of user-customized alternatives. We believe that in the future customers could order their own mobile phones through such a system as VMPDS, just like the practice in the automobile industry today.

VMPDS is implemented and tested on Windows NT4.0, Windows XP, and Windows 98 using IE 6.0 and IE 5.5 as web browsers. The VRML plug-ins used are Cortona 4.0 (Parallel Graphics Ltd.) and CosmoPlayer 2.0 (Silicon Graphics Inc.). The techniques described here can be applied to other web-based virtual product design.

References

- [1] Kuljis, J., and Paul, R. J., 2000, "A Review of Web Based Simulation: Whither We Wander?" *Proc. of 2000 Winter Simulation Conference*, Orlando, IEEE, New York, Vol. 2, pp. 1872–1881.
- [2] Chuang, M. C., Chang, C. C., and Hsu, S. H., 2001, "Perceptual Factors Underlying User Preferences Toward Product Form of Mobile Phones," *J. Indust. Ergonom.*, **27**, pp. 247–258.
- [3] VRML Consortium, 1997, "The Virtual Reality Modeling Language: International Standard ISO/IEC DIS 14772-1."

- [4] Silicon Graphics, Inc., The External Authoring Interface (EAI), <http://cosmosoftware.com/developer/eai.html>
- [5] Flake, S., Geiger, C., Mueller, W., Ruf, J., Paelke, V., and Rosenbach, W., 2001, "Customer-Oriented Systems Design Through Virtual Prototypes," *Proc. of 10th IEEE Workshop on Enabling Technologies*, Cambridge, MA, IEEE, New York, pp. 263–268.
- [6] Dos Santos, C. L. N., Cunha, G. G., and Landau, L., 2001, "Use of VRML in Collaborative Simulations for the Petroleum Industry," *Proc. of 34th Annual of Simulation Symposium*, Seattle IEEE, New York, pp. 319–324.
- [7] Durstewitz, M., Kiefner, B., Kueke, R., Putkonen, H., Repo, P., and Tuikka, T., 2002, "Virtual Collaboration Environment for Aircraft Design," *Proc. of 6th International Conference on Information Visualization*, London, IEEE, New York, pp. 502–507.
- [8] Campbell, D. A., 2000, "Architectural Construction Documents on the Web: VRML as a Case Study," *J. Autom. Construct.*, **9**, pp. 129–138.
- [9] Duffy, V. G., Wu, F. F., and Ng, P. P. W., 2003, "Development of an Internet Virtual Layout System for Improving Workplace Safety," *J. Comput. Indust.*, **50**, pp. 207–230.
- [10] Rojdestvenski, I., and Cottam, M., 2002, "Visualizing Metabolic Networks in VRML," *Proc. of 6th International Conference on Information Visualization*, London, IEEE, New York, pp. 175–180.
- [11] Ramos, J. J. G., Maeta, S. M., Bergerman, M., Bueno, S. S., Mirisola, L. G. B., and Bruciapaglia, A., 1999, "Development of a VRML/Java Unmanned Airship Simulating Environment," *Proc. of International Conference on Intelligent Robots and Systems*, Kyongju, Korea, IEEE/RSJ, New York, Vol. 3, pp. 1354–1359.
- [12] Buffa, M., and Lafon, J.-C., 2000, "3D Virtual Warehouse on the WEB," *Proc. of International Conference on Information Visualization*, London, IEEE, New York, pp. 479–484.