Islamic University of Gaza

Deanship of Graduate Studies

Faculty of Information Technology

Information Technology Program

Web Cube: A New Model for 3-D Web Browsing Based on Hand Gesture Interaction

By:

Raed S. Rasheed

120091248

Supervisor:

Dr. Ashraf Alattar

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Technology

February, 2012

Acknowledgment

First and foremost I thank Allah for everything.

I must also thank my family for their continuous support and patience throughout my project.

I also thank Mr. Bader Ajour, Mr. Mohammed Abu Quwik, Mr. Hazem Abu Quwik, and Mr. Mohammed Fannana for their valuable technical help.

Thanks must also go to my colleagues at the college of information technology for the many inspiring and fruitful discussions and suggestions.

Finally I should extend special thanks to my supervisor Dr. Ashraf Alattar for his patience, time, ideas and guidance.

Abstract

3-D web browsing is a promising trend for interaction with web content. However it is still illusive between virtual reality applications on the one side, and conventional web browsing on the other. In this research we propose a new model for 3-D web browsing that capitalizes on features of virtual reality technology with those of conventional browsing in order to provide an enhanced interactive user experience with web content. The new model is based on representing information content elements in 3-D perspective and organizing them inside a 3-D container that we call a "Web Cube" for 3-D web browsing. Furthermore, the model defines appropriate interaction mechanisms based on hand gestures. The model has been evaluated using an experimental technique to evaluate its efficiency, and a questionnaire to evaluate user satisfaction.

Keywords: 3-D Web, Web browsing, Virtual reality, Hand gestures, Hand tracking, HCI, Interaction evaluation.

عنوان البحث

وب كيوب: طريقة جديدة لتصفح الانترنت باستخدام التفاعل باليد

الملخص

تصفح الوب ثلاثي الأبعاد يعد بمستقبل أفضل في تحسين تفاعل المستخدم مع مواقع الوب. إلا أنه لم تتبلور بعد أطر أو نماذج محددة لتصفح الوب ثلاثي الأبعاد، فلا تزال تطبيقاته المتاحة تتراوح بين إما أسلوب الواقع الافتراضي، أو أسلوب التصفح التقليدي دون الجمع بين الأسلوبين بشكل أو آخر. في هذا البحث نقترح نموذجاً جديداً لتصفح الوب ثلاثي الأبعاد يهدف للاستفادة من مزايا الواقع الافتراضي ومزايا التصفح التقليدي لتوفير تجربة متطورة للمستخدم في التفاعل ثلاثي الأبعاد مع محتوى الوب. ويستند هذا النموذج الجديد إلى تمثيل عناصر محتوى المعلومات بمنظور ثلاثي الأبعاد وتنظيمها داخل حاوية ثلاثية الأبعاد نطلق عليها اسم "Web Cube". إضافة إلى ذلك يشتمل النموذج المقترح على آليات التفاعل المناسبة باستخدام إيماءات اليد. وقد تم تقييم النموذج الجديد باستخدام طريقة تجريبية لتقييم كفاءة أداء المستخدم، وطريقة الاستبانة لتقييم مدى رضا المستخدم في التفاعل مع النموذج.

الكلمات الرئيسية: وب ثلاثي الأبعاد، تصفح مواقع الإنترنت، الواقع الافتراضي، الإيماءات اليدوية، تتبع اليد، تفاعل الانسان والحاسوب، تقييم التفاعل .

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AdaBoost	Adaptive Boost
API	Application Programming Interface
CPU	Central Process Unit
CSS	Cascade Style Sheet
CSS3	Cascade Style Sheet level 3
fps	Frame Per Second
GB	1024 Mega Byte
GHz	1024 Mega Hertz
HCI	Human Computer Interaction
HTML	Hypertext Markup Language
HTML5	Hypertext Markup Language version 5
MMOGs	Massively Multiplayer Online Games
OpenCV	Open Source Computer Vision
OpenGL	Open Graphic Library
OS	Operating System
RAM	Random Access Memory
UI	User Interface
URL	Uniform Resource Locator
VBI	Vision Based Interface
VR	Virtual Reality
W3C	World Wide Web Consortium
WebGL	Web-Based Graphic Library
WHATWG	Web Hypertext Application Technology Working Group
WoW	World of Warcraft
WWW	World Wide Web

List of Appendices

Chapter 1

Introduction

This chapter introduces the thesis work by explaining the thesis problem, research objectives, thesis significance, scope and limitation of the thesis, methodology, resources and tools. However, it is necessary first to prelude to the problem by introducing some common interaction styles.

1.1. Interaction styles

Early World Wide Web (WWW) provides a decent environment for information presentation and exchange. Essentially, Internet web pages relied on hyperlinks that were an efficient navigation-oriented interaction paradigm for WWW and web pages navigation. The navigation-oriented interaction paradigm enables the system to be able correctly expect the user's actions [2].
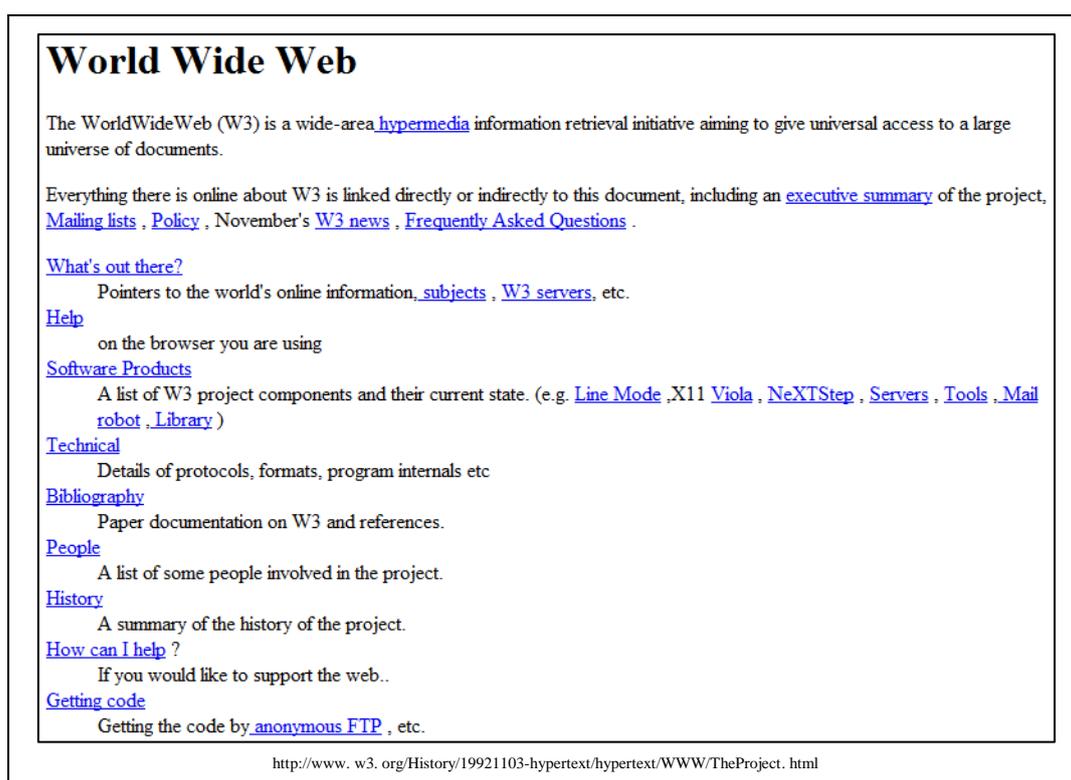
Figure 1.1: The first web page published by Tim Berners-Lee.

The essential form for Internet information was the textual form, Figure 1.1 depicts the first web page published by Tim Berners-Lee the inventor of World Wide Web (WWW). During the last decade non-textual information such as photos, videos and animations were used to present information in the Internet [20]. Nowadays Internet has become a large

environment that reaches billions of users through information sharing and processing, last statistics at June 15th, 2011 illustrate 230 million website and about 2.1 billion users are available on the Internet [14]. In the following subsections we discuss interaction styles used for our model.

1.1.1. Web Interaction

As seen in Figure 1.1 the only job webpage does is representing static textual information and the interactive part in the page is the textual hyperlink. Information transfer over the web is mostly one way, from the web to the user. There are few techniques make the web more interactive. One of these techniques is to add a section to the web page that gives people a form to fill it with appropriate information, and then submit it. These can be used for email feedback, entry into guest books that appear on the page as general comment forms that get saved into a database. Furthermore interactivity can be obtained by polls and article comments. Another technique of making the web more interactive is by setting up customization features for users, this can be done by storing information about how the users want information to be presented to them, what information they want to be presented, and also whether they want to be notified when updating information, for example being notified by email when a webpage was updated. A different technique used by some web sites by making the whole web site editable by any user or by selected group of users.

In order to understand why new web interactivity would be useful, we have to understand the currently employed interaction techniques by reviewing sites those use a few interactive components to those fully interactive sites.

With the increase of Internet speed, it would be possible to include rich content in web sites like videos, images, animations and 3-D content that simulates real objects such as marketing products, human body parts and museum collections. 3-D graphical interfaces were used widely for the developing video games, and then they were used for products marketing by adding human feeling interaction to the Internet [33].

In the last few years, 3-D web environment was one of the most interesting and evolving research areas, but the effort in this area of research is not on the level of ambition [5]. Nowadays, there are numerous of researches and projects that exploit 3-D environment, there is a promised future for this trend of using 3-D technology due to an opportunity to improve human computer interaction (HCI). So, users of different domains such as commercial,

2

medical, etc. interact with 3-D graphical interfaces which open more doors for research in this area.

Figure 1.2: Virtual Reality 3-D web.

We classify these projects into two main types according to deploying 3-D environment in the Internet. The first type based on creating a full 3-D environment in the Virtual Reality (VR) "3-D interaction" while the second uses 3-D presentation as a style for organizing search results. In the first, 3-D interaction environment consist of 3-D buildings, 3-D streets, 3-D gardens, 3-D surfaces, and so on. Users in this environment can tour through it and interact with its content. This type put users in VR environment enables them to tour freely with unconfined area Figure 1.2 depicts this type of 3-D web. Projects of this type have significant impact for simulating the real world locations to improve user knowledge of locations which cannot be reached easily in the real world. The main problem of this type is that abandoned hyperlink page style.

3
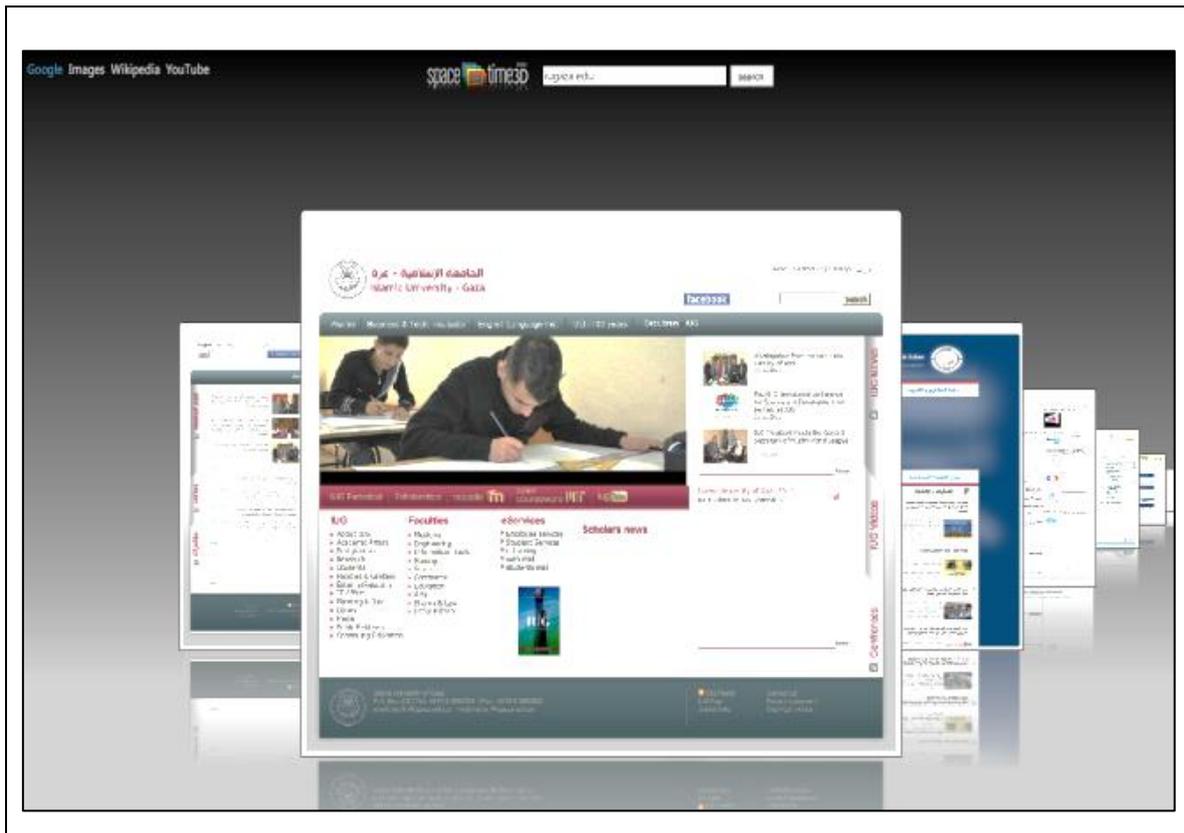
Figure 1.3: 3-D web presentation as a style.

In the other hand, the second type of 3-D environment still in the beginning and a few attempts were found. Some websites try to employ 3-D web environment only for browsing media files such as images and videos in 3-D perspective. Figure 1.3 depicts the second type of 3-D web. This type of 3-D environment used as a style for organizing search results and no 3-D interaction taken into account.

1.1.2. Hand Gesture Based Interaction

In the last few years new interface devices used for more interaction, some of them are vision based interfaces (VBI) that have been widely used in virtual reality, but they are rarely deployed in the web environment [36]. Project Natal from Microsoft Corporation is a project in which no control device required to interact with except user body gestures. As shown in Figure 1. 4 the girl does not use any control device to interact with computer. Also, Wii-remote is a remote control device that enables the user to interact with game applications distantly [10]. The use of hand gesture interfaces is one of the most important parts of VBI because hand gesture is convenient and can offer more information, taking advantage of this massive information gives more interactions, for example moving in the third dimension cannot be applied easily using conventional interfaces such as mouse or keyboard but it is

4

easy to apply using user's hand [18]. Hand as interaction interface is more realistic than conventional interfaces such as mouse and keyboard [3].

Besides the shortcoming of being biased to one of the two 3-D web types mentioned previously, neither group employs the more natural 3-D interaction mechanism of hand gestures. When interacting with 3-D objects in 3-D spaces users should not be deprived of natural interaction mechanisms such as hand gestures.
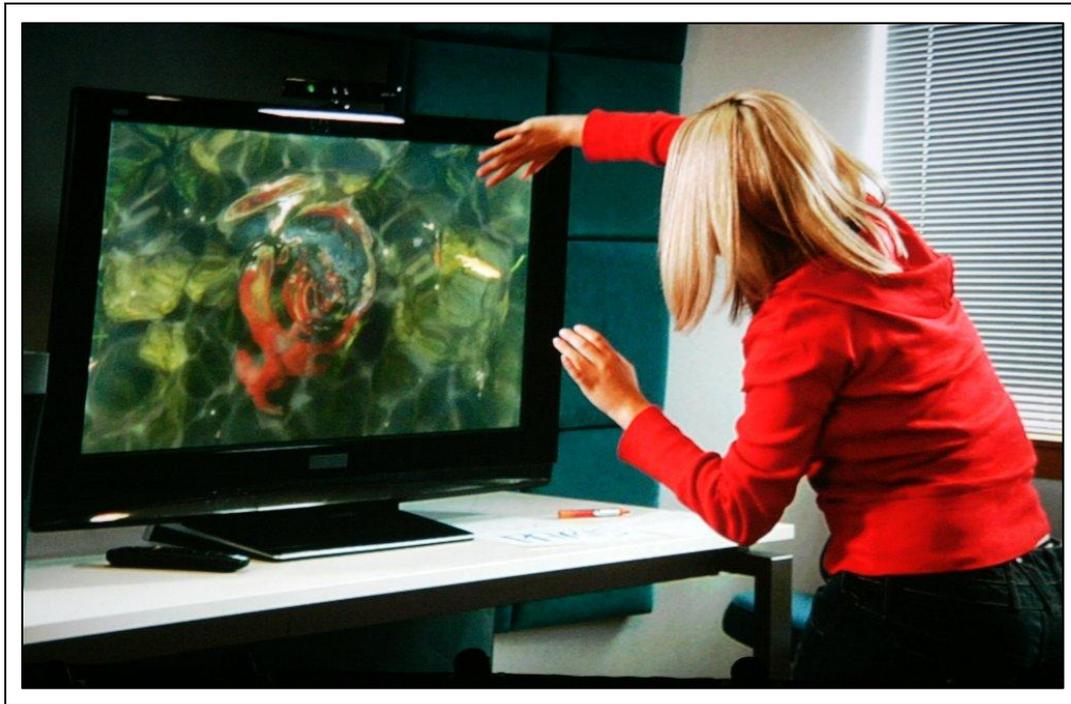
Figure 1.4: No control device required to interact with computer.

1.2. Problem Statement

Obviously none of the aforementioned web based interaction styles offer the user a full 3-D interaction experience. What is clearly missing are two things:

a) the combining of features from both 3-D interaction and conventional web browsing, and

b) the employment of a suitable interaction style such as hand gesture based interaction.

Both requirements should be implemented effectively and efficiently in order to provide web users a better browsing experience. That is, 3-D web browsing is not about virtual reality alone, and is not about navigation alone, it is about both plus the proper interaction style, altogether aimed for a better user experience. The challenge, therefore, is to develop a model for such 3-D web browsing experience.

5

1.3.   Objectives

In this section we explain our objectives in term of a main objective, and a set of specific objectives.

1.3.1.   Main Objective

Our goal in this research is to develop a new approach to 3-D browsing of web sites that effectively combines features from both 3-D web design and conventional web browsing, and allows for natural user interaction using hand gestures in order to provide a more effective and satisfying browsing experience for web users.

Figure 1.5 depicts the concept of integrating the virtual 3-D environment with conventional web browsing. This model is not to replace the 2-D flat web page model. It will rather complement it in areas where 3-D presentation and interaction is more suitable. Designers will have a new alternative for designing web content based on what is best for users.
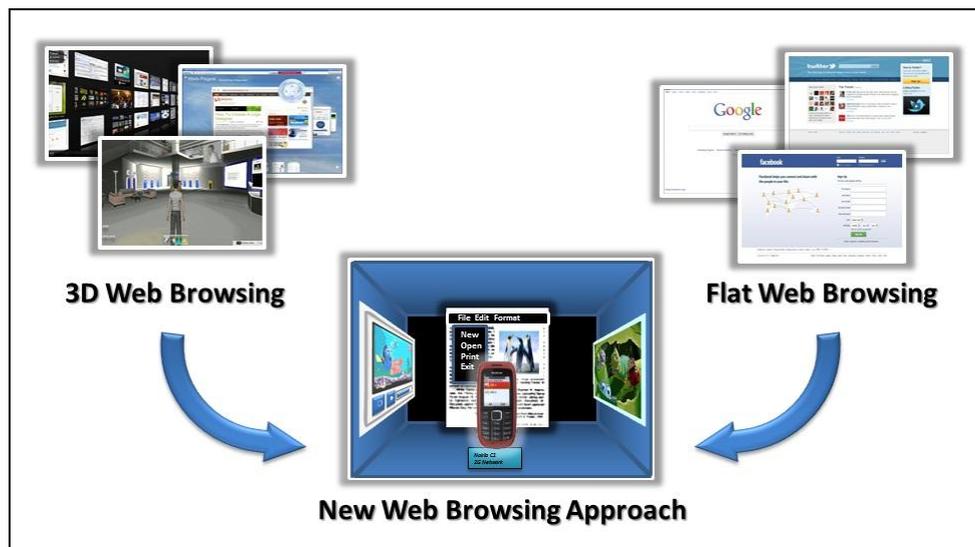
Figure 1.5: Proposed web browsing approach.

1.3.2.   Specific Objectives

1. Creating a model of restricted 3-D space "3-D Web Cube" that contains 3-D web contents.

2. Creating 3-D containers of most common web contents such as label, image, video, calendar, gallery, link and frame.

6

3. Developing actions used to manipulate the 3-D web contents, these actions: zoom, flip, rotate, drag and drop, translate and media playback.

4. Developing the real time hand detection and tracking module.

5. Integrating the actions with the hand detection module.

6. Evaluating the new model using experimental and questionnaire evaluation methods.

1.4. Significance

The significance of our newly proposed model lies in a set of important benefits:

1. Adopting the more convenient conventional navigation-oriented paradigm within 3-D interaction environment using hyperlinks.

2. Putting users in a semi-realistic environment, and allowing him to easily interact with contents of the website as in real life.

3. Overcoming shortcomings of the conventional input mechanism by using hand gestures. For example, simple actions such as transferring or rotating in the third dimension with conventional interfaces needs prior training while using hand gestures depends on the natural hand motion.

4. Increasing user satisfaction which leads to an improved browsing experience.

5. Providing business owners a new alternative for displaying their products

6. Providing designers with a new alternative for designing web content based on what is best for users.

These benefits are direct results of the two main basic features of our model that have been outlined previously which are:

1. The integration of features of 3-D web browsing technology and conventional web browsing technology.

2. The use simple and natural input interface such as hand gestures.

7

1.5. Scope and limitations

1. In order to manage the project within time constraints, the proposed solution will implement a proof-of-concept prototype of the Web Cube that will encompass primary features and operations.

2. The Web Cube prototype will include some basic 3-D web content components such as text frames, photo cards, video panels, and a simple 3-D model of some commercial object.

3. On the interaction side, the prototype will implement a few hand-gesture-based interaction operations, as well as conventional keyboard and mouse interactions.

4. Various aspects of information exchange, especially security will not be handled in our work since they depend on the underlying technology which is clearly outside the scope of this work.

1.6. Methodology

To accomplish the research objectives, the following methodology will be followed:

1. Evaluate and investigate the existing technologies can be used for developing the Web Cube prototype.

2. Developing the Web Cube model prototype which covers:

   a. Creating the Web Cube, and

   b. Creating web content elements inside the Web Cube.

3. Developing the Real Time Hand Detection and Tracking Module

4. Developing the Interaction Operations Module.

5. Integrating all modules together.

6. Finally, applying some HCI evaluation techniques, such as experimental evaluation and questionnaire to evaluate the proposed prototype.

1.7. Resources and Tools

Our work required some IT resources and a set of development tools. Resources have included:

1. Desktop computer with Intel® Core™ i7 CPU 860@ 2.80GHz 2.93GHz, 4.00 GB RAM,

2. MS-Windows 7 64-bit Operating System (OS),

3. LG displaying screen with 1680×1050 resolution,

4. MegaPixel cheap webcam with 320×240 resolution and 15 (fps).

5. MS Office Word 2010,

6. Printer, and

7. Paper.

Development tools were mainly are web programming technologies which include:

1. HTML5 specification,

2. CSS3 specification,

3. JavaScript scripting language, and

4. WebGL "Three. js" API.

Evaluation tools were used are HCI evaluation methods: experimental and questionnaire.

1.8. Thesis Structure

This thesis consists of six chapters: Introduction, Theoretical and Technical Foundations, Related Work, Web Cube Model Prototype, Evaluation and Results, and Conclusion and Future Work. The main points discussed in the chapters are:

1. Chapter 1 "Introduction" gives a short introduction about 3-D web interactions, thesis problem and objectives.

9

2. Chapter 2 "Theoretical and Technical Foundations" describes theoretical foundations needed for thesis such as HCI Evaluation methods and technical foundations such as HTML5, CSS3, JavaScript, WebGL and OpenCV.

3. Chapter 3 "Related Work" presents related work to the thesis.

4. Chapter 4 "Web Cube Model Prototype" is devoted to the presenting the implementation of the new model prototype "Web Cube" and describes the prototype architecture and the developed modules.

5. Chapter 5 "Evaluation and Results" presents the evaluation of the new model using experimental and questionnaire evaluations methods.

6. Chapter 6 "Conclusion and Future Work" discusses the final conclusions and presents possible future works.

Chapter 2
Theoretical and Technical Foundations

Our proposed model has both theoretical and technical foundations. With theoretical foundations we mean the underlying scientific theories and models which guide the conceptual design of our proposed model. These come from the discipline of HCI (Human-Computer Interaction), which is basically a multi-disciplinary field that aims to study and suggest methods for designing interaction to fit users.

Technical foundations, on the other hand, are the set of technologies based on which our proposed model can be implemented and brought to life. These include various technological tools and services most of which are recent developments in web and graphics design.

In the following sections we elaborate on both of these theoretical and technical foundations and their various constituent elements.

2.1. Theoretical Foundations

HCI provides not only the theoretical background for understanding the user and developing the right design for him, but it also provides the methods and techniques for evaluating whether or not a particular design provides the right interaction for its target user base. In the following subsections we discuss the relationship between HCI and our proposed model and present the evaluation schemes that will be used to test the suitability of our proposed model.

2.1.1. HCI Foundation

The purpose behind our proposed web browsing model is to enhance user experience on the web through 3-D content and interaction. For this reason, our work falls under HCI which studies the interaction between people (users) and computers in order to suggest methods for designing the interaction of end-user systems that are user-friendly and responsive to users' needs so that they can achieve their tasks faster, with less mistakes, and greater satisfaction [8]. In HCI, these objectives are collectively termed "usability". The degree to which an application meets its usability goals can be measured using various evaluation methods that have been developed in HCI to assess that application's usability on different dimensions including: usefulness, learnability, efficiency, and user satisfaction.

The evaluation methods are classified into two main approaches each has its own methods. First approach is based on expert evaluation like analytic methods, review methods, model-based methods. Second approach involves users within the evaluation like experimental methods, observational methods and query methods. Section 2.1.2 elaborates more on evaluation methods.

2.1.1.1. Usability

Usability is a quality attribute that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process. Usability is defined by 5 quality components:

- Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design؟

- Efficiency: Once users have learned the design, how quickly can they perform tasks؟

- Memorability: When users return to the design after a period of not using it, how easily can they reestablish proficiency؟

- Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors؟

- Satisfaction: How pleasant is it to use the design؟

2.1.2. Evaluation Methods

HCI proposes various techniques for evaluating an end-user application. Selecting the appropriate evaluation technique(s) depends on the application itself. At the broadest level, these techniques can be divided into two main categories:

- **Expert-based evaluation**: refers to any form of usability evaluation which involves an HCI expert examining the application and estimating its likely usability for a given user population. In such cases, users are not employed and the basis for the evaluation lies in the interpretation and judgment of the evaluator [8].

- **User-based evaluation**: tests an application with a sample of users performing a set of pre-determined tasks is generally considered yielding the most reliable and valid estimate

of an application's usability. Performed either in a usability test laboratory or a field site, the aim of such a test is to examine the extent to which the application supports the intended users in their work. Tightly coupled to the operational approach to usability definition, the user-based approach draws heavily on the experimental design tradition of human factors psychology in employing task analysis, pre-determined dependent variables and, usually, quantitative analysis of performance supplemented with qualitative methods [8].

The following table gives a brief idea about some of the techniques that fall under each of these categories and notes their advantages and disadvantages.

Table 2.1: Evaluation techniques

	Technique	*Description*	*Advantages*	*Disadvantages*
Expert-based	Cognitive walkthrough	expert 'walks though' design to identify potential problems using psychological principles	Focuses on goals and knowledge	Expensive and need long time.
	Heuristic checklist	usability criteria (heuristics) are identified	Cheap and fast	Expert-variability unduly affects outcome, may overestimate true number of problems
	Review-based	Results from the literature used to support or refute parts of design.	Provides rigorous estimate of usability criterion, can be performed on interface specification	Measures only one component of usability, limited task applicability
User-based	Experiment	Controlled evaluation of specific aspects of interactive behavior.	specialist equipment available, uninterrupted environment	Lack of context
	Questionnaire	Set of fixed questions given to users.	Quick and reaches large user group, can be analyzed more rigorously	Less flexible and less probing
	Observation	user observed performing task or asked to describe what he is doing and why, what he thinks is happening etc.	Simplicity and requires little expertise	Subjective, selective, act of describing may alter task performance

13

2.1.2.1. Experimental Evaluation

One of the strongest methods of evaluating is to use a controlled experiment. This provides experimental proof for a particular claim or hypothesis. It can be used to study a wide range of different issues at different levels of detail [8]. The aim of experimental evaluation is to highlight the effect of the experimental factor and avoid undesired effects. This is achieved by strictly controlling the impact of irrelevant variables [12]. Underlying most experimental evaluation is the idea of comparing the performance of users who have experienced different levels of the experimental factor. The impact of irrelevant subject-related variables can be controlled by randomly allocating the users to the various conditions of the experiment [12].

Advantages

1. Specialist equipment available.

2. Uninterrupted environment.

Disadvantages

1. Lack of context.

2. Difficult to observe several users cooperating.

2.1.2.2. Questionnaire

A list of a research or survey questions asked to respondents, and designed to extract specific information. It serves four basic purposes to:

1. Collect the appropriate data.

2. Make data comparable and amenable to analysis.

3. Minimize bias in formulating and asking question.

4. Make questions engaging and varied.

The questions should seek to describe such diverse user attributes as past behaviors, expectations, attitudes and opinions towards the user interface.

14

Advantages

1. Questionnaires are cheap to make and easy to apply to a large sample of users.

2. They can quickly provide both quantitative and qualitative data.

Disadvantages

1. Questions are fixed: there is seldom the possibility to include new questions on request from the user. There is no standardized way in which more expanded information can be forthcoming.

2. The evaluator cannot always control the situation or the manner in which the questionnaire is answered.

3. Questions misunderstanding may produce inaccurate results [12].

From the above discussion we can see that the Experimental and Questionnaire techniques are the ones that can be used to evaluate our model for the following reasons:

1. The model is partially implemented, so cognitive walkthrough cannot be used.

2. Our model is newly proposed; therefore it is difficult to find heuristics (or criteria) for evaluating it.

3. No previous similar models to review.

4. We need to evaluate specific aspects of interactive behavior.

Chapter 5 "Evaluation of the Web Cube Model" is dedicated for the evaluation, and includes detail descriptions of the evaluation experiments and questionnaires we used to evaluate our proposed model.

2.2. Technical Foundations

As mentioned at the beginning of this chapter, technical foundations include the set of technologies based on which our model can be implemented. Basically, our model relies on two sets of technologies:

- **Web and 3-D Web design technologies**: these are the tools needed for designing the web cube and its components; both flat and 3-D components. This set includes: HTML5, CSS3, JavaScript, WebGL, and OpenCV.

- **Hand gesture technologies**: this set includes the technology and algorithm used for hand detection and gesture recognition. This set includes OpenCV for image processing operations, and robust real-time object detection as the algorithm for hand detection and gesture recognition.

In the following subsections we discuss these technologies in further detail, and explain in what way each technology was necessary for the development of our model.

2.2.1. HTML5

HTML is the primary language for designing web pages and has a strong potential for that. The language of HTML5 is the latest version of HTML and is currently under development. The new version contains new features such as videos, sounds and animations which were previously dependent on external plug-ins such as Flash from the Adobe or Silverlight from Microsoft. HTML5 is a product of cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). They work together on HTML5 since 2004, and it is expected to remain under development for years, but completed parts will be supported in the new versions of major web browsers.

There are new elements in HTML5 such as a drawing element <CANVAS> elements, video element <VIDEO> and audio element <AUDIO> for playing the media. HTML5 provides better support for offline storage. New content elements such as <ARTICLE>, <FOOTER>, <HEADER> are included, and new input types such as a calendar, date, time and e-mail and research are added [35]. The most interesting features in HTML5 are:

- **Better control:** the language includes a distinct set of Application Programming Interfaces (API) which provides access to microphone and camera that allow the programmer to build interactive web applications.

- **Offline browsing:** where it is possible to browse web without a connection to the Internet by sorting webpage on the client.

- **Multimedia support:** You can add audio and video without any need to install any external plug-ins like Flash or Silverlight plug-in.

16

- **Graphical effects:** It includes graphics libraries which provide many tools for drawing and animations.

- **Compatibility:** It is compatible with previous versions of HTML.

2.2.2. CSS3

CSS is a language to describe the presentation and control the appearance of web pages such as colors, fonts and layout. It also allows adapting the layout, according to the different types of devices, such as large screens and small screens. It is an independent language of HTML, and this separation of HTML and CSS facilitates maintaining the appearance of sites in different environments. The language of CSS3 is the latest version of CSS that provides a range of new features that were not available in previous versions. It would be difficult to determine exactly the differences between the new version and previous versions, because the main change in the CSS3 is that it has been divided into several smaller units instead of one large unit. It has been divide the old version into smaller units and also added new units. The most important of these units: the selectors, backgrounds, borders, text effects and 2-D and 3-D transforms, animation and multiple columns on the page [36].

2.2.3. JavaScript

JavaScript is a programming language designed to adding interactivity to webpages. It has been applied for the first time in Netscape Navigator beta 2 in 1995. JavaScript is a language different from the Java language developed in 1990 by Sun Microsystems. It is executed on the client-side and integrated with HTML code. It can also be executed on server-side, but this area outside the scope of this research. JavaScript can be executed according several events like loading webpage, mouse drag-and-drop and keyboard typing, submitting a form, or leaving the page. JavaScript is an object oriented programming language.

2.2.4. WebGL

Web-based Graphics Library (WebGL) is a library of 3-D graphics for webpage development. WebGL used to increase the effectiveness of the Java language by enabling it to produce 3-D graphics with high interactivity without the need for external plug-ins. The first version was adopted in March 2011, Mozilla and Google are the most supporters of this library. The development of the WebGL based on OpenGL SE 2. 0 that used by major 3-D applications especially games. For that WebGL was supported in major Internet browsers

17

such as Mozilla Firefox, Google Chrome, and Opera with the exception of Microsoft Internet Explorer that support Direct 3-D in Silverlight.

WebGL features:

1. WebGL based on previously known technology OpenGL SE 2. 0.

2. Ease of integration with the language of HTML, and events handling.

3. The use of graphics processors in a Web environment to speed up the graphics rendering.

4. Use of JavaScript simplifies the development of webpages.

2.2.4.1. Three.js

3-D web environment development complexity encouraged developers to develop many libraries based on WebGL for easier 3-D web development using WebGL. Three. js is one of these libraries. It is a lightweight and low-complexity library. It makes easier development way for creating 3-D Web environment. It enables the developer to call functions that do many operations those require a great effort using WebGL pure programming. Three. js eases the preparation of display, loading materials and construction of 3-D objects and facilitated the interaction with these objects.

2.2.5. OpenCV

Computer Vision library OpenCV has been developed by Intel cooperation. The main advantage of OpenCV is being open-source and work on many operating systems such as Linux, Microsoft Windows and Mac OS. OpenCV have been written in C/C++ also supports other languages Python and Matlab. It focuses on real time applications. A lot of functions available in OpenCV that cover many areas of user interface, camera calibration, robotics and so on. It applies many operations such as image-data and video- data transformation, decisions making, and objects detection. One of the most important objectives of OpenCV library is to provide infrastructure for easy to use Computer Vision, where the library contains more than 500 functions and the algorithm used in this area.

Chapter 3

Related Work

In this chapter we discuss various works related to our proposed model. Since our work has two different primary components: a) 3-D Web Cube, and b) Hand gesture interaction, it is natural to divide the related work discussion into two sections; one relative to each component. Furthermore, it is important to direct the reader's attention to the fact that the first component is more technically-oriented while the second is more scientifically-oriented. This difference in orientation will be obvious by the various related works presented in each of the two sections.

3.1. 3-D Web Related Work

In this section we address some web-based systems that utilize 3-D graphics technologies. One thing that will be clear in this section is that these examples vary in the degree and purpose for which they utilize 3-D graphics technologies. This fact reflects the essential problem we address in our research work, which is basically that utilizing 3-D graphics on the web has not yet taken a well-defined form and purpose.

The first related work is Google Earth that provides massive information of earth maps captured by satellites. It is one of the most used products in the world. This product enables users to explore cities, building, and streets in 3-D environment [13]. With Google Earth version 6 new features added such as 3-D street touring which enables users to walk through buildings. Figure 3.1 depicts how Google Earth 6 views 3-D buildings and streets.
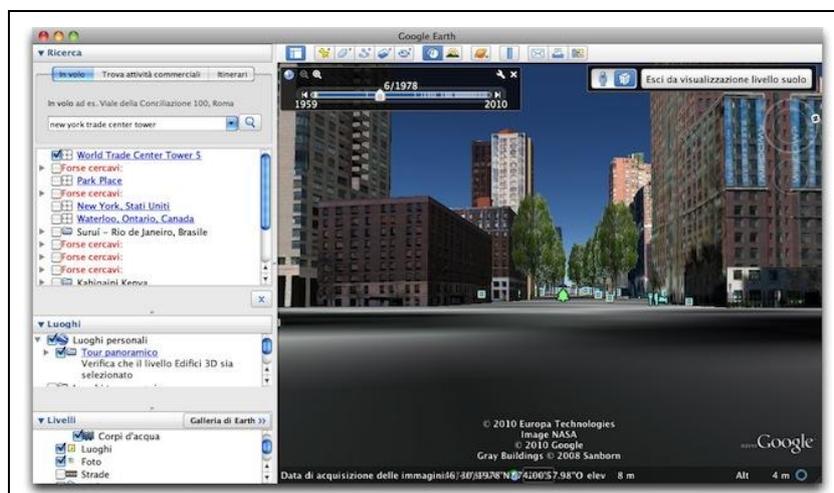
Figure 3.1: Google Earth 6 3-D street view.

Massively Multiplayer Online Games (MMOGs) has grown fast [12], World of Warcraft (WoW) game shown in Figure 3.2 is one of the most famous Internet games based on MMOGs. WoW allows user to choosing a persona and to join fighting adventure with other users in a 3-D environment.

Figure 3.2: World of Warcraft web game.

3-D interaction environment such as SecondLife shown in Figure 3.3 could be a good educational environment over the Internet [27]. SecondLife was launched in 2003 as other world corresponding real human world. Nowadays, millions of users join and tour this 3-D interaction world. Users in this 3-D interaction world can buy and sell everything such as houses, cars, lands etc. Moreover, they can contact each other as in the real world.

Figure 3.3: People gathered at SecondLife.

Other educational products use this type of 3-D interaction environment such as Edusim shown in Figure 3.4. Edusim was developed to help primary school students to use 3-D interaction environment and touch technology in their classes [34].
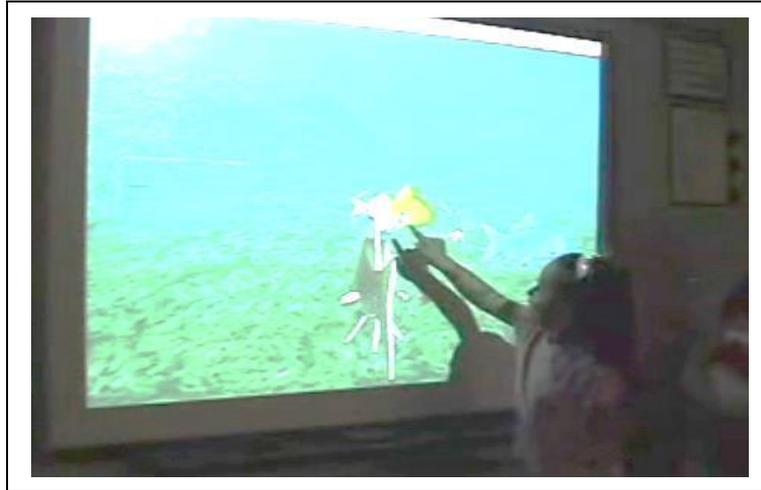
Figure 3.4: Edusim project used at primary school.

www.cooliris.com is an example of the web based on 3-D environment. It displays images and videos in 3-D perspective as shown in Figure 3.5 (a). Another example of this type is a search website that views the results of user search in 3-D frames. Figure 3.5 (b) depicts search. spacetime.com search site browsing the search results in 3-D perspective.

| (a) Cooliris plug-in used in web browser. | (b) Search for "iugaza. edu. ps" using spacetime. com. |

Figure 3.5: 3-D web presentation as a style.

3.2. Hand Gesture Related Work

This section presents several hand detection and tracking approaches introduced by various researchers.

In [30] Qureshi et al propose an algorithm converts an input image into gray scale image and then into a binary image using adaptive thresholding techniques. Edges are also sharpened and distortion is minimized for a pretty clear and perfect image of the hand. In the proposed algorithm it has been assumed that if a finger is active it's ending point will be at farthest from the center of the hand.

21

A neural network algorithm proposed by Stergiopoulou et al [38] for hand detection. In this algorithm hand recognition process is the detection of the hand region, this achieved through color segmentation, i.e. classification of the pixels of the input image into skin color and non-skin color clusters. The proposed method extracts features that describe the properties of the fingers.

Wang and Popovic [37] design a glove consists of many color patches makes camera be able to detect the hand and track it. The glove designers use data-driven technique to track the hand with the designed glove. They use Cyber-glove II motion capture to collect a large number of finger positions. Finally they enhance their system by increasing the database searching speed Figure 3.6 depicts the colored gloves used.

Figure 3.6: Colored gloves used for hand detection and tracking [37].

A widely used real time object detection algorithm proposed by Voila and Jones [33] modified for the task of detecting hand presence, then combined the new method of real time hand detection with skin-color filtering to achieve hand detection in complex backgrounds. For 2-D hand detection and tracking they use single web camera and propose new haar-like features [18]. For 3-D hand detection and tracking we must propose new features or use more than one camera [28].

22

Figure 3.7: SixthSense presented by Mistry and Maes [21].

SixthSense presented by Mistry and Maes [21] depicted in Figure 3.7, is a wearable gestural interface that augments the physical world around us with digital information and lets us use natural hand gestures to interact with that information. By using a tiny projector and a camera coupled in a pendant like mobile wearable device, SixthSense sees what the user sees and visually augments surfaces, walls or physical objects the user is interacting with; turning them into just-in-time information interfaces. SixthSense attempts to free information from its confines by seamlessly integrating it with the physical world.

23

Chapter 4

Web Cube Model Prototype

In this chapter we present in through details the prof-of-concept Prototype that we have built to realize our proposed Web Cube model and hand gesture based interaction modality. The prototype consists of the following three modules:

1. Web Cube model and its contents,

2. Hand detection and tracking module, and

3. Interaction module.

In the following three sub-sections we describe each module in more details.

4.1. Web Cube Model and Its Contents

The main Web Cube structure as in Figure 4.1 consists of five surfaces: floor, ceiling, and three walls (right, left, and back; the front is omitted to make an entrance to the Web Cube). While these surfaces are holders of content components also, they limit Web Cube 3-D space. Just as a webpage is limited by its four boundaries Web Cube is limited by these surfaces. Furthermore, users navigate from one Web Cube to another through links.
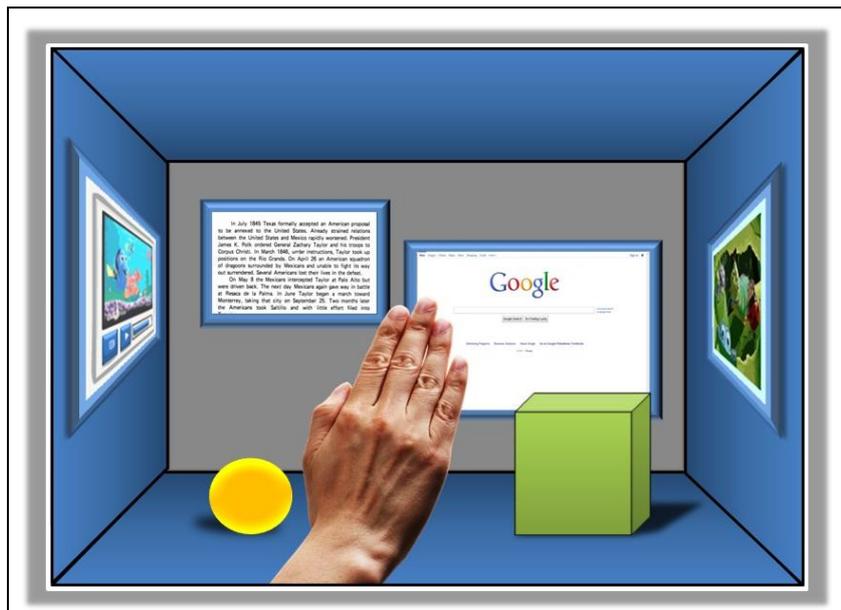
Figure 4.1: Proposed Web Cube and hand gestures interaction model.
The Web Cube is a 3-D structure, information components inside the cube or 3-D
objects, interaction by hand gestures in 3-D space.

24

A Web Cube may contain any number of content components Figure 4.2 shows an example of Web Cube with various such content components. In our prototype we have implement the following set of content types: text-label, image, video, calendar, gallery, link, frame and object-model. These types will described individually next.

Web Cube will be a text file on the server side that containing details of the design and content. This file will be loaded via HTTP protocol and rendered at the client side by the Internet browser.
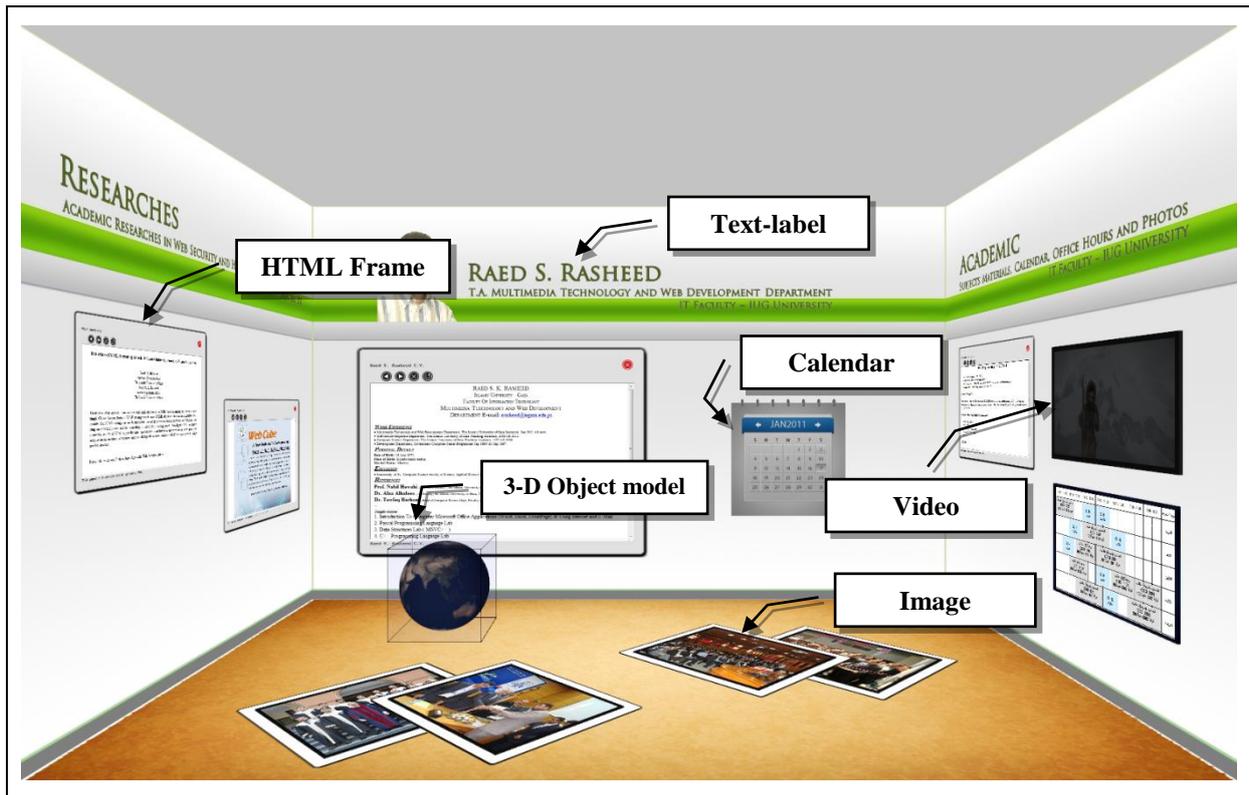
Figure 4.2: Example of Web Cube and its components.

● *Text-label:*   is the static text used to label or describe something, as seen in Figure 4.2 the text-label "RAED S. RASHEED" and "T.A. MULTIMEDIA TECHNOLOGY…" on the front wall describes the owner name and his title.

● *Image:*   this component consists of two sides the front side which contains the picture or image itself and the back side that contains the image details. Figure 4.3 depicts the front and the back of an image.

(a) Image front side.   (b) Image details in the back side.

Figure 4.3: Web Cube image component.

● *Video:*   as the image component, this component consists of two sides the front side which contains the video itself and the back side that contains the video details. Figure 4.4 depicts the front and the back of a video.

(a) Video front side.   (b) Video details in the back side.

Figure 4. 4: Web Cube video component.

● *Calendar:*   a very simple calendar enables users to explorer the date. It contains years, months and days Figure 4.2 depicts a simple calendar on the front wall.

● *Link*:   a hyperlink to another Web Cube used to load another Web Cube and its contents. The link can be used also to select another Web Cube component in the same Web Cube.

● *Gallery:*   a photo album used to arrange photos or pictures together and scroll them. Figure 4.5 depicts the implemented gallery.

26

Figure 4.5: Web Cube gallery components.

● *HTML-Frame:* is a container of textual information or HTML document. This component can be a simple web browser inside the Web Cube. Figure 4.6 depicts the Web Cube frame component.

● *3-D Object Model:* a model of any object that can be manipulated in 3-D, the object can be created using any 3-D modeling software and exported as an object file (.obj) then converted to JavaScript file (.js) using converter programs. The JavaScript file can be loaded to the Web Cube easily. Figure 4.7 depicts car object created using 3-D Studio Max software and loaded in the Web Cube after converting it to JavaScript file. See Appendix A for components code.

We develop the Web Cube environment and its content using WebGL, JavaScript, HTML5 and CSS3. The canvas element in HTML5 is used to draw 2-D graphics on the web. Drawing 3-D graphics on the canvas element needs the WebGL library that was implemented in JavaScript.

27
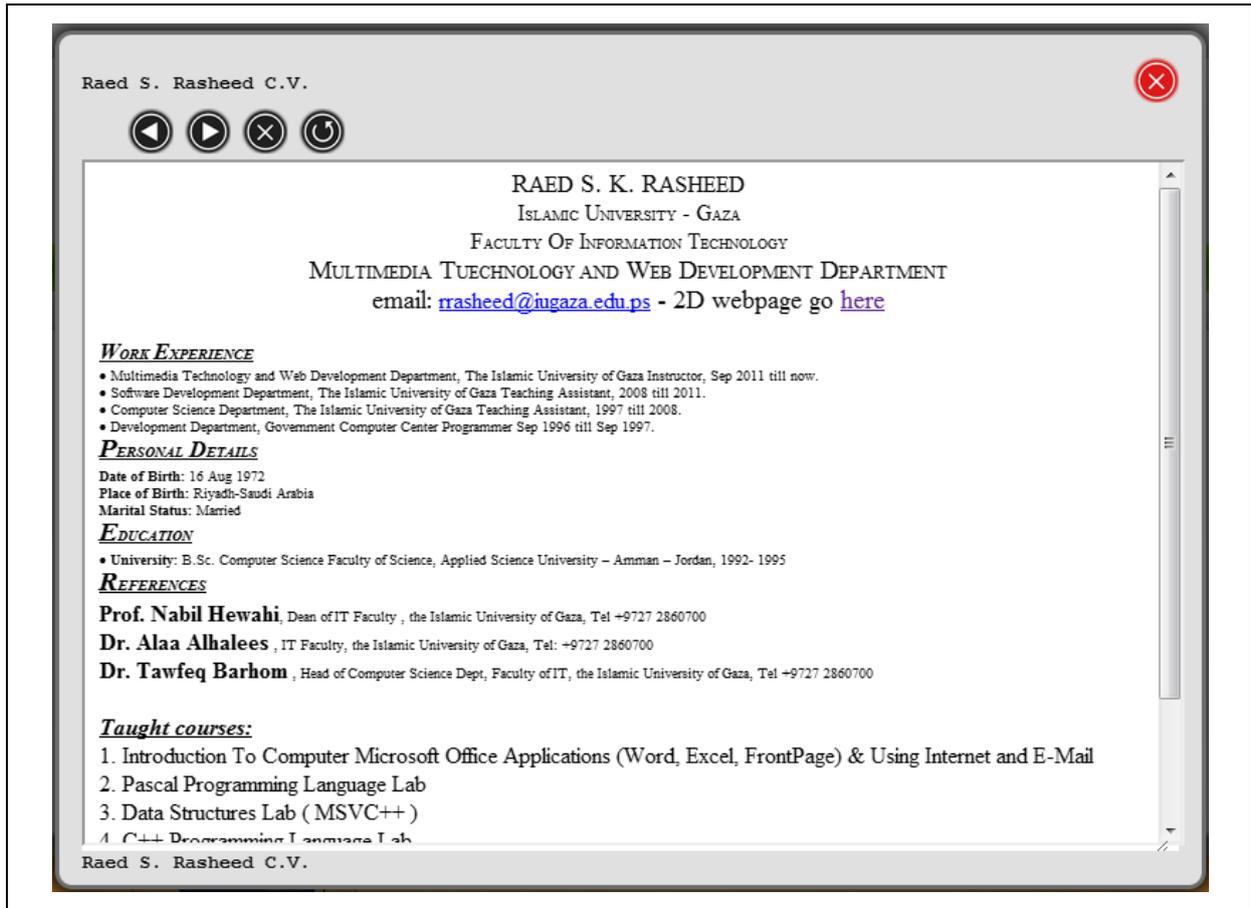
Figure 4.6: Web Cube HTML frame components.
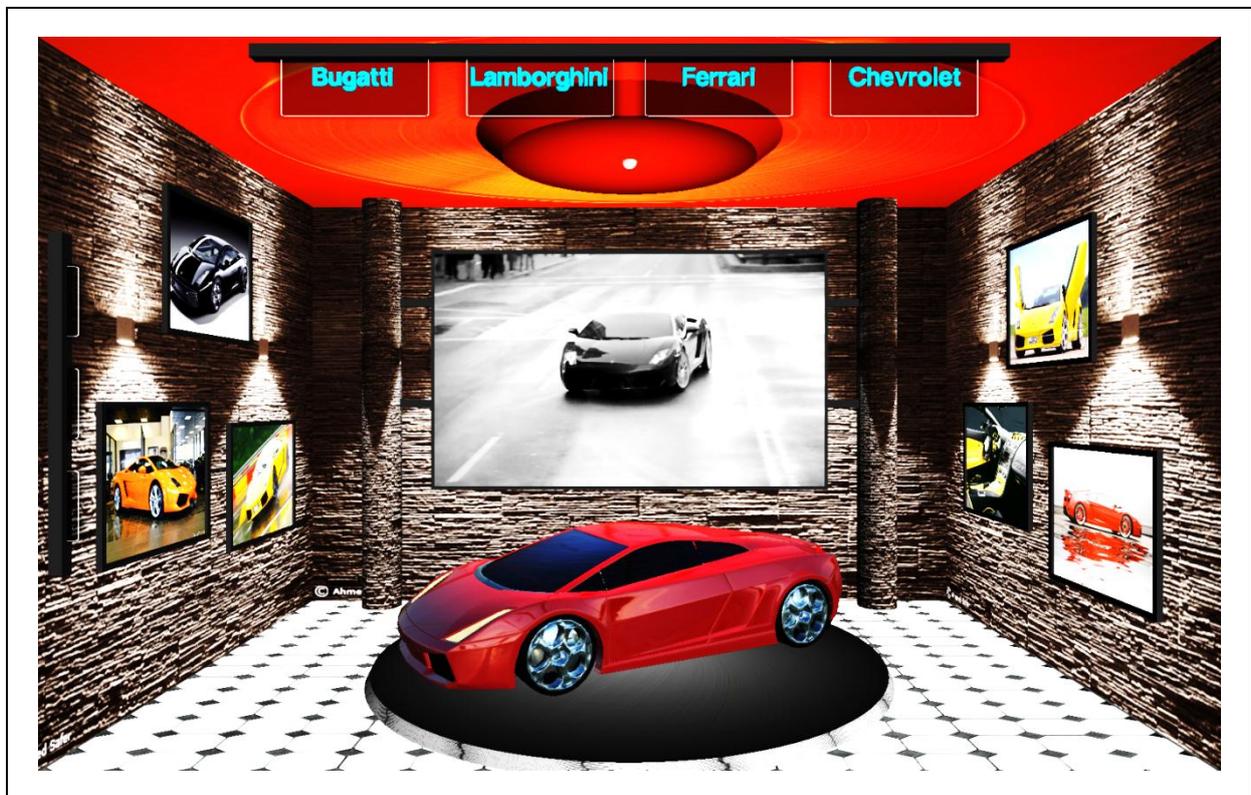
Figure 4.7: Web Cube 3-D object model components.

4.2. Real Time Hand Detection and Tracking Module

In this section we describe the hand gesture detection module. This module is responsible for detecting a set of eight predefined hand gestures. These gestures are associated with Windows mouse and keyboard operations. Therefore, in order for users to carry out a mouse or keyboard operation they will need to perform the corresponding hand gestures. Table 4.1 describes each of these gestures.

Table 4.1: The eight hand gestures associated with mouse and keyboard operations.

Hand Gesture	Description	Operation
	Hand opened.	Mouse up.
	Hand closed.	Mouse down.
	Hand closed then opened.	Mouse click.
	Hand closed then opened twice.	Mouse double-click.
	Hand closed then moved.	Mouse drag.
	Hand closed and moved then opened.	Mouse drag-and-drop.
	Hand forward.	Keyboard up arrow.
	Hand backward.	Keyboard down arrow.
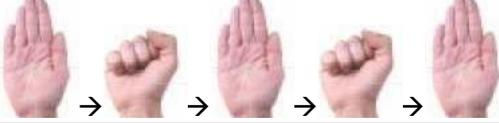
A simple webcam has been used as an input device. The real time hand detection and tracking module reads the camera input and detects hand gestures. This module uses haar-like features and Voila and Jones object detection. See Appendix B for more details about haar-like features and Voila and Jones algorithm.

We developed this module in C++ using the computer vision open source library OpenCV. This library implements Voila and Jones algorithm. By adding hand features to the implemented algorithm the module can detect and track hand. We add two hand-features sets one set for opened hand and the other for closed hand, so the module can detect the open hand and the close hand.

4.3. Interaction Operations Module

In this section we describe the interaction operation module. This module is responsible for the interaction operations with the Web Cube and its components. This module will recognize two types of operations: a) the manipulation operations type, and b) navigation operations type. The manipulation operations type is further divided into two sub-types: simple manipulation operations and complex manipulation operations. In the following we explain each of these types.

Simple manipulation operations: these interaction operations handle basic object manipulations like restore to origin position, flipping the object from its front side to its back side and vice versa, rotating the object 180o on the z-axis and vice versa, and drag-and-drop in 3-D space. Figure 4.9 depicts the buttons used for these interaction operations.
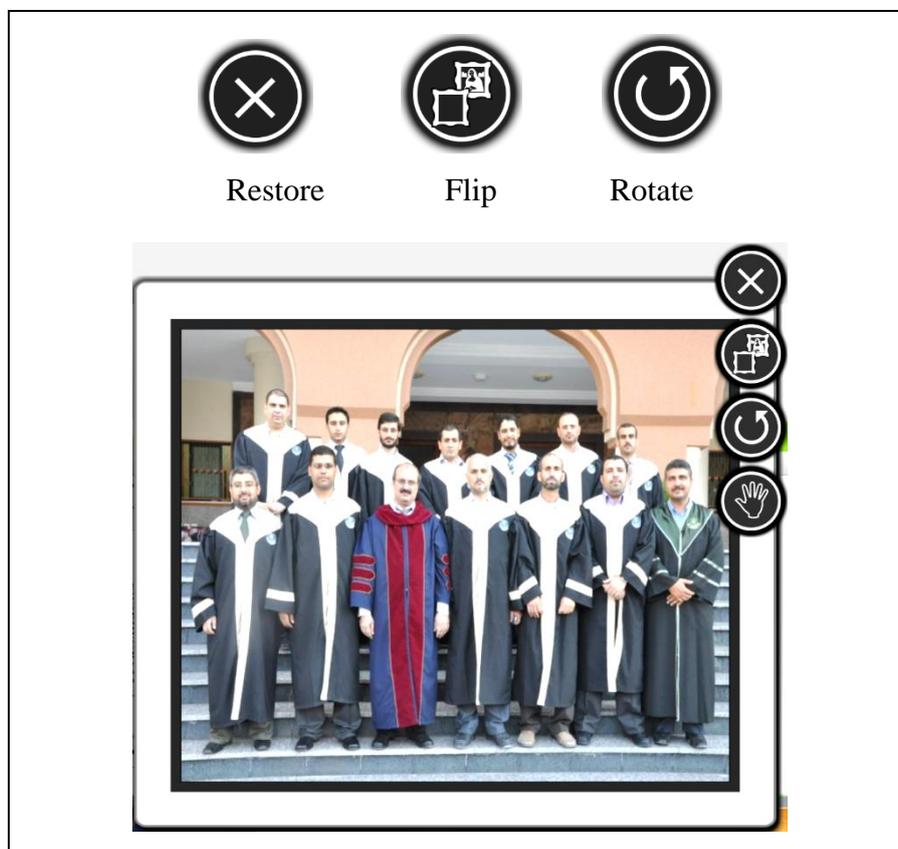
Figure 4.8: Manipulation simple-interaction operations buttons.

Complex manipulation operations: these interaction operations handle advanced object manipulations that consist of more than one manipulation simple-interaction operations like free flipping and rotation in 3-D space as depicted in Figure 4.9 the earth model can be rotated freely.

Figure 4.9: Earth model rotated freely and no angle restrictions.

Navigation operations: these interaction operations handle user navigation like walking forward and back, right and left and up and down in 3-D space as depicted in Figure 4.10 where the user walk through the Web Cube to see the left wall.

Figure 4.10: Walking through Web Cube an example of navigation operation.

31

Chapter 5

Evaluation and Results

In this chapter we present the evaluation procedures we carried out to evaluate the visibility of the proposed model. As explained in Chapter 2 we used two evaluation methods:

1. **Experimental**: to measure efficiency in performing certain tasks, and compare it with conventional interaction.

2. **Questionnaire**: to measure user satisfaction with the new experience.

In each of the following sections we present each of these two methods and its results in details.

5.1. Experimental Evaluation

This section explains how the experimental test method described earlier was applied to evaluate the visibility of the Web Cube prototype.

Experimental evaluation requires a hypothesis and an experimental design that can be used to test the hypothesis. For that purpose we formulated the following hypothesis as our main hypothesis: "Web Cube improves web interaction efficiency".

Because we evaluate a new model and participants use it for the first time we redefine efficiency in main hypothesis as the time spent to perform a specific task.

We farther decomposed this hypothesis into a set of four sub-hypotheses which were separately tested each by an experiment. Table 5.1 lists these sub-hypothesize and the user tasks which were used to test it.

Table 5.1: The description of the four experiments.

#	Sub-hypothesis	Tasks
1	Web Cube content is more visible than webpage content	- Percentage of web-content visible on screen - Time to reach goal content - No of errors
2	Web Cube navigation is easier than webpage navigation	- Number of attempts for completing a successful manipulation task
3	Object manipulation in Web Cube easier than in webpages.	- Number of manipulation forms - Number of errors
4	Data manipulation in Web Cube is as easy as in webpage	- Time taken to enter data in a data-field - Coping, pasting and cutting text data - Number of errors

5.1.1. Experiments Design

We set up a single test environment to test all four sub-hypotheses. This environment consisted of desktop computer with Intel® Core™ i7 CPU 860@ 2.80GHz 2.93GHz, 4.00 GB RAM, 64-bit MS-Windows 7 Operating System (OS), LG displaying screen with 1680×1050 resolution and MegaPixel cheap webcam with 320×240 resolution and 15 frame per second (fps). We ran four rounds of experiment one for each sub-hypotheses. In each round a different task given to the users to test the related sub-hypotheses; see Table 5.1. The tasks performed on two different sites, one uses Web Cube model and the other uses conventional model. The two sites approximately contain the same content. Figure 5.1 depicts the two sites.

(a) Conventional.
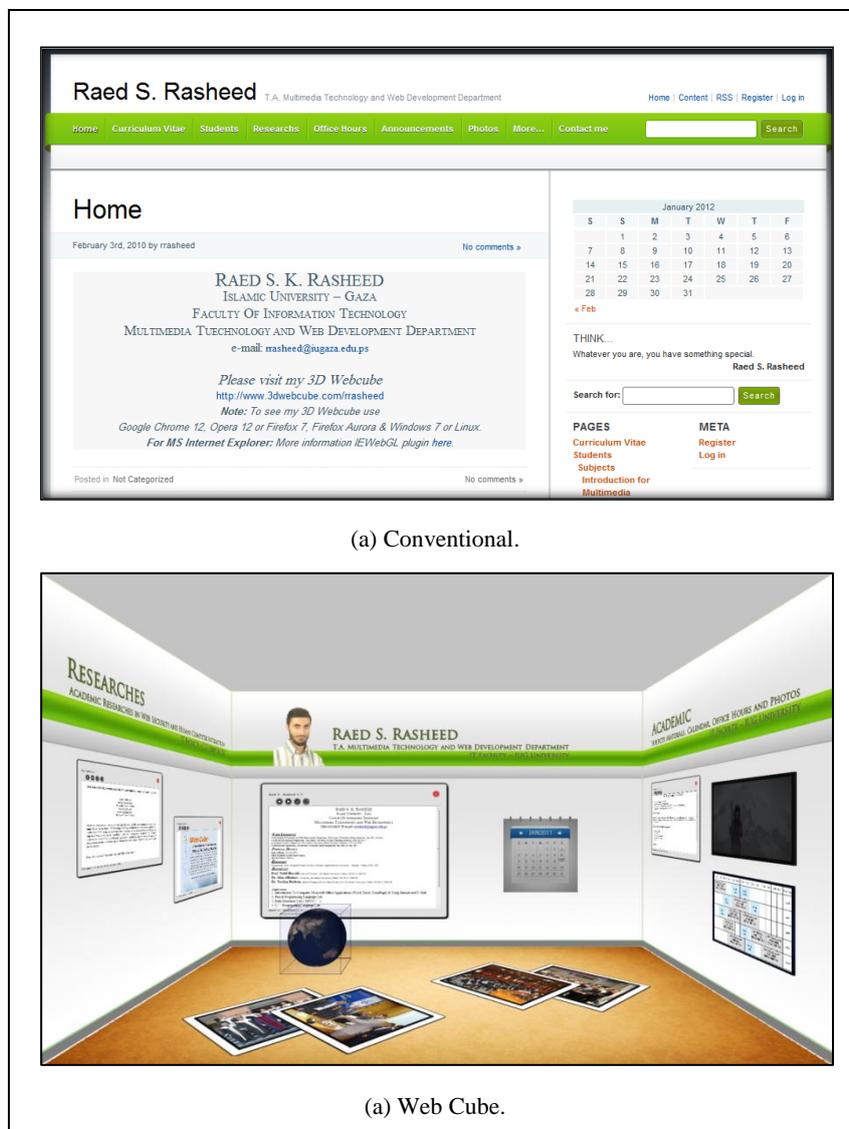
(a) Web Cube.

Figure 5.1: The two sites used for evaluation.

Web Cube Experiment		
Name:	Specialization:	Date: ....
Experiment #:	Participant #:	

Web Cube site	Web Page Site
http://www. 3dwebcube. com/rrasheed	http://site. iugaza. edu. ps/rrasheed

Sub-hypothesis:

> Web Cube content is more visible than webpage content.

- *Open the site, how many of content you can see (in percentage) when the site is loaded?*
- Web Cube: • Web Page:
- *Find the syllabus of "Web-Programming" course in the site.*
- Time taken in Web Cube: • Time taken in Web Page:
- *Number of errors during two minutes of navigation.*
- Number of errors in Web Cube: • Number of errors in Web Page:

Results:
....
....
....

Additional comments:
....
....
....

Figure 5.2: Experiment designed to test the sub-hypothesis.

Figure 5.2 depicts a sample of the experiment result form, where the participant asked to accomplish some tasks. In the form shown in Figure 5.2 the sub-hypothesis "Web Cube content is more visible than webpage content" is tested by asking the participant to do three tasks. The first task is "Open the site, how many of content you can see (in percentage) when the site is loaded?", then the participant answer the question for each site mentioned in the form. The second task is "Find the syllabus of 'Web-Programming' course in the site", then we record the time spent by participant searching in each site. The third task is "Number of errors during two minutes of navigation" means how much error occurs by the participant while navigating each site. Finally we write the final results and add comments if there exist. See Appendix C for all experiment forms used.

5.1.2. Experiments Results

All participants accomplished the tasks as instructed and they did not complain in any task. For each experiment, the net result will be the average of all participants. Table 5.2 to Table 5.5 present the data for all experiments. **P#** indicates Participant Number, **WC** indicates Web Cube with conventional input devices, **WCH** indicates Web Cube with Hand gesture input interface and **WP** indicates conventional Web Page.

5.1.2.1. First Experiment (Visibility)

Table 5.2: The results of the first experiment.

	Experiment #1 Sub-hypothesis: Web Cube content is more visible than webpage content.							
	Tasks							
	Percentage of web-content visible on screen		Time to reach goal content (in seconds)			Number of errors		
P#	WC	WP	WC	*WCH*	WP	WC	*WCH*	WP
P1	95.00%	10.00%	18.00	19.94	27.00	1.00	3.00	0.00
P2	85.00%	20.00%	24.63	28.55	33.93	1.00	2.00	4.00
P3	85.00%	17.00%	2.48	31.06	11.82	0.00	3.00	1.00
P4	60.00%	30.00%	7.50	32.69	67.48	1.00	6.00	7.00
P5	90.00%	30.00%	1.26	28.04	20.97	0.00	3.00	1.00
Avg.	*83.00%*	*21.40%*	*10.77*	*28.06*	*32.24*	*0.60*	*3.40*	*2.60*

In the first experiment we examine the sub-hypothesis "Web Cube content is more visible than webpage content" by accomplishing of the tasks shown in Table 5.2. Results Show that:

1. The average percentage of web-content visible on screen in the WC is 83% and in the WP is 21.4%. Figure 5.3 depicts the results.
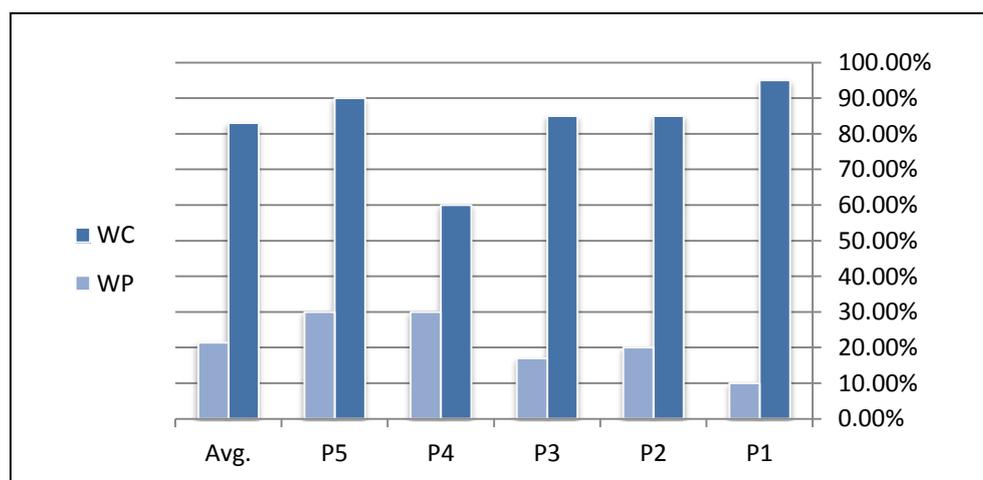
Figure 5.3: The percentage of web-content visible on screen.

2. The average time spent to reach goal content in the WC is 10.77 sec., in the WCH is 28.06 sec. and in the WP is 32.24 sec. Figure 5.4 depicts the results.

35
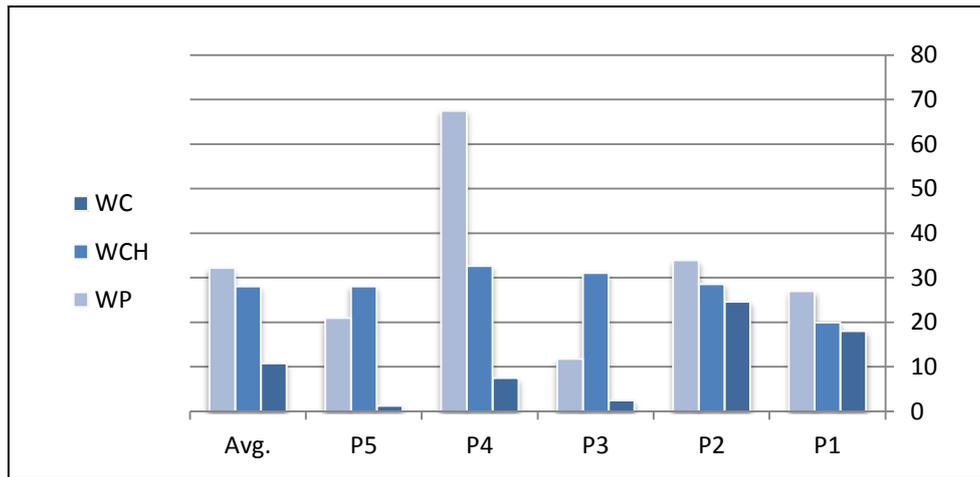
Figure 5.4: The time spent to reach goal content.

3. Finally the average number of error occurs in the WC are 0.6, in the WCH are 3.4 errors and in the WP are 2.6. Figure 5.5 depicts the results.
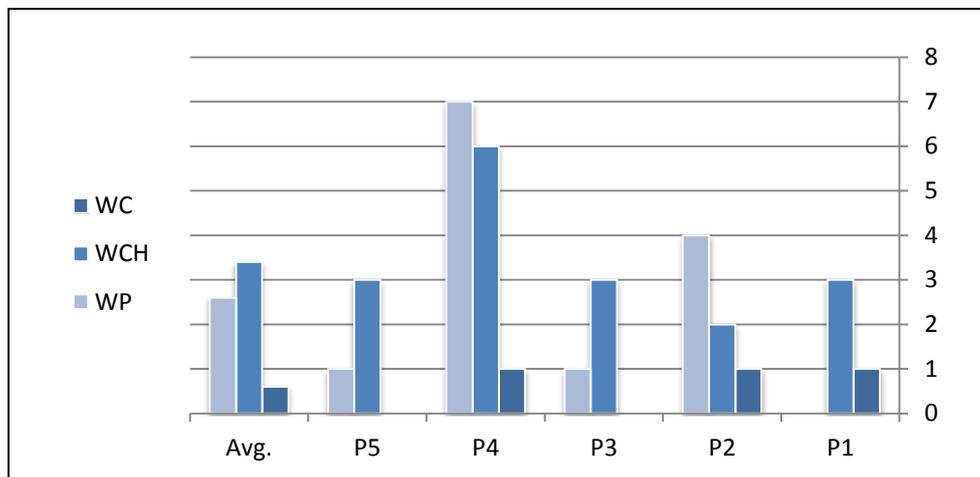
Figure 5.5: The number of error occurs.

5.1.2.2. Second Experiment (Navigation)

Table 5.3: The results of the second experiment.

Experiment #2 Sub-hypothesis: Web Cube navigation is easier than webpage navigation																
	Tasks															
	Number of attempts to find a particular content						Number of attempts to sending a note or a comment				Number of errors occurred to check calendar					
	WC		*WCH*		*WP*		**WC**		**WP**		**WC**		*WCH*		*WP*	
P#	Try	Time	Try	Time	Try	Time	Try	Time	Try	Time	Err.	Time	Err.	Time	Err.	Time
P1	3.00	34.83	3.00	16.66	1.00	17.78	1.00	47.11	1.00	41.63	1.00	0.00	4.00	76.30	0.00	0.00
P2	2.00	9.93	1.00	11.36	5.00	57.67	1.00	43.00	1.00	36.82	0.00	7.00	9.00	132.42	0.00	11.00
P3	1.00	2.08	4.00	12.04	1.00	7.93	1.00	20.86	1.00	24.68	0.00	7.02	7.00	83.17	0.00	6.45
P4	2.00	3.68	1.00	9.78	2.00	6.23	1.00	30.40	1.00	18.14	1.00	13.50	9.00	88.79	0.00	13.99
P5	2.00	3.34	6.00	45.22	1.00	3.31	1.00	35.90	1.00	27.60	1.00	20.44	7.00	110.37	0.00	10.05
Avg.	*2.00*	*10.77*	*3.00*	*19.01*	*2.00*	*18.58*	*1.00*	*30.45*	*1.00*	*29.77*	*0.60*	*9.59*	*7.20*	*98.21*	*0.00*	*8.30*

In the fourth experiment we examine the sub-hypothesis "Web Cube navigation is easier than webpage navigation" by performing the tasks shown in Table 5.5. Results show that:

1. The average number of attempts to find a particular content in WC is two while it is three in WCH and it is two WP. The average time taken to perform this task in WC is 10.77 sec. while it is 19.01 sec. in WCH and it is 18.58 sec. in WP. Figure 5.6 depicts the comparison of spent time.
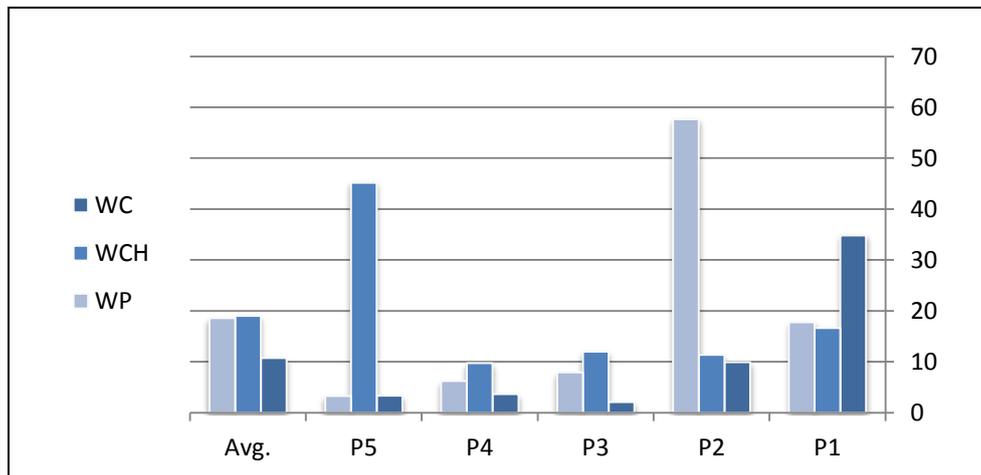
Figure 5.6: The time spent to find a particular content.

2. The average of attempts to send a note or a comment in WC and in WP is one. The average time taken to perform this task in WC is 30.45 sec. while it is 29.77 sec. in WP. Figure 5.7 depicts the comparison of spent time.
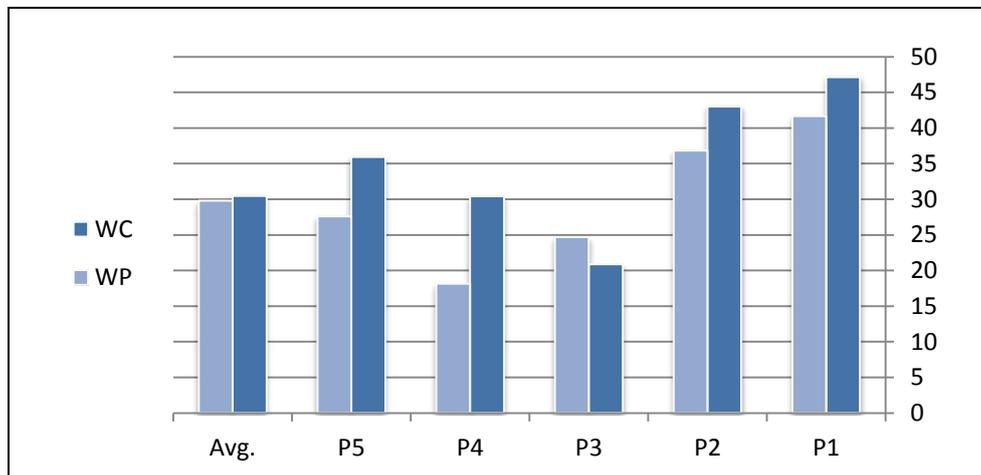
Figure 5.7: The time spent to send a note or a comment.

3. The average number of errors occurred in the WC is 0.6 within 9.59 sec. while it is 7.2 WCH within 98.21 sec. and no errors in the WP within 8.30 sec. Figure 5.8 depicts the comparison of spent time.

37
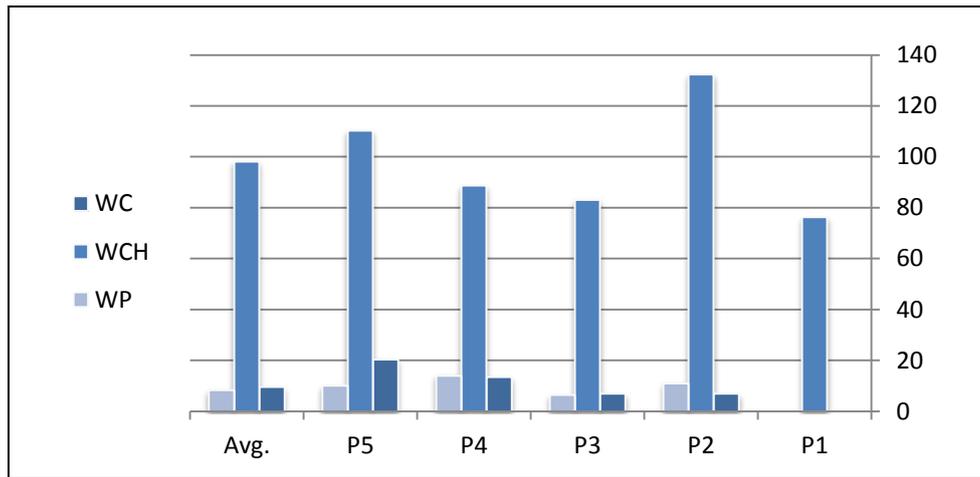
Figure 5.8: The time spent with errors occurred.

5.1.2.3. Third Experiment (Object Manipulation)

Table 5.4: The results of the third experiment.

Experiment #3 Sub-hypothesis: Object manipulation (drag, rotate, etc.) in Web Cube easier than in webpages.					
	Tasks				
	Number of manipulation forms		Number of errors		
P#	**WC**	**WP**	**WC**	***WCH***	**WP**
P1	5.00	1.00	1.00	10.00+	0.00
P2	5.00	1.00	1.00	10.00+	0.00
P3	5.00	1.00	1.00	10.00+	0.00
P4	5.00	1.00	2.00	10.00+	0.00
P5	5.00	1.00	1.00	10.00+	0.00
Avg.	***5.00***	***1.00***	***1.20***	***10.00+***	***0.00***

In the third experiment we test the sub-hypothesis "Object manipulation (drag, rotate, etc.) in Web Cube is easier than in webpages" by performing the tasks shown in Table 5.4. Results show that:

1. The total number of manipulation forms that discovered by participants in WC is five are (select, flip, rotate, drag and drop, and restore) while the average number of manipulating forms in WP is one. Figure 5.9 depicts the results.

38
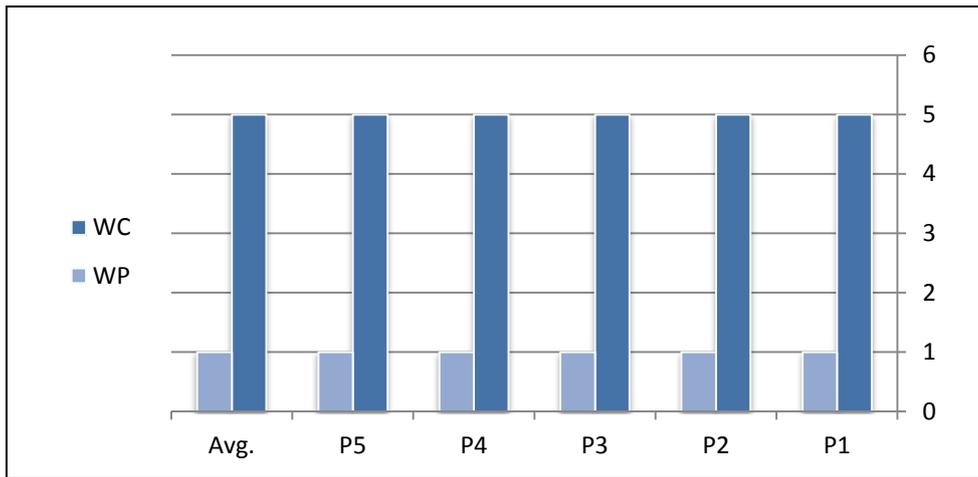
Figure 5.9: The number of manipulation forms.

2. Since there is a single form of manipulation in the webpage no error occurred while the average number of errors occurred in WC is 1.2 and in WCH is more than ten. Figure 5.10 depicts the results.
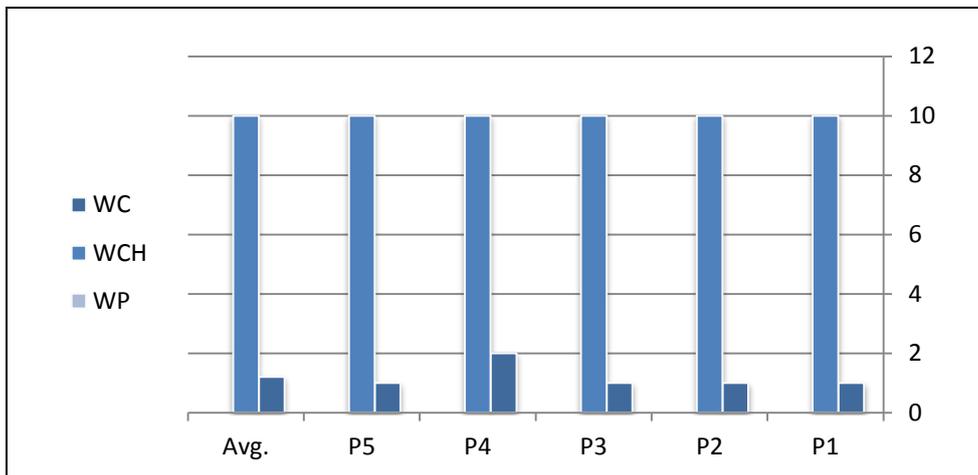
Figure 5.10: The number of errors occurred.

5.1.2.4. Fourth Experiment (Data Manipulation)

Table 5.5: The results of the fourth experiment.

Experiment #4					
Sub-hypothesis: Data manipulation in Web Cube is as easy as in webpage					
	Tasks				
	Time taken to enter data in a data-field (in seconds)		Time taken to copy, paste and cut text data (in seconds)		Number of errors
P#	WC	WP	WC	WP	WC	WP
P1	2.33	2.68	3.56	3.86	0.00	0.00
P2	1.67	1.21	3.48	3.25	0.00	0.00
P3	3.09	2.62	3.78	3.62	0.00	0.00
P4	1.74	1.39	2.18	1.71	0.00	0.00
P5	2.27	1.69	2.96	3.25	0.00	0.00
Avg.	*2.22*	*1.92*	*3.19*	*3.14*	*0.00*	*0.00*

In the second experiment we examine the sub-hypothesis "Data manipulation in Web Cube is as easy as in webpage" by performing the tasks shown in Table 5.3. Results show that:

1. The average time spent to enter data in a data-field in the WC is 2.22 sec. while it is 1.92 sec. in the WP. Figure 5.11 depicts the results.
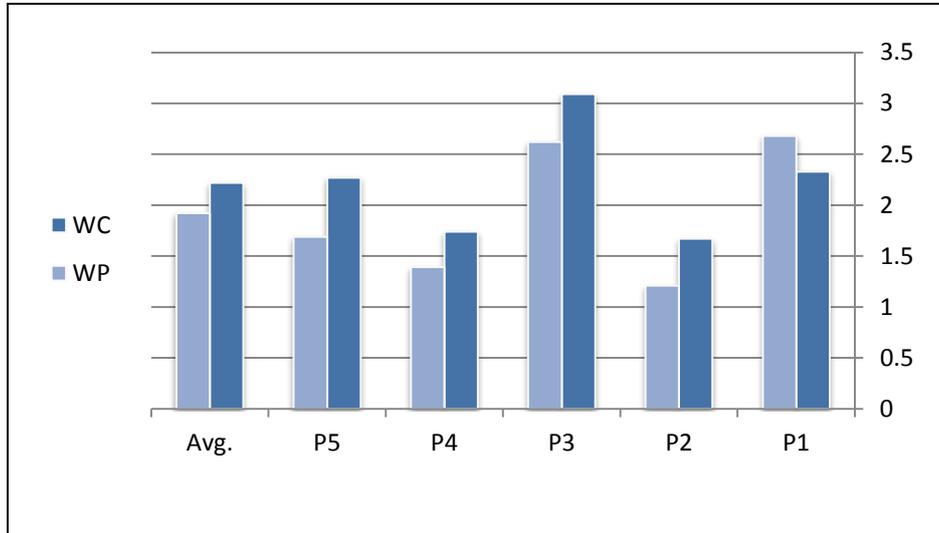
Figure 5.11: The time spent to enter data in a data-field.

2. The average time spent to coping, pasting or cutting text data in the WC is 3.19 sec. and in the WP is 3.14 sec. Figure 5.12 depicts the results.
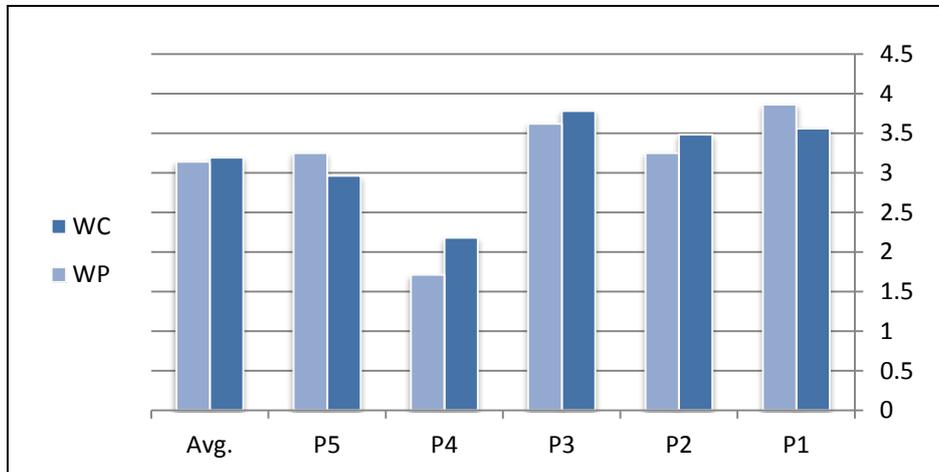
Figure 5.12: The time spent to coping, pasting or cutting text data.

5.1.2.5. Summary

The first experiment shows that the percentage of content in the Web Cube and the webpage respectively are 83.0% and 21.4%. And the average times spent to reach goal content are 10.77 sec. in the Web Cube, 28.06 sec. in the Web Cube with hand gesture, and 32.24 sec. in the webpage. Finally, the average number of errors occurred are 0.6 in the Web Cube, 3.4 in Web Cube with hand gesture and 2.6 in the webpage. With the exception of

40

number of errors in the Web Cube with hand gesture that can be reduced by practice, the Web Cube is more visible than webpage, which prove the sub-hypothesis "Web Cube content is more visible than webpage content".

The second experiment shows that the average number of attempts to find a particular content in the Web Cube, the Web Cube with hand gesture, and the webpage respectively are 2 within 10.77 sec., 3 within 19.01, and 2 within 18.58 sec. And the average number of attempts to send a note is 1 within 30.45 sec. in the Web Cube and 1 within 29.77 sec. in the webpage. And the average number of errors occurred to check the calendar is 0.6 within 9.59 sec. in the Web Cube, 7.2 within 98.21 sec. in Web Cube with hand gesture, and no errors occurred within 8.3 sec. in the webpage. With the exception of number of errors in the Web Cube with hand gesture that can be reduced by practice, this experiment proves the sub-hypothesis "Web Cube navigation is easier than webpage navigation".

The third experiment shows that the number of manipulation forms in the Web Cube and the webpage respectively are 5 to 1. And the average number of errors occurred is 1. 0 in the Web Cube, more than ten in the Web Cube with hand gesture, and no errors occurred in the webpage. With the exception of number of errors in the Web Cube with hand gesture that can be reduced by practice, this experiment proves the sub-hypothesis "Object manipulation (drag, rotate, etc.) in Web Cube easier than in webpages".

The fourth experiment shows that the average time taken to enter data in a data filed in the Web Cube and webpage respectively are 2.22 and 1.92 with 0.3 as difference for the webpage. And the average times spent to copy, paste, or cut are 3.19 sec. in the Web Cube, and 3.14 sec. with 0.05 as difference for the webpage. Finally, no errors occurred in both sites. With no significant differences in between both sites, this experiment proves the sub-hypothesis "Data manipulation in Web Cube is as easy as in webpage". Therefore, the proof of the previous four sub-hypotheses leads us to proof the main hypothesis "Web Cube improves user's web interaction experience".

5.2.   Questionnaire

As mentioned earlier the purpose of this evaluation is to measure user satisfaction. This evaluation comprises a validated 15-item paper-based questionnaire in which users score each item using Likert scale the commonly scale involved in research that employs questionnaires. It is the most widely used approach to scaling responses in survey research (1 is strongly disagree, 2 is disagree, 3 is neutral, 4 is agree and 5 is strongly agree).

5.2.1.   Questions

The questionnaire is designed to measure scales of ease of learning, control, memorable, Efficiency of use, errors, and subjective (user) satisfaction.

1.   *Ease of learning:*   How fast can a user who has never seen the UI before learn it sufficiently well to perform simple interactions?

2.   *Control:*   The feeling that the UI responds to user interaction in a consistent way and that its workings can easily be internalized.

3.   *Memorable:*   If a user has used the UI at some earlier date, can that user remember enough to use it more effectively next time, or does the user have to start over again, learning everything every-time?

4.   *Efficiency of use:*   Once an experienced user has learned to use the UI, how fast can interactions are performed?

5.   *Errors:*   How often do users make mistakes when using the UI, and how serious are these errors.

6.   *Subjective satisfaction:*   How much does the user like using the UI? What is the level of comfort and acceptability of the UI users and to other people affected by its use?

Questionnaire depicted in Figure 5.13 is posted online and submitted electronically. This can significantly reduce the time involved in administering and analyzing the questionnaire.

Web Cube Survey

Name: …………… …………….   Specialization: …………… …………….   Date: …………………

Site: http://www. 3dwebcube. com/rrasheed

Introduction

We need to form a clear idea of a new web browsing model called "Web Cube". This questionnaire represents to help us in identifying the key points of user interaction and satisfaction using the new proposed model. Please kindly answer the following questions so that we can evaluate the new proposed model. This evaluation heavily depends on your answers. We will be thankful to you for your time to answer questions. We would like you to answer these questions to the best of your ability. If you do not know the answer, please say so.

Criteria: (1 = strongly disagree, 2 = disagree, 3 = neutral, 4 = agree and 5 = strongly agree).

		1	2	3	4	5
1	Do you think that you would like to use this UI frequently?					
2	Do you think that, in its existing form, this UI is easy enough?					
3	Do you think this UI is easy to interact?					
4	Do you think that you would need the support of a technical person to be able to use this UI?					
5	Did you find the various interactions in this UI were well integrated?					
6	Did you detect much inconsistency in this UI?					
7	Would you imagine that most people could learn this UI quickly?					
8	Did you find this UI difficult to use?					
9	Do you need to know lot of things before you could get going with this UI?					
10	Did you feel very confident using this UI?					
11	Do you think that you can dispense conventional web interface using this UI?					
12	Do you think that you can dispense conventional input devices such as mouse using this UI?					
13	Did you find that contents of this UI are easy to manipulate?					
14	Do you think that this kind of interaction will be accepted by people?					
15	Do you think that hand gesture as input interface will be accepted by people?					

Additional comments:

………
………
………
………

Figure 5.13: Web Cube questionnaire used for evaluation.

See Appendix D for questionnaires designed in English and Arabic languages.

5.2.2. Questionnaire Results

Table 5.6: The results of questionnaire.

#	Questions	1	2	3	4	5
1	Do you think that you would like to use this UI frequently?	0	1	2	17	31
2	Do you think that, in its existing form, this UI is easy enough?	1	0	13	23	14
3	Do you think this UI is easy to interact?	0	2	8	23	18
4	Do you think that you would need the support of a technical person to be able to use this UI?	17	23	7	3	1
5	Did you find the various interactions in this UI were well integrated?	0	1	12	31	7
6	Did you detect much inconsistency in this UI?	15	23	11	1	1
7	Would you imagine that most people could learn this UI quickly?	0	3	17	21	10
8	Did you find this UI difficult to use?	18	20	9	3	1
9	Do you need to know lot of things before you could get going with this UI?	20	19	7	4	1
10	Did you feel very confident using this UI?	0	0	10	19	22
11	Do you think that you can dispense conventional web interface using this UI?	4	4	12	17	14
12	Do you think that you can dispense conventional input devices such as mouse using this UI?	4	8	15	10	14
13	Did you find that contents of this UI are easy to manipulate?	1	1	9	20	20
14	Do you think that this kind of interaction will be accepted by people?	0	0	17	25	9
15	Do you think that hand gesture as input interface will be accepted by people?	0	3	12	19	17

The results have been collected from participant's responses. The questionnaire was filled mostly by information technology specialists. During five days, fifty one participants submitted the questionnaire. Table 5.6 lists the participant's responses.

5.3. Evaluation Summary

In this section we will summarize and discuss the results of the two evaluation methods while the integration of both results leads to get a comprehensive evaluation. As we mentioned previously, the experimental evaluation used to evaluate the user performance and the questionnaire used to evaluate the user satisfaction.

Two factors coerce us to use the Web Cube versus basic conventional webpage with no additional tools such as Adobe Flash, Ajax, JQuery or other used. As best we know, there is no tool presents a true 3-D environment as Web Cube can do, and the other one is that Web Cube still in a prototype model and does not fully implemented yet to compare with fully implemented tool. By the way, these tools can be used in the Web Cube too.

In most experiments results we found that the participants performance when using Web Cube with conventional input devices was better than participant performance when using

hand gesture input interface. This difference in performance was significant according to many factors, first the lack of the participants experience, second the use of heavy calculating algorithm that delayed the use of hand, third the webcam quality and the surrounding environment of the participants which affect the accuracy of the webcam.
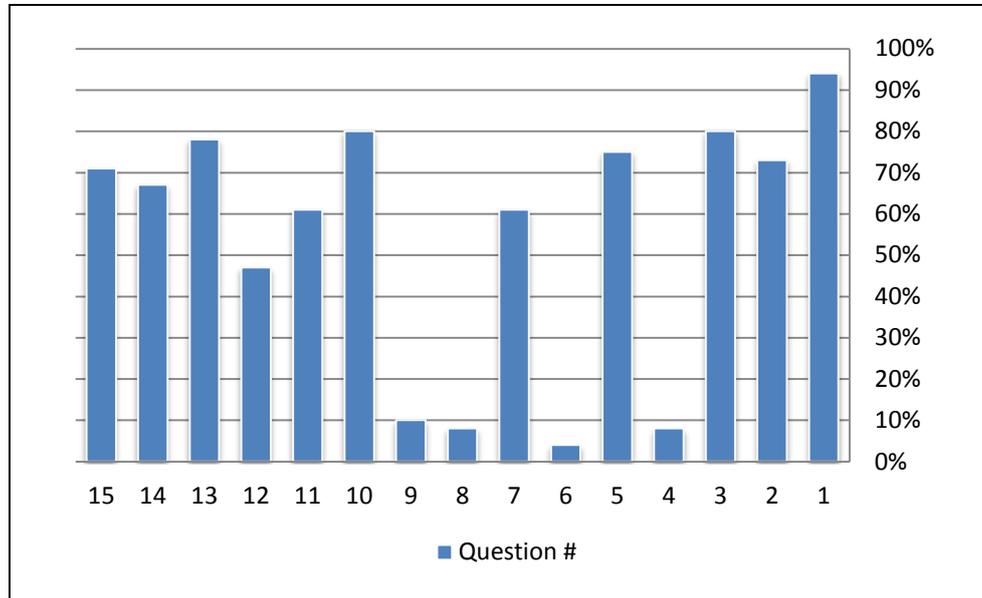
Figure 5.14: Questionnaire results for total of agree and strongly agree score.

From the questionnaire results and in order of total responses agree (4) with strongly agree (5) for each question Figure 5.14 depicts that:

1. More than 94% of the participants will use the Web Cube frequently.

2. More than 72% believe that Web Cube is easy enough in the existing form.

3. More than 80% believe that Web Cube is easy to interact.

4. Less than 8% need technical support to be able to use the Web Cube.

5. More than 75% believe that various interactions in the Web Cube were well integrated.

6. Less than 4% detect much inconsistency in the Web Cube.

7. More than 61% believe that most people could learn this Web Cube quickly.

8. Less than 8% find that Web Cube is difficult to use.

9. Less than 10% need to know lot of things before they can use the Web Cube.

45

10. More than 80% feel very confident using the Web Cube.

11. More than 60% believe that they can dispense conventional web interface using the Web Cube.

12. More than 47% believe that they can dispense conventional input devices such as mouse using the Web Cube.

13. More than 78% find that contents of the Web Cube are easy to manipulate.

14. More than 67% find that this kind of interaction will be accepted by people.

15. More than 70% find that hand gesture as input interface will be accepted by people.

Table 5.7 shows the averages of the questionnaire results where 94% of the participants will use the Web Cube frequently, 80% of them can use it and interact with it easily, 78% of them believe that most people quickly cloud learn it without any technical support, 72% can dispense conventional web and may dispense the conventional input devices too, and 80% of them believe that the Web Cube acceptable by people. Therefore, it is clear that majority of participants satisfied with the Web Cube.

Table 5.7: The averages of the questionnaire results.

Question #	(4+5) %	Average State	
1	94%	4.5	Strongly agree
2	73%	4.0	Agree
3	80%	4.1	Agree
4	8%	2.0	Disagree
5	75%	3.9	Agree
6	4%	2.0	Disagree
7	61%	3.7	Agree
8	8%	2.0	Disagree
9	10%	2.0	Disagree
10	80%	4.2	Agree
11	61%	3.6	Agree
12	47%	3.4	Neutral
13	78%	4.1	Agree
14	67%	3.8	Agree
15	71%	4.0	Agree

Finally, from previous evaluations we conclude that Web Cube was acceptable by majority of web users those use it, and improves the user browsing experience.

46

Chapter 6

Conclusion and Future Work

In this chapter we present the conclusions of our research work, and an outlook on how it can be complemented in future work.

6.1. Conclusion

In this thesis we have presented our research on a newly proposed 3-d web browsing model that we call "Web Cube". We first presented an overview of current web browsing applications, and showed that their primary shortcomings lie in the fact that they do not efficiently combine 3-D virtual reality capabilities with conventional web interaction, and further lack a natural interaction style.

We presented our proposed web browsing model, Web Cube, which combines features of 3-D virtual reality with features of conventional web browsing. We also presented how gestures we incorporated in the model as the primary interaction style hand.

We used two evaluation techniques adopted from HCI to evaluate our newly proposed model. One was an experimental technique to test the efficiency of users in performing sample tasks, and the other was a questionnaire to query users' satisfaction with the new interaction experience.

The experimental technique included four experiments that were administered on 15 subjects and showed that task performance efficiency was higher than conventional interaction styles (keyboard and mouse). Furthermore, results of the questionnaire showed that user satisfaction with the new interaction experience was much higher than with the conventional interaction style.

Thus, the main contribution of this research work is that it introduces a new hand-gesture based 3-D web browsing model, the Web Cube, which achieves higher rates of task performance efficiency and user satisfaction than conventional web browsing styles.

6.2. Future Work

- In this research work only a prototype of the proposed model was implemented, so in a future work we look forward to implementing a more complete model.

- We will also be researching ways to solve some difficulties with hand gesture interaction.

- Success of our proposed Web Cube model encourages us to look for ways to standardize the model and make it supported by prominent web browsers.

- We shall be looking for ways for extending HTML to include support for the Web Cube model.

References

[1] Bardin J. , Lalanda P. , Escoffier C. , and Murphy A. , 2011. Improving user experience by infusing web technologies into desktops. In Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion (SPLASH '11). ACM, New York, NY, USA, 225-236.

[2] Celentano A., and Pittarello F., Observing and adapting user behavior in navigational 3-D interfaces, Proceedings of the working conference on Advanced visual interfaces, 2004, Gallipoli, Italy.

[3] Chen Q., Rahman M., Shen X., El Saddik A. and Georganas N. , Navigating a 3-D virtual environment of learning objects by hand gestures, International Journal of Advanced Media and Communication, Volume 1, Number 4 / 2007, pp 351 – 368.

[4] Chen B., Xu Z., "A framework for browser-based Multiplayer Online Games using WebGL and WebSocket," Multimedia Technology (ICMT), 2011 International Conference , pp. 471-474, 26-28 July 2011.

[5] Chittaro L. and Ranon R., Dynamic generation of personalized VRML content: a general approach and its application to 3-D e-commerce, Proceedings of the seventh international conference on 3-D Web technology, 2002, Tempe, Arizona, USA.

[6] Congote J., Segura A., Kabongo L., Moreno A., Posada J., and Ruiz O., 2011. Interactive visualization of volumetric data with WebGL in real-time. In Proceedings of the 16th International Conference on 3D Web Technology (Web3D '11). ACM, New York, NY, USA, 137-146.

[7] Coors V. and Jung V., Using VRML as an interface to the 3-D data warehouse, Proceedings of the third symposium on Virtual reality modeling language, 1998, Monterey, California, USA.

[8] Dix A., Finlay J., Abowd G., and Beale R., Human Computer Interaction 3rd Edition, 2004

[9] Eno J., Gauch S. and Thompson C., Linking Behavior in a Virtual World Environment, Proceedings of the 15th International Conference on Web 3D Technology, 2010, Los Angeles, USA.

[10] Esnault N., Royan J. Cozot and R., A flexible framework to personalize 3-D Web users experience, Proceedings of the 15th International Conference on Web 3D Technology, 2010, Los Angeles, USA

[11] Grady M., 2010. Functional programming using JavaScript and the HTML5 canvas element. J. Comput. Small Coll. 26, 2 (December 2010), 97-105.

[12] Haj-Rashid M., Usability Evaluation: Applying the Evaluation methods and Techniques to evaluate the Usability of Web Sites, Master's Thesis, Faculty of Information Sciences University of Tampere, 2001

[13] Hongyu Z., Google Earth Software Application in Urban Design of China Proceedings of fourth International Conference on Computer Sciences and Convergence Information Technology, 2009, Seoul, Korea.

[14] HTTP://www.Internetworldstats.com/stats.htm

[15] Jianping Y., Zhang Jie Z., "Towards HTML 5 and interactive 3D graphics," Educational and Information Technology (ICEIT), 2010 International Conference on , vol. 1, no. , pp. V1-522-V1-527, 17-19 Sept. 2010.

[16] Krňoul Z., 2011. Web-based sign language synthesis and animation for on-line assistive technologies. In The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (ASSETS '11). ACM, New York, NY, USA, 307-308.

[17] Lecocq D., Hadwiger M., and Rockwood A., 2010. WebGLot: high-performance visualization in the browser. In ACM SIGGRAPH 2010 Talks (SIGGRAPH '10). ACM, New York, NY, USA, , Article 35 , 1 pages.

[18] Mao G., Wu Y., Hor M. and Tang C., Real-Time Hand Detection and Tracking against Complex Background, Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009, Kyoto, Japan.

[19] Marcus C., Alencar R., Alberto B. Raposo, and Simone D. J. Barbosa. 2011. Composition of HCI evaluation methods for hybrid virtual environments. In Proceedings of the 2011 ACM Symposium on Applied Computing (SAC '11). ACM, New York, NY, USA, 1237-1244.

[20] McCabe H., Interactive 2D and 3-D Graphics over the World-Wide Web with ActionScript, ITB Journal, Issue 10, December 2004, pp. 4-24.

[21] Mistry P. and Maes P. 2009. SixthSense: a wearable gestural interface. In ACM SIGGRAPH ASIA 2009 Sketches (SIGGRAPH ASIA '09). ACM, New York, NY, USA, , Article 11 , 1 pages.

[22] Nebeling M., Matulic F., Streit L., and Norrie M., 2011. Adaptive layout template for effective web content presentation in large-screen contexts. In Proceedings of the 11th ACM symposium on Document engineering (DocEng '11). ACM, New York, NY, USA, 219-228.

[23] Ortiz S., "Is 3D Finally Ready for the Web?," International Journal of Machine Learning and Computing, Vol. 1, No. 3, August 2011, vol. 43, no. 1, pp. 14-16, Jan. 2010.

[24] Pfeiffer S., and Parker C., Accessibility for the HTML5 <video> element. In Proceedings of the 2009 International Cross-Disciplinary Conference on Web Accessibility (W4A) (W4A '09). ACM, New York, NY, USA, 98-100.

[25] Pohja M., Honkala M., Penttinen M., Vuorimaa P., and Ervamaa P, Web User Interaction – Comparison of Declarative Approaches, m Proc. of the 2nd International Conference on Web Information Systems and Technologies (WEBIST 2006), 2006, Setbal, Portugal, pp 295-302.

[26] Qureshi M, Aziz A., Saeed, M., Hayat M., and Rasool J. , "Implementation of an efficient algorithm for Human hand gesture identification," Electronics, Communications and Photonics Conference (SIECPC), 2011 Saudi International, pp. 1-5, 24-26 April 2011

[27] Quy P., Lee J., Kim J., Kim J. and Kim H., Collaborative Experiment and Education based on Networked Virtual Reality. Proceedings of fourth International Conference on Computer Sciences and Convergence Information Technology, 2009, Seoul, Korea.

[28] Sato Y. and Saito M., Koik H., Real-Time Input of 3-D Pose and Gestures of a User's Hand and Its Applications for HCI, Proceedings IEEE Virtual Reality Conference , 2001, Yokohama, Japan.

[29] Sons K., Klein F., Rubinstein D., Byelozyorov S., and Slusallek P., XML3D – Interactive 3D Graphics for the Web, : interactive 3d graphics for the web. In Web3D '10: Proceedings of the 15th International Conference on Web 3D Technology (New York, NY, USA, 2010), ACM, pp. 175–184.

[30] Stergiopoulou E., and Papamarkos N., "Hand gesture recognition using a neural network shape fitting technique," Eng. Applic. Artif. Intell. 8, 1141–1158 (2009). —No Citation Hand gesture recognition using a neural network shape fitting technique

[31] Svoboda, P., Karner, W., Rupp, M., Traffic Analysis and Modeling for World of Warcraft, Communications, 2007. ICC '07. IEEE International Conference, pp. 1612-1617, 24-28 June 2007, Glasgow, Scotland.

[32] Thorne C., Exploiting an evolutionary accident in web3-D communications to integrate application components. ACM SIGGRAPH, 2005, New York, USA.

[33] Vani V. and Mohan S., Interactive 3-D Class Room – A Framework for Web3-D using J3-D and JMF, Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India, 2010, India

[34] Viola P. and Jones M., Robust Real-time Object Detection, IEEE Int. Journal of Computer Vision, 2002.

[35] W3C, HTML5 Specification. http://www.w3.org/TR/html5/

[36] W3C, Cascading Style Sheets. http://www.w3.org/TR/CSS/

[37] Wang R. and J. Popovic, Real-Time Hand-Tracking with a Color Glove, ACM Transaction on Graphics (SIGGRAPH 2009), 2009.

[38] Yeh C., Jiang S., Bai J., Liou J., Yeh R., Wang S., and Sung P., Vision-Based Virtual Control Mechanism via Hand Gesture Recognition, Journal of Computers Vol.21, No.2 : Special issue on "Advances in Multimedia", pp 55-66, 2010

51

Appendices

Appendix A: Web Cube and Its Components Code.

Figure A.1 depicts the code that creates the Web Cube.

```
function RSR_webCube(width, height, depth, cameraPosition){
camera = new THREE. Camera( 0/*fov*/, window. innerWidth / window. innerHeight, 1, 10000 );
camera. position. set( 0, 0, cameraPosition);
camera. target. position. x = 0;camera. target. position. y = 0;camera. target. position. z =-depth;
var RSR_webCubeMaterial = new THREE. MeshBasicMaterial({color: 0xFFFFFF, opacity: 1. 0, wireframe:
true});
var RSR_webCubeBackFaceTextureMaterial = THREE. ImageUtils. loadTexture("rrasheedwall. jpg");
var RSR_webCubeBackFaceMaterial = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_webCubeBackFaceTextureMaterial, opacity: 1. 0 } );
RSR_webCubeBackFace = new THREE. Mesh( new THREE. PlaneGeometry( width, height, 1, 1),
RSR_webCubeBackFaceMaterial );
RSR_webCubeBackFace. position. x = 0;RSR_webCubeBackFace. position. y =
0;RSR_webCubeBackFace. position. z = -depth;
RSR_webCubeBackFace. rotation. x = 0;RSR_webCubeBackFace. rotation. y =
0;RSR_webCubeBackFace. rotation. z = 0;
scene. addObject( RSR_webCubeBackFace );
var RSR_webCubeRightFaceTextureMaterial = THREE. ImageUtils. loadTexture("rrasheedwall. jpg");
var RSR_webCubeRightFaceMaterial = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_webCubeRightFaceTextureMaterial, opacity: 1. 0 } );
RSR_webCubeRightFace = new THREE. Mesh( new THREE. PlaneGeometry( depth, height, 1, 1),
RSR_webCubeRightFaceMaterial );
RSR_webCubeRightFace. position. x = width / 2;RSR_webCubeRightFace. position. y =
0;RSR_webCubeRightFace. position. z = -depth/2;
RSR_webCubeRightFace. rotation. x = 0;RSR_webCubeRightFace. rotation. y = RSR_radians(-
90);RSR_webCubeRightFace. rotation. z = 0;
scene. addObject( RSR_webCubeRightFace );
RSR_webCubeLeftFace = new THREE. Mesh( new THREE. PlaneGeometry( depth, height, 1, 1),
RSR_webCubeRightFaceMaterial );
RSR_webCubeLeftFace. position. x = -width / 2;RSR_webCubeLeftFace. position. y =
0;RSR_webCubeLeftFace. position. z = -depth/2;
RSR_webCubeLeftFace. rotation. x = 0;RSR_webCubeLeftFace. rotation. y =
RSR_radians(90);RSR_webCubeLeftFace. rotation. z = 0;
scene. addObject( RSR_webCubeLeftFace );
var RSR_webCubeTopFaceTextureMaterial = THREE. ImageUtils. loadTexture("rrasheedceil. jpg");
var RSR_webCubeTopFaceMaterial = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_webCubeTopFaceTextureMaterial, opacity: 1. 0 } );
RSR_webCubeTopFace = new THREE. Mesh( new THREE. PlaneGeometry( width, depth, 1, 1),
RSR_webCubeTopFaceMaterial );
RSR_webCubeTopFace. position. x = 0;RSR_webCubeTopFace. position. y = height /
2;RSR_webCubeTopFace. position. z = -depth/2;
RSR_webCubeTopFace. rotation. x = RSR_radians(90);RSR_webCubeTopFace. rotation. y =
0;RSR_webCubeTopFace. rotation. z = 0;
scene. addObject( RSR_webCubeTopFace );
var RSR_webCubeBottomFaceTextureMaterial = THREE. ImageUtils. loadTexture("floor_wood. jpg");
var RSR_webCubeBottomFaceMaterial = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_webCubeBottomFaceTextureMaterial, opacity: 1. 0 } );
RSR_webCubeBottomFace = new THREE. Mesh( new THREE. PlaneGeometry( width, depth, 1, 1),
RSR_webCubeBottomFaceMaterial );
RSR_webCubeBottomFace. position. x = 0;RSR_webCubeBottomFace. position. y = -height /
2;RSR_webCubeBottomFace. position. z = -depth/2;
RSR_webCubeBottomFace. rotation. x = RSR_radians(-90);RSR_webCubeBottomFace. rotation. y =
0;RSR_webCubeBottomFace. rotation. z = 0;
scene. addObject( RSR_webCubeBottomFace );
}
```

Figure A.1: Web Cube creation code.

Figure A.2 depicts the code that creates the Web Cube image component.

```
function RSR_Image01(ImageURL, width, height, depth, pX, pY, pZ, rX, rY, rZ, alpha){
RSR_Image01Plane = new THREE. Object3D();
//////////////line border
var RSR_windowTextureMaterial = THREE. ImageUtils. loadTexture("RSRwindow. png");
var RSR_windowMaterial = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_windowTextureMaterial, opacity: 1. 7 ,transparent: true} );
//RSR_windowMaterial. map. needsUpdate = true;
var RSR_windowGeometry = new THREE. PlaneGeometry( width+RSR_webCubeWidth*0. 036,
height+RSR_webCubeHeight*0. 057 );//new THREE. CubeGeometry( 150, 80, 5 );
RSR_window = new THREE. Mesh(RSR_windowGeometry,RSR_windowMaterial);
RSR_window. doubleSided = true;
RSR_Image01Plane. addChild(RSR_window);
//////////////
var RSR_Image01TextureMaterial = THREE. ImageUtils. loadTexture(ImageURL);
var RSR_Image01Material = new THREE. MeshBasicMaterial( { color:0xffffff, map:
RSR_Image01TextureMaterial, opacity: 1. 0, transparent: false} );
//RSR_Image01Material. map. needsUpdate = true;
var RSR_Image01Geometry = new THREE. PlaneGeometry( width, height );//new THREE. CubeGeometry(
150, 80, 5 );
RSR_Image01 = new THREE. Mesh(RSR_Image01Geometry,RSR_Image01Material);
RSR_Image01. position. set(0,0,0. 5);
RSR_Image01Plane. addChild(RSR_Image01);
/////////////// Border
var RSR_Image01BorderTextureMaterial = THREE. ImageUtils. loadTexture("back05. jpg");
var RSR_Image01BorderMaterial = new THREE. MeshLambertMaterial( { color:0x202020, opacity: 1. 0,
transparent: false } );
var RSR_Image01TopBorderGeometry = new THREE. CubeGeometry( width+10, 5, depth );//new THREE.
PlaneGeometry( 150, 80, 4, 4 );//
var RSR_Image01LeftBorderGeometry = new THREE. CubeGeometry( 5, height+0, depth );//new THREE.
PlaneGeometry( 150, 80, 4, 4 );//
RSR_Image01TopBorder = new THREE.
Mesh(RSR_Image01TopBorderGeometry,RSR_Image01BorderMaterial);
RSR_Image01TopBorder. position. set(0,((height+5)/2),0);
RSR_Image01Plane. addChild(RSR_Image01TopBorder);
RSR_Image01BottomBorder = new THREE.
Mesh(RSR_Image01TopBorderGeometry,RSR_Image01BorderMaterial);
RSR_Image01BottomBorder. position. set(0,-((height+5)/2),0);
RSR_Image01Plane. addChild(RSR_Image01BottomBorder);
RSR_Image01LeftBorder = new THREE.
Mesh(RSR_Image01LeftBorderGeometry,RSR_Image01BorderMaterial);
RSR_Image01LeftBorder. position. set(-((width+5)/2),0,0);
RSR_Image01Plane. addChild(RSR_Image01LeftBorder);
RSR_Image01BottomBorder = new THREE.
Mesh(RSR_Image01LeftBorderGeometry,RSR_Image01BorderMaterial);
RSR_Image01BottomBorder. position. set(((width+5)/2),0,0);
RSR_Image01Plane. addChild(RSR_Image01BottomBorder);
///////////Border
return RSR_Image01Plane;
                          }
```

Figure A.2: Web Cube Image component creation code.

Figure A.3 depicts the code that creates the Web Cube 3-D text or label.

```
function RSR_text3D(text,size, width, height, pX, pY, pZ, rX, rY, rZ,color){
var curveSegments = 6,
bezelThickness = 2,
bezelSize = 1. 5,
bezelEnabled = true,
font = "helvetiker", // helvetiker, optimer, gentilis, droid sans, droid serif
weight = "bold",          // normal bold
style = "normal";          // normal italic
textGeo = new THREE. TextGeometry( text, {
size: size,
height: height,
curveSegments: curveSegments,
font: font,
weight: weight,
style: style,
bezelThickness: bezelThickness,
bezelSize: bezelSize,
bezelEnabled: bezelEnabled
});
var textMaterial = new THREE. MeshLambertMaterial( { color: color, wireframe: false ,opacity: 1. 0  } );
var RSR_text3D = new THREE. Mesh( textGeo, textMaterial );
return RSR_text3D;
}
```

Figure A.3: Web Cube 3-D text creation code.

Figure A.4 depicts the code that creates the Web Cube 2-D text or label.

```
function RSR_text2D(text,txtSize, width, height, pX, pY, pZ, rX, rY, rZ){
var RSR_textCanvas = document. createElement( "canvas" );
RSR_textCanvas. id="text2DCanvas";
var RSR_textCanvasContext = RSR_textCanvas. getContext("2d");
RSR_textCanvas. width = width;
RSR_textCanvas. height = height;
RSR_textCanvasContext. fillStyle = "#000";
RSR_textCanvasContext. fillRect(0,0,width,height);
RSR_textCanvasContext. fillStyle = "#FFFFFF";
RSR_textCanvasContext. font = txtSize+"pt arial bold";
RSR_textCanvasContext. fillText(text, 1, height - 10);
}
var RSR_textMaterial = new THREE. MeshBasicMaterial( { map: new THREE. Texture( RSR_textCanvas ) }
);
RSR_textMaterial. map. needsUpdate = true;
RSR_text = new THREE. Mesh( new THREE. PlaneGeometry( width, height ), RSR_textMaterial );
RSR_text. position. set(pX, pY, pZ);
RSR_text. rotation. set(rX, rY, rZ);
RSR_text. doubleSided = true;
RSR_text. updateMatrix();
return RSR_text;
}
```

Figure A.4: Web Cube 2-D text creation code.

Figure A.5 (a) and (b) depicts the code that detects hand gestures.

```
void detect_and_draw( IplImage* img ){
static CvMemStorage* storage = 0;
static CvHaarClassifierCascade* cascade_handsup = 0;
static CvHaarClassifierCascade* cascade_handsdown = 0;
int scale = 1;
CvPoint pt1, pt2;
int i;
CvSeq* handsUp = 0;
CvSeq* handsDown = 0;
gerakan=motionDet(img);
cascade_handsup = (CvHaarClassifierCascade*)cvLoad( cascade_namehandsup, 0, 0, 0 );
cascade_handsdown = (CvHaarClassifierCascade*)cvLoad( cascade_namehandsdown, 0, 0, 0 );
if( !cascade_handsup ){
fprintf( stderr, "ERROR: Could not load classifier cascade\n" );
return;}
storage = cvCreateMemStorage(0);
cvNamedWindow( "result", 1 );
cvClearMemStorage( storage );
handsUp = cvHaarDetectObjects( img, cascade_handsup, storage, 1. 2, -1,
CV_HAAR_DO_CANNY_PRUNING, cvSize(40,40));//40, 40) );
if(gerakan > gerakanYgTerjadi)//jika ad gerakan,selalu update r{
if(handsUp->total>=1){
numberOfObjects++;
for( i = 0; i < 1; i++){//(hands ? hands->total : 0); i++ ){
CvRect* r = (CvRect*)cvGetSeqElem( handsUp,i );
if(!firstMouseWidth) firstMouseWidth = r->width;
EX = r->x; YE = r->y; WID = r->width; HEI = r->height;
pt1. x = r->x*scale;
pt2. x = (r->x+r->width)*scale;
pt1. y = r->y*scale;
pt2. y = (r->y+r->height)*scale;
// Moving the Mouse Pointer
mouseWidth = r->width;
if(mouseWidth > firstMouseWidth+20){//- oldMouseWidth) > 10){
printf("Mouse Forward\n");
GenerateKey(VK_UP, FALSE);
}else if(mouseWidth < firstMouseWidth-20){ //- oldMouseWidth) < -10){
printf("Mouse Backward\n");
GenerateKey(VK_DOWN, FALSE);
}else {
printf("Mouse Same Depth\n");
}
oldMouseWidth = mouseWidth;
mousePos. x=r->x * double( 1680/ 320)*1. 75;//pt1. x+r->width/2;//(pt1. x+(pt2. x-pt1. x)/2);//
mousePos. y=r->y * double( 1050/ 240)*2;//pt1. y+r->width/2;//(pt1. y+(pt2. y-pt1. y)/2);
CvPoint mousePointer1,mousePointer2;
mousePointer1. x = mousePos. x - 2;
mousePointer1. y = mousePos. y - 2;
mousePointer2. x = mousePos. x + 2;
mousePointer2. y = mousePos. y + 2;
cvRectangle( img, pt1, pt2, CV_RGB(255,0,0), 1, 8, 0 );
cvRectangle( img, mousePointer1, mousePointer2, CV_RGB(230,20,32), 4, 8, 0 );
RSR_MOUSE_UP_COUNT++;
if(!RSR_MOUSE_UP){
if(RSR_MOUSE_UP_COUNT>15){
printf("Mouse Up. . %d\n",RSR_MOUSE_UP_COUNT);
RSR_MOUSE_UP_COUNT=0;
RSR_MOUSE_UP = 1;
RSR_MOUSE_DOWN = 0;
mouseLeftUp();
}}
```

Figure A.5 (a): Hand gesture detection code (Part 1).

56

```
if(!inSameArea(mousePos,oldMousePos)){
time(&startTime);
MetainPointer(img);
if(oldMousePos. x+500<=mousePos. x){
GenerateKey(VK_RIGHT, FALSE);
}else if(oldMousePos. x-500>=mousePos. x){
GenerateKey(VK_LEFT, FALSE);
}
mousePos. y*(1200/240));
oldMousePos = mousePos;
}
CV_RGB(30,20,232), 1, 8, 0 );
}
}else{
handsDown = cvHaarDetectObjects( img, cascade_handsdown, storage, 1. 2, -1,
CV_HAAR_DO_CANNY_PRUNING, cvSize(40,40));//40, 40) );
if(handsDown->total>=1){
numberOfObjects++;
for( i = 0; i < 1; i++){//(hands ? hands->total : 0); i++ ){
CvRect* k = (CvRect*)cvGetSeqElem( handsDown,i );
EX2 = k->x;       YE2 = k->y;      WID2 = k->width; HEI2 = k->height;
pt1. x = k->x*scale;
pt2. x = (k->x+k->width)*scale;
pt1. y = k->y*scale;
pt2. y = (k->y+k->height)*scale;
mouseWidth = k->width;
oldMouseWidth = mouseWidth;
mousePos. x=k->x * double( 1680/ 320)*1. 75;//pt1. x+r->width/2;//(pt1. x+(pt2. x-pt1. x)/2);//
mousePos. y=k->y * double( 1050/ 240)*2;//pt1. y+r->width/2;//(pt1. y+(pt2. y-pt1. y)/2);
CvPoint mousePointer1,mousePointer2;
mousePointer1. x = mousePos. x - 2;
mousePointer1. y = mousePos. y - 2;
mousePointer2. x = mousePos. x + 2;
mousePointer2. y = mousePos. y + 2;
cvRectangle( img, pt1, pt2, CV_RGB(0,255,0), 1, 8, 0 );
cvRectangle( img, mousePointer1, mousePointer2, CV_RGB(230,20,32), 4, 8, 0 );
if(!inSameArea(mousePos,oldMousePos)){
time(&startTime);
oldMousePos = mousePos;
}
RSR_MOUSE_DOWN_COUNT++;
if(!RSR_MOUSE_DOWN){
if(RSR_MOUSE_DOWN_COUNT>2){
printf("Mouse Down. . %d\n",RSR_MOUSE_DOWN_COUNT);
RSR_MOUSE_DOWN_COUNT=0;
RSR_MOUSE_UP = 0;
RSR_MOUSE_DOWN = 1;
mouseLeftDown();
}
}
}
}
}
}else if(gerakan <= gerakanYgTerjadi && handsUp->total > 0){//tidak ada gerakan
MetainPointer(img);
}
cvShowImage( "result", img );//cvMoveWindow("result",430,350);
}
```

Figure A.5 (b): Hand gesture detection code (Part 2).

Figure A.6 depicts the code handle the interaction operations.

```
function RSR_keydown(RSR_event){
switch(RSR_event. keyCode){
case 37:if (RSR_albumSelected){
RSR_albumCounter = ( RSR_albumCounter % 6 ) + 1 ;
RSR_albumMoveNext(RSR_albumCounter); }
break;//Left Arrow
case 38:
if(camera. position. z>-500) camera. position. z -=10;
break;//UP
case 39:if (RSR_albumSelected){
RSR_albumCounter = ( RSR_albumCounter % 6 ) - 1 ;
RSR_albumMoveNext(RSR_albumCounter); }
break; //Right
case 40 :if(camera. position. z<1100) camera. position. z +=10;
break;//Down
case 65: camera. target. position. x -= 3; break; //a
case 68: camera. target. position. x += 3; break; //d
case 87: camera. target. position. y += 3; break; //w
case 83: camera. target. position. y -= 3; break; //s
case 82: camera. target. position. set(0,0,0);    //r
camera. position. set(0,0,window. innerHeight);break;
case 17: //Ctrl
var RSR_mouseX = mouse. x*100;//(((mouse. x+1)/2)*window. innerWidth);// - (window. innerWidth/2);
var RSR_mouseY = mouse. y*100;//(((mouse. y-1)/2)*window. innerHeight);// - (window. innerHeight/2);
camera. target. position. set(RSR_mouseX,RSR_mouseY,0);
break;
case 16: //Sheft
var RSR_mouseX = mouse. x*100;//(((mouse. x+1)/2)*window. innerWidth);// - (window. innerWidth/2);
var RSR_mouseY = mouse. y*100;//(((mouse. y-1)/2)*window. innerHeight);// - (window. innerHeight/2);
camera. position. set(RSR_mouseX,RSR_mouseY,window. innerHeight - 20);
break;
```

Figure A.6: Interaction operation handling code.

58

Appendix B: Haar-Like Feature and Voila and Jones algorithm.

Hand-based Human Computer Interface (HCI) should meet the requirements of real-time, accuracy and robustness. The purpose of Haar-like features is to meet the real-time requirement. The purpose of the cascade of Adaptive boost (AdaBoost) classifiers is to achieve both accuracy and speed. The algorithm has been used for face detection which achieved high detection accuracy and approximately 15 times faster than any previous approaches. The algorithm is a generic objects detection/recognition method.

Each Haar-like feature consists of two or three jointed "black" and "white" rectangles shown in Figure B.1:
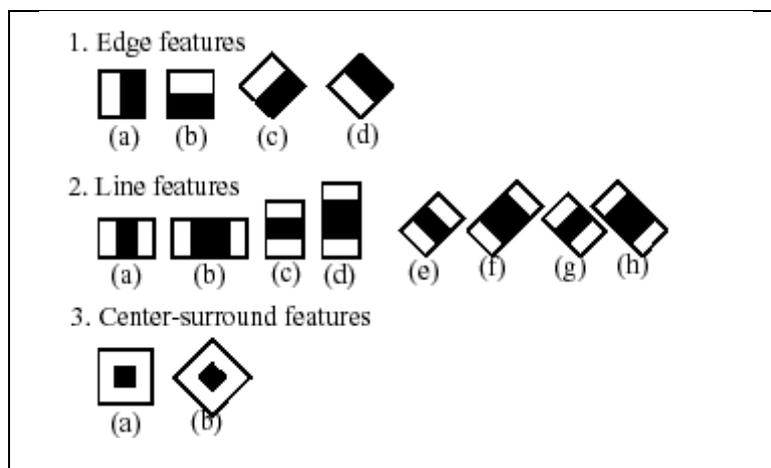
Figure B.1: Examples of haar-like features.

The value of a Haar-like feature is the difference between the sum of the pixel gray level values within the black and white rectangular regions:

$f(x)$=Sum black rectangle (pixel gray level) – Sum white rectangle (pixel gray level)

Compared with raw pixel values, Haar-like features can reduce/increase the in-class/out-of-class variability, and thus making classification easier.

The rectangle Haar-like features can be computed rapidly using "integral image". Integral image at location of x, y contains the sum of the pixel values above and left of x, y, inclusive as Figure B.2 depicts:

59
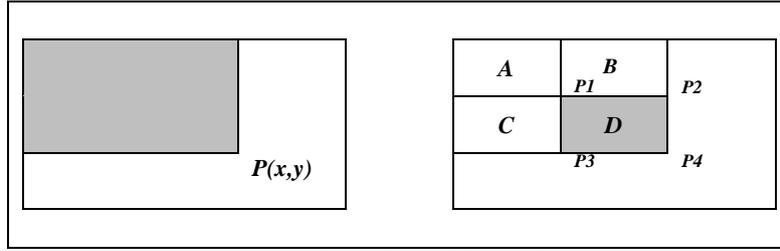
Figure B.2: The integral image [34].

$$P(x, y) = \sum_{x' \le x, y' \le y} i(x', y')$$

The sum of pixel values within "D":

$$P_1 = A, P_2 = A + B, P_3 = A + C, P_4 = A + B + C + D$$
$$P_1 + P_4 - P_2 - P_3 = A + A + B + C + D - A - B - A - C = D$$

To detect the hand, the image is scanned by a sub-window containing a Haar-like feature. Based on each Haar-like feature fj a weak classifier hj(x) is defined as:

$$h_j(x) = \begin{cases} 1 & if \quad p_j f_j(x) < p_j \theta_j \\ 0 & otherwise \end{cases}$$

Where x is a sub-window and θ is a threshold. pj indicating the direction of the inequality sign.

The computation cost using Haar-like features:

Example: original image size: 320X240, sub-window size: 24X24, frame rate: 15 fps, the total number of sub-windows with one Haar-like feature per second:

$$(320-24+1) \times (240-24+1) \times 15 = 966,735$$

Considering the scaling factor and the total number of Haar-like features, the computation cost is huge. AdaBoost is an iterative learning algorithm to construct a "strong" classifier using only a training set and a "weak" learning algorithm. A "weak" classifier with the minimum classification error is selected by the learning algorithm at each iteration. AdaBoost is adaptive in the sense that later classifiers are tuned up in favor of those sub-windows misclassified by previous classifiers.

Adaboost starts with a uniform distribution of "weights" over training examples. The weights tell the learning algorithm the importance of the example.

- Obtain a weak classifier from the weak learning algorithm, hj(x).

- Increase the weights on the training examples that were misclassified.

- (Repeat)

- At the end, carefully make a linear combination of the weak classifiers obtained at all iterations.

$$f_{\text{final}}(\mathbf{x}) = \alpha_{final,1} h_1(\mathbf{x}) + \ldots + \alpha_{final,n} h_n(\mathbf{x})$$

A series of classifiers are applied to every sub-window. The first classifier eliminates a large number of negative sub-windows and pass almost all positive sub-windows (high false positive rate) with very little processing. Subsequent layers eliminate additional negatives sub-windows (passed by the first classifier) but require more computation. After several stages of processing the number of negative sub-windows has been reduced radically as Figure B.3 depicts.
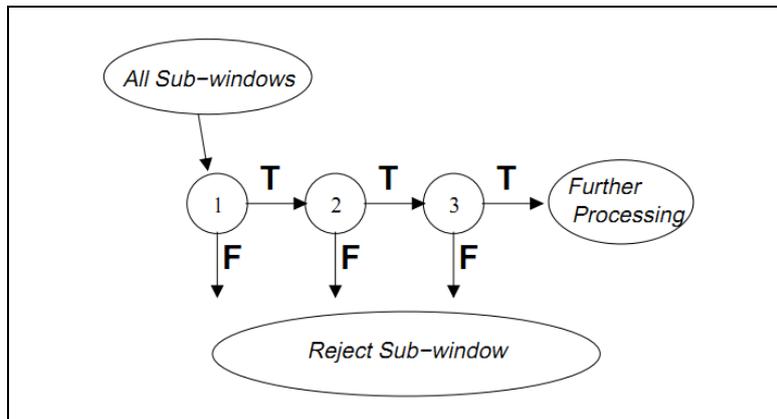
Figure B.3: Cascade Classifiers [34].

Negative samples: non-object images. Negative samples are taken from arbitrary images. These images must not contain object representations.

Positive samples: images contain object (hand in our case). The hand in the positive samples must be marked out for classifier training.

Appendix C: Experiments Forms.

Experiment number 1

Web Cube Experiment		
Name:	Specialization:	Date: ....
Experiment #: ....1........... .	Participant #:	

Web Cube site	Web Page Site
http://www. webcube. com/rrasheed	http://site. iugaza. edu. ps/rrasheed

---

Sub-hypothesis:

Web Cube content is more visible than webpage content.

- *Open the site, how many of content you can see (in percentage) when the site loaded?*

• Web Cube:	• Web Page:

- *Find the syllabus of "Web-Programming" course in the site.*

• Time taken in Web Cube:	• Time taken in Web Page:

- *Number of errors during two minutes of navigation.*

• Number of errors in Web Cube:	• Number of errors in Web Page:

Results:

………
………
………

Additional comments:

………
………
………

| **Web Cube Experiment** |
| Name: .... Specialization: .... Date: .... |
| Experiment #: ....2............. Participant #: .... |

| Web Cube site | Web Page Site |
| http://www. webcube. com/rrasheed | http://site. iugaza. edu. ps/rrasheed |

Sub-hypothesis:

   Data manipulation in Web Cube is as easy as in webpage.

- *Fill the name field at the note or comment form?*
- Time taken in Web Cube: • Time taken in Web Page:
- *Copy the data entered in the name field into the email field.*
- Time taken in Web Cube: • Time taken in Web Page:
- *Number of errors during data entry.*
- Number of errors in Web Cube: • Number of errors in Web Page:

Results:

………………………………………………………………………………………………………
………………………………………………………………………………………………………
………………………………………………………………………………………………

Additional comments:

………………………………………………………………………………………………………
………………………………………………………………………………………………………
………………………………………………………………………………………………

63

Experiment number 3

Web Cube Experiment

Name: .... Specialization: .... Date: ....

Experiment #: ....3............

Participant #: ... .

.

Web Cube site	Web Page Site
http://www. webcube. com/rrasheed	http://site. iugaza. edu. ps/rrasheed

Sub-hypothesis:

   Object manipulation (drag, rotate, etc.) in Web Cube easier than in webpages.

- *Selecting an image and manipulate it.*

- Number of manipulation forms the participant can use in Web Cube:
- Number of manipulation forms the participant can use in Web Page:

- *Number of errors during image manipulation.*

- Number of errors in Web Cube:
- Number of errors in Web Page:

Results:

……

……

……

Additional comments:

……

……

……

Experiment number 4

Web Cube Experiment
Name: ....   Specialization: ....   Date: ....
Experiment #: ....4............   Participant #: ....
.

Web Cube site	Web Page Site
http://www. webcube. com/rrasheed	http://site. iugaza. edu. ps/rrasheed

---

Sub-hypothesis:

Web Cube navigation is easier than webpage navigation.

- *Finding the details of the research titled*

  "Detection of XML Rewriting Attack: Enhance Inline Approach by Element Position"?

- Number of attempts in Web Cube: • Number of attempts in Web Page:

- *Sending note or comment to site owner?*

- Number of attempts in Web Cube: • Number of attempts in Web Page:

- *There is an activity at Feb, 4 2011. Check what day name is that day.*

- Number of errors in Web Cube: • Number of errors in Web Page:

Results:

……

……

……

Additional comments:

……

……

……

Appendix D: Questionnaire in English and Arabic Languages.
English Questionnaire

Name: .... Specialization: .... Date: ....

Site: http://www. webcube. com/rrasheed

Introduction

We need to form a clear idea of a new web browsing model called "Webcube". This questionnaire represents to help us in identifying the key points of user interaction and satisfaction using the new proposed model. Please kindly answer the following questions so that we can evaluate the new proposed model. This evaluation heavily depends on your answers. We will be thankful to you for your time to answer questions. We would like you to answer these questions to the best of your ability. If you do not know the answer, please say so.

Criteria: (1 = strongly disagree, 2 = disagree, 3 = neutral, 4 = agree and 5 = strongly agree).

		1	2	3	4	5
1	Do you think that you would like to use this UI frequently?					
2	Do you think that, in its existing form, this UI is easy enough?					
3	Do you think this UI is easy to interact?					
4	Do you think that you would need the support of a technical person to be able to use this UI?					
5	Did you find the various interactions in this UI were well integrated?					
6	Did you detect much inconsistency in this UI?					
7	Would you imagine that most people could learn this UI quickly?					
8	Did you find this UI difficult to use?					
9	Do you need to know lot of things before you could get going with this UI?					
10	Did you feel very confident using this UI?					
11	Do you think that you can dispense traditional web interface using this UI?					
12	Do you think that you can dispense traditional input devices such as mouse using this UI?					
13	Did you find that contents of this UI are easy to manipulate?					
14	Do you think that this kind of interaction will be accepted by people?					
15	Do you think that hand gesture as input interface will be accepted by people?					

Additional comments:

………
………
………

66

استبانة **Web Cube**

الأسم: .... التخصص: .... التاريخ: ....

الموقع: <u>http://www. 3dwebcube. com/rrasheed</u>

مقدمة

نحن بحاجة لتكوين فكرة واضحة عن نموذج مقترح جديد لتصفح الانترنت والمسمى "Webcube". هذه الاستبانة ستساعدنا في تحديد النقاط الاساسية في تفاعل ورضا المستخدم لهذا المقترح. الرجاء التكرم بالاجابة عن الاسئلة التالية حتى نستطيع تقييم المقترح الجديد، ويعتمد هذا التقييم بشكل كبير على اجاباتك. سنكون شاكرين لك على وقتك للاجابة على الاسئلة. كما نرجوا منك أن تجيب على الأسئلة بموضوعية وحسب رأيك الشخصي فقط. وإذا كنت لا تعرف الجواب، نرجو أن تخبرنا بذلك.

المعايير : (1 = لا أوافق بشدة، 2 = لا أوافق، 3 = متوسط، 4 = أوافق، 5 = أوافق بشدة).

5	4	3	2	1		
					هل تعتقد أنك سترغب في استخدام واجهة المستخدم هذه في المستقبل؟	1
					هل تعتقد أن واجهة المستخدم ـ في شكلها الحالي ـ سهلة بما فيه الكفاية؟	2
					هل تعتقد أنه من السهل التفاعل مع واجهة المستخدم؟	3
					هل تعتقد أنك سوف تحتاج إلى دعم من شخص تقني لتكون قادر على استخدام واجهة المستخدم؟	4
					هل وجدت أن التفاعلات المختلفة في واجهة المستخدم متكاملة بشكل جيد؟	5
					هل وجدت الكثير من التناقض في واجهة المستخدم؟	6
					هل تتصور أن معظم الناس يمكن أن يتعلمو واجهة المستخدم بسرعة؟	7
					هل وجدت صعوبة في استخدام واجهة المستخدم؟	8
					هل أنت بحاجة الى معرفة الكثير لتتعامل مع واجهة المستخدم؟	9
					هل تشعر بالثقة عند استخدام واجهة المستخدم؟	10
					هل تعتقد أنه يمكنك الاستغناء عن واجهة الويب التقليدية باستخدام واجهة المستخدم هذه؟	11
					هل تعتقد أنه يمكنك الاستغناء عن جهاز الإدخال التقليدي مثل الماوس باستخدام واجهة المستخدم هذه؟	12
					هل كان التعامل مع محتوى واجهة المستخدم سهل؟	13
					هل تعتقد أن هذا الأسلوب في التفاعل سيلقى قبولاً عند الناس؟	14
					هل تعتقد أن استخدام حركة اليد كأداة إدخال سيكون مقبولاً عند الناس؟	15

ملاحظات إضافية:

....

....

....