Enhancing the use of online 3d multimedia content through the analysis of user interactions

Thi Phuong Nghiem

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Image, Information et Hypermédia

Présentée et soutenue par :

Mme THI PHUONG NGHIEM

le mercredi 2 juillet 2014

Titre :

ENHANCING THE USE OF ONLINE 3D MULTIMEDIA CONTENT THROUGH THE ANALYSIS OF USER INTERACTIONS

Ecole doctorale :

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

Unité de recherche :

Institut de Recherche en Informatique de Toulouse (I.R.I.T.)

Directeur(s) de Thèse :

M. VINCENT CHARVILLAT

M. ROMULUS GRIGORAS

Rapporteurs :

M. ARNAUD REVEL, UNIVERSITE DE LA ROCHELLE

M. RICHARD CHBEIR, UNIVERSITE DE PAU ET DES PAYS DE L ADOUR

Membre(s) du jury :

M. GILLES GESQUIERES, UNIVERSITE LYON 1, Président

Mme GÉRALDINE MORIN, INP TOULOUSE, Membre

M. VINCENT CHARVILLAT, INP TOULOUSE, Membre

At all the moments, I carry in my heart :
my family

Ministry of Education and Training
of Vietnam for funding my thesis.
USTH for selecting and supporting me.

First of all, I would like to express the most profound gratitude to my supervisors Prof. Vincent CHARVILLAT and Associate Prof. Géraldine MORIN, who have given me valuable advices and feedbacks during this thesis. I have learned many skills and knowledge that are essential for my research work. I would like to thank Dr. Romulus GRIGORAS who instructs me the first steps of the thesis. Many thanks go to Prof. Arnaud REVEL and Prof. Richard CHBEIR for giving me valuable reviews of this thesis. I would like to thank Prof. Gilles GESQUIÈRES for accepting to be the examiner of this thesis and being the president of my PhD defence.

I would also thank my friends – members from the VORTEX group and from the DEVATICS company. We have a memorial period of time working together. I especially thank Axel CARLIER for working with me during the two papers of this thesis.

I would like to thank the Ministry of Education and Training (MOET) of Vietnam for funding my thesis. Many thanks go to the University of Science and Technology of Ha Noi (USTH) for selecting and supporting me to complete this work. I would also thank IRIT-ENSEEIHT for giving me a financial complement as well as funding for my trips to the conferences.

Finally, I would like to express my deep gratitude to my parents who have grown me up. Many thanks go to my sisters and brothers for the material and spiritual supports. Last but not least, I would like to thank my husband and my daughter for being my love and strength.

Title :

Enhancing the Use of Online 3D Multimedia Content through the Analysis of User Interactions

Abstract :

Recent years have seen the development of interactive 3D graphics on the Web. The ability to visualize and manipulate 3D content in real time seems to be the next evolution of the Web for a wide number of application areas such as e-commerce, education and training, architecture design, virtual museums and virtual communities. The use of online 3D graphics in these application domains does not mean to substitute traditional web content of texts, images and videos, but rather acts as a complement for it. The Web is now a platform where hypertext, hypermedia, and 3D graphics are simultaneously available to users. This use of online 3D graphics, however, poses two main issues.

First, since 3D interactions are cumbersome as they provide numerous degrees of freedom, 3D browsing may be inefficient. We tackle this problem by proposing a new paradigm based on crowdsourcing to ease online 3D interactions, that consists of analyzing 3D user interactions to identify Regions of Interest (ROIs), and generating recommendations to subsequent users. The recommendations both reduce 3D browsing time and simplify 3D interactions.

Second, 3D graphics contain purely rich visual information of the concepts. On the other hand, traditional websites mainly contain descriptive information (text) with hyperlinks as navigation means. The problem is that viewing and interacting with the websites that use two very different mediums (hypertext and 3D graphics) may be complicated for users. To address this issue, we propose to use crowdsourcing for building semantic associations between texts and 3D visualizations. The produced links are suggested to upcoming users so that they can readily locate 3D visualization associated with a textual content.

We evaluate the proposed methods with experimental user studies. The evaluations show that the recommendations reduce 3D interaction time. Moreover, the results from the user study showed that our proposed semantic association is appreciated by users, that is, a majority of users assess that recommendations were helpful for them, and browsing 3D objects using both mouse interactions and the proposed links is preferred compared to having only mouse interactions.

Keywords : Interactions, Crowdsourcing, 3D Graphics, Multimedia, the Web, Hypertext.

Titre :

Amélioration de l'utilisation de contenus multimédia 3D en ligne par l'analyse des interactions d'utilisateurs

Résumé :

De plus en plus de contenus 3D interactifs sont disponibles sur la toile. Visualiser et manipuler ces contenu 3D en temps réel, de façon naturelle et intuitive, devient donc une nécessité. Les applications visées sont nombreuses : le e-commerce, l'éducation et la formation en ligne, la conception, ou l'architecture dans le contexte par exemple de musées virtuels ou de communautés virtuelles. L'utilisation de contenus 3D en ligne ne propose pas de remplacer les contenus traditionnels, tels que les textes, les images ou les vidéos, mais plutôt d'utiliser la 3D en complément, pour enrichir ces contenus. La toile est désormais une plate-forme où les contenus hypertexte, hypermédia, et 3D sont simultanément disponibles pour les utilisateurs. Cette utilisation des contenus 3D pose cependant deux questions principales.

Tout d'abord, les interactions 3D sont souvent lourdes puisqu'elles comprennent de nombreux degrés de liberté; la navigation dans les contenus 3D peut s'en trouver inefficace et lente. Nous abordons ce problème en proposant un nouveau paradigme basé sur l'analyse des interactions (*crowdsourcing*). En analysant les interactions d' utilisateurs 3D, nous identifions des régions d'intérêt (ROI), et générons des recommandations pour les utilisateurs suivants. Ces recommandations permettent à la fois de réduire le temps d'interaction pour identifier une ROI d'un objet 3D et également de simplifier les interactions 3D nécessaires.

De plus, les scènes ou objets 3D contiennent une information visuelle riche. Les sites Web traditionnels contiennent, eux, principalement des informations descriptives (textuelles) ainsi que des hyperliens pour permettre la navigation. Des sites contenants d'une part de l'information textuelle, et d'autre part de l'information 3D peuvent s'avérer difficile à appréhender pour les utilisateurs. Pour permettre une navigation cohérente entre les information 3D et textuelles, nous proposons d'utiliser le *crowdsourcing* pour la construction d' associations sémantiques entre le texte et la visualisation en 3D. Les liens produits sont proposés aux utilisateurs suivants pour naviguer facilement vers un point de vue d'un objet 3D associé à un contenu textuel.

Nous évaluons ces deux méthodes par des études expérimentales. Les évaluations montrent que les recommandations réduisent le temps d'interaction 3D. En outre, les utilisateurs apprécient l'association sémantique proposée, c'est-à-dire, une majorité d'utilisateurs indique que les recommandations ont été utiles pour eux, et préfèrent la navigation en 3D proposée qui consiste à utiliser les liens sémantiques ainsi que la souris par rapport à des interactions utilisant seulement la souris.

Mots-clés : Interactions, Crowdsourcing, Graphiques 3D, Multimédia, le Web, Hypertexte.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Introduction

The Web evolved from a read-only text-based system to the currently rich and interactive Web that supports 2D graphics, audio, and videos. This evolution, however, continues. Tim Berners-Lee, the founder of the Web, indicated in his book "Weaving the Web", 1999 [30] that the dream behind the Web is to create a space where information of various kinds can be linked and accessed by everyone.

Ivan Sutherland, the domain expert of Interactive Computer Graphics, Graphical User Interfaces, and Human-Computer Interaction, in his work "Ultimate Display", 1965 [82], showed his vision of 3D experience. He proposed that interactive 3D graphics would be the ultimate achievement of visual media, in which information is expressed through 3D visualization in order to reproduce the real world objects or to represent the imaginary worlds.

Behr et al., 2009 [28] presented that right after the first 2D HTML pages went online, people were thrilled with the idea of having 3D content on the Internet. The ability to visualize and manipulate 3D content in real time seems to be the next evolution of the Web for a wide number of application areas such as e-commerce,

education and training, architecture design, virtual museums and virtual communities. In a sense, this shows that the use of 3D graphics on the Web is a natural trend that merges the developments of interactive 3D graphics and the Web.

This thesis is inspired by the development of interactive 3D graphics on the Web. The goal is to ease the use of online 3D graphics as daily multimedia content. We target the context where users can view and interact with online 3D graphics as simple as with other traditional media such as images and videos. To reach this goal, we studied two main issues of using online 3D graphics.

First, since 3D interactions are cumbersome as they provide numerous degrees of freedom, 3D browsing may be inefficient. For example, new-to-3D users, who may be domain experts, but have marginal knowledge about interaction techniques, can find it difficult to control how to interact with 3D visualizations. It is common for them to face the trouble of "where my object is in the scene"; or sometimes, it may take them too long to access a meaningful 3D region. Inadequate support to user interactions with 3D graphics in these cases may result in users leaving the site.

Second, 3D graphics contain purely rich visual information of the concepts. On the other hand, traditional websites mainly contain descriptive information (text) with hyperlinks as navigation means. The problem is that viewing and interacting with the websites that use two very different mediums (hypertext and 3D graphics) can be complicated for users: they need to handle text browsing to look for general information and text searching for more specific information. On the other hand, they also need to manage how to interact with 3D visualizations (e.g. navigating around the 3D space or inspecting the virtual 3D objects) to gain a better understanding of the data. This separation of interactions between the two modalities (text and 3D graphics) requires too much effort from users to browse the site.

Taking these issues in mind, this thesis introduces novel methods using *Crowdsourcing* to solve the problems, as summarized in the following section.

1.2 Contributions

We propose the use of *Crowdsourcing* to enhance 3D browsing, that is, to reduce 3D browsing time and simplify 3D interactions. Specifically, our contributions are summarized as follows:

- We propose a new paradigm based on crowdsourcing to ease online 3D interactions, that consists of analyzing 3D user interactions to identify Regions of Interest (ROIs), and generating recommendations to subsequent users. The recommendations both reduce 3D browsing time and simplify 3D interactions. We also introduce a simplified 3D user interface that allows users to access the proposed recommendations using simple mouse button clicks.
- We propose a new approach based on crowdsourcing to associate text and 3D for enhancing 3D browsing. Specifically, we propose to use crowdsourcing for building semantic associations between texts and 3D visualizations. The produced links allow users to readily locate 3D visualization associated with a textual content. This enhancement reduces 3D browsing time and improves 3D user experience, that is, users find semantic associations helpful for them, and the enhanced 3D representation with the proposed links are preferable. The use of crowdsourcing provides a natural access to online 3D viewing behaviors, as well as to user opinions and preferences.

1.3 Thesis Outline

The remainder of this dissertation is organized as follows:

- Chapter 2 provides the foundation and motivation of this thesis. We introduce the context we choose to study and present our main research objectives. We begin the chapter with an introduction about the basic concepts of hypertext, and the emergence of the Web that uses hypertext as the key navigation means. Then we give an overview about the evolution of the Web which provides our studied context: the use of 3D interactions as new web navigation model and the emergence of crowdsourcing as a key feature of Web 2.0 to motivate user participation in web content creation and edition. From this context, in the rest of this chapter, we present two main objectives of our research work. The first one is the use of crowdsourcing to create and edit web content. The second

one is the employment of 3D content and 3D interactions to collect informative user feedback. In this thesis, we illustrate these research ideas in e-commerce context.

- Chapter 3 presents our first contribution of this work. We first provide our motivation to ease online interactions with virtual 3D models and review the state-of-the-art methods and related works in the literature. We then describe our proposed pipeline using crowdsourcing to simplify online 3D interactions.
- Chapter 4 presents our second contribution of this work. We first provide our motivation to associate text and 3D for enhancing 3D browsing. Then, we give a review of the state-of-the-art methods and related works in the literature. Finally, we describe our proposed approach using crowdsourcing to build semantic associations between textual contents and 3D visualizations so as to ease 3D browsing.
- Chapter 5 concludes the presented work and discusses perspectives for future work.

Chapter 2

Online 3D Graphics and Web Browsing

Contents

This chapter presents the foundation and motivation of our research work with three main sections. In the first section, we provide the context of this thesis, including the use of 3D interactions as new web navigation model and the employment of crowdsourcing in Web 2.0 to engage users in content creation and edition. In the second section, we give a review about the state-of-the-art usages of crowdsourcing in the websites and provide our first research objective, that is the use of crowdsourcing to create more web content. Finally, in the third section, we present the state-of-the-art 3D web techniques and technologies we use for our research work and provide our second research objective, that is the employment of 3D content and 3D interactions to collect informative user feedback.

2.1 Hypertext and the Web

This section provides the context of this thesis. We first give an overview about the hypertext systems which allow to link different non-linear text documents. Then, we present the concept of the Web that integrates hypertext into the Internet to access and share information. We end the section with a review of the web developments that inspire this research work, including the introduction of new web navigation model – 3D interactions, and the emergence of participatory Web (Web 2.0), where users are allowed to contribute to the edition and creation of new web content.

2.1.1 Hypertext

Hypertext is seen as originating in the Vannevar Bush's article "As We May Think", 1945 [32] with the idea of using cross-references within and across documents to organize and manage the huge amount of available scientific information. Vannevar Bush's proposal was based on the inference that human mind works by association. The author proposed to build a device, namely *memex*, which could store his books and records, and associate them with cross-references so that information could be accessed with better speed and flexibility.

Inspired by Vannevar Bush' vision, Theodor Nelson, in his paper while working on Xanadu, 1965 [72], coined the term "hypertext" to mean a body of written or pictorial material, which is interconnected in such a way that could be used for non-sequential writing and reading. The author proposed to create an information infrastructure where hypertext enables non-linear organization of information, and cross-reference

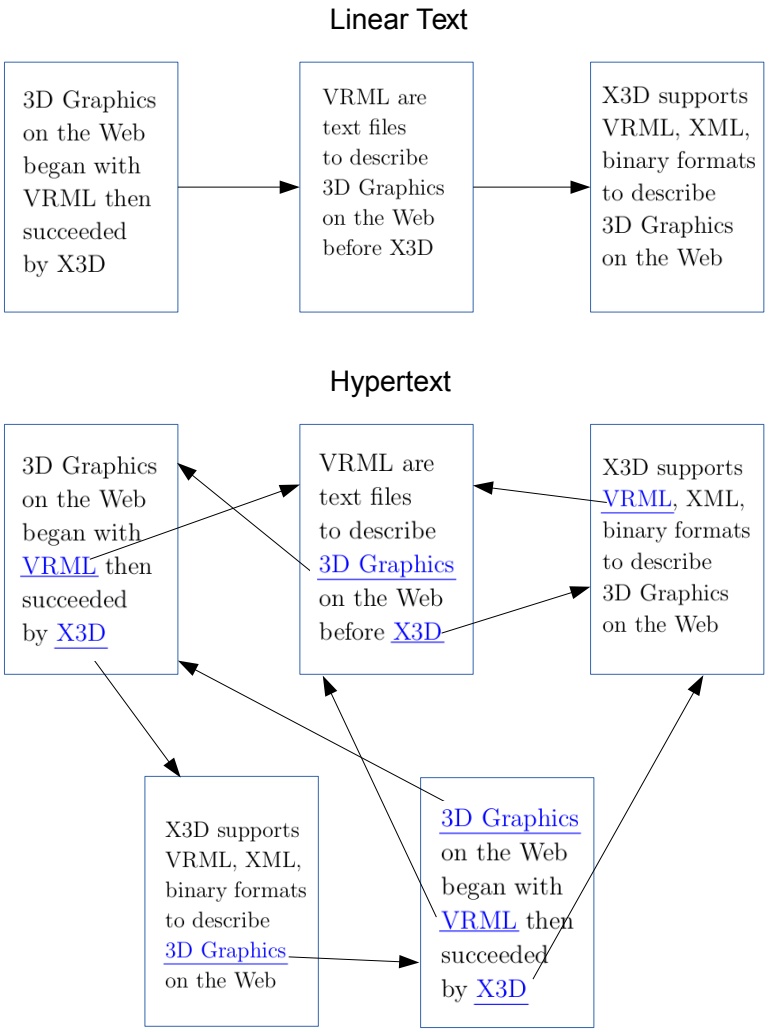

Figure 2.1: Difference between Hypertext (bottom part) and Linear Text (top part) – adapted from the demonstration of hypertext introduced by Jankowski, 2011 [56].

access to such information through the use of links and connections, called *hyperlinks*. Figure 2.1 illustrates an example of hypertext in comparison with linear text. We create this example based on the demonstration of the difference between hypertext and linear text introduced by Jankowski, 2011 [56]. As can be seen from the figure, linear text (top part) provides users with a sequential organization of information that he/she can follow to read. In contrast, hypertext (botton part) gives users a set of non-sequential hyperlinks, presented as highlighted keywords or phrases (the underlined blue texts in the figure 2.1), so that he/she can use to browse information across documents.

The first application of hypertext, namely *oNLine* system was developed by Engelbart and English, 1968 [46]. oNLine aimed to structure information for easy

access using pointing devices such as a mouse. The system stored academic research works into a shared workspace for cross-references between each other. *oNLine* became the first computer system that employed hypertext using point-and-click method with the use of mouse interactions [56]. Since then, there have been many hypertext-based systems developed in the literature based on this application of hypertext using mouse interactions. A detail of these systems can be found in the survey about hypertext and the Web, published by Jeff Conklin, 1987 [42].

A direct extension of hypertext is the concept of *hypermedia* where graphics, audio, videos, or animations are used to enable non-linear organization of information and cross-reference access of associated information. It can be said that hypermedia is the evolution of hypertext thanks to the development of various types of rich media (e.g. 2D graphics and videos) along with texts. Many systems using both hypertext and hypermedia have been proposed in the literature. A detail of those systems can be found in the survey about multimedia and hypertext, published by Jakob Nielsen, 1995 [73]. In this work, we use the word hypertext to mean both hypertext and hypermedia. This is since all the contexts we study in this work about hypertext can also be directly applied to the case of hypermedia.

We have presented the basic concepts of hypertext and hypermedia as non-linear mediums of information. In the next section, we will introduce the origin of the Web that uses hypertext as the mainstream navigation paradigm.

2.1.2 The Web

In this section, we introduce the definition of the Web that integrates hypertext into the Internet for accessing and sharing information. Then we give a review of the web developments that we interest in this work, including the introduction of participatory Web and the popular use of rich multimedia content (e.g. 3D graphics).

2.1.2.1 Definition

The World Wide Web (WWW), or the Web is defined as a set of interlinked hypertext documents accessible via the Internet. It is the heart of Internet services to provide users the access to the numerous collection of information resources (e.g. text, images, videos and 3D graphics). The history of the Web started in March 1989 when Tim Berners-Lee, a British computer scientist working at CERN in Geneva, introduced the idea of using hyperlinks to link and access information of various kinds as a web of nodes in which the users can browse at will on the Internet [31].

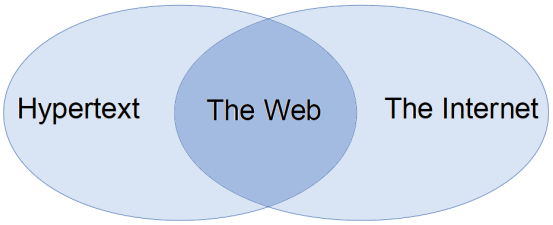

Figure 2.2: Definition of The Web.

The Berners-Lee's breakthrough is the integration of hyperlinks on the *Internet* in which web information resources can be used and shared by everyone all over the world. Figure 2.2 illustrates the emergence of the Web where hypertext is used on the Internet to access and link information globally. During his project, Berners-Lee developed the three fundamental technologies for a working web [31] as follows:

- Hypertext Markup Language - HTML. It is the language used to design web pages that may contain text, images, videos, and other multimedia resources;
- Uniform Resource Locator - URL, and Uniform Resource Identifier - URI. URLs and URIs are used to identify resources on the Web;
- Hypertext Transfer Protocol - HTTP, the main access protocol of the Web. It allows web browsers to request resources from remote HTTP servers. In 1990, Berners-Lee used the *NeXT Computer* as the first web server, and to write the first web browser, *WorldWideWeb*, as well as the first web page.

In order to browse the Web, a user needs a web browser that is run on an Internet-connected device. The most common web browsers are Mozilla Firefox, Google Chrome, Microsoft Internet Explorer and Apple Safari. The first thing the user has to do is to enter in his/her browser the URL of the site. After landing to the site, he/she can view its web pages that may contain text, images, videos and other multimedia, and navigate between them via hyperlinks presented as highlighted keywords or phrases. From usage point of view, the Web contains a set of websites. Each website presents a set of related web pages associated with a single domain. Some examples of popular websites are Facebook for social activities, Wikipedia for knowledge retrieval, and Amazon for online purchases.

The emergence of the World Wide Web brings the closed hypertext systems to the Internet that connects every hypertext user into a common shared global space, called hyperspace. This space allows all web users to access and share information

with each other using supported hyperlinks. In this way, the Web broadcasts information resources all over the world with the use of hypertext as the mainstream navigation means.

We have presented that the Web is the system which integrates the use of hypertext on the Internet as the navigation paradigm to access the worldwide information resources. We now give an overview about the evolution of the Web from the origin (Web 1.0) to the current Web (Web 2.0), which is the participatory Web with the popular use of rich multimedia resources (videos and 3D graphics).

2.1.2.2 Evolution

The Web has experienced its evolution from a read-only text-based system (Web 1.0) to the currently rich and interactive Web (Web 2.0) that supports 2D graphics, audio and videos. In this part, we use a top-down approach to introduce this evolution – we focus on two main developments of the Web. The first one is the introduction of the Social Web in Web 2.0 that allows users to participate in the edition and creation of web content. In this part, we want to notify the reader a key feature of Web 2.0, called *Crowdsourcing* that we use in our work. Crowdsourcing emerges thanks to the popularity of the Social Web to encourage user collaboration in their web activities. The second interest of this work is the popularity of rich multimedia contents, particularly 3D graphics that provide a new web navigation model using 3D interactions.

Web 1.0

Web 1.0 or the read-only Web was the early stage of the Web invented by Tim Berners-Lee in 1989 with the idea of creating a common information space in which people could communicate by sharing information via hypertext. In Web 1.0, source materials in web pages are mostly static text, images, navigation icons, and menus created by the web masters and provided to users for viewing and sharing. Such web contents are called published contents (see the left part of figure 2.3). The web 1.0 sites did not regard the use of rich media contents (e.g. videos and 3D graphics) and the participation of web users: they did not allow users to contribute to the creation of web contents and simply acted as consumers of the web pages [43]. In this way, most web 1.0 sites were known as brochureware, that were produced by taking the printed brochures of the organizations and directly copying to the Web for viewing and sharing. Web 1.0 applications were mostly read-only and the interactivity was often limited with the submit forms.

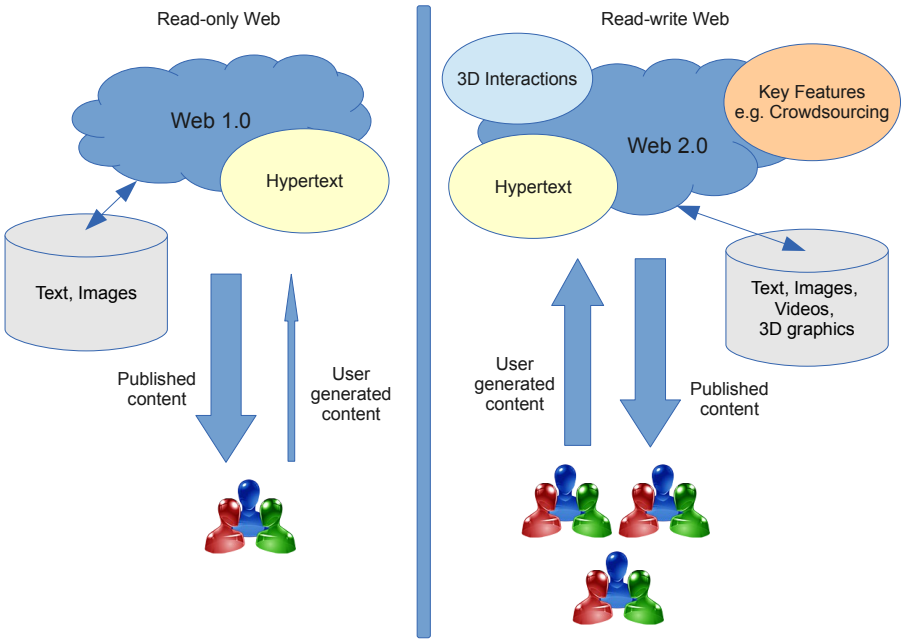

Figure 2.3: Developments of Web 2.0 studied in this work: Online 3D Interactions as New Web Navigation Model, and the use of Crowdsourcing to motivate the contributions of many users in web content creation and edition.

At this period of the Web, there was only a very limited number of websites available and most of them were used for specific purposes such as scientific research and military. The first website was introduced by Berners-Lee in August 1991, and in June 1993, there were about 130 websites [26]. Until 1996, it was estimated to have around 250 thousands Web 1.0 sites available. The term Web 2.0 was firstly coined by Darcy DiNucci in her article "Fragmented Future" in January 1999 [45] as an opening for the next generation of web pages.

Web 2.0

Web 2.0 or the read-write Web is the next generation of Web 1.0 that describes the websites with technologies beyond the static pages of earlier sites. Web 2.0 does not refer to an update to any technical specification, but rather to support more techniques and technologies that change the way web pages are made and used. Two main developments of Web 2.0 in comparison with Web 1.0 are concerned in this work (see the right part of figure 2.3). The first one is the introduction of the Social Web [76], which allows users to participate in the edition and generation of web content. The second one is the use of rich media contents, particularly 3D graphics. The following gives more details about these developments.

The Social Web [76] is a part of Web 2.0 which contains a number of online tools and platforms where people can share perspectives, opinions, thoughts and experiences. The Social Web defines how Web 2.0 makes users as an integral part of the Web – the web participants. It allows users to do more than just retrieve information. A Web 2.0 site may enable users to interact and collaborate with each other as creators and contributors of web content. They generate the so-called user-generated content, in contrast to Web 1.0 sites, where users just act as passive content viewers (see the right part of figure 2.3). Examples of Web 2.0 sites are blogs, wikis, video sharing sites (e.g. YouTube) and social networking sites (e.g. Facebook, MySpace, and Twitter). The Social Web is considered as the most fundamental part which revolutionizes the way web pages are created and used all over the world. It also mainly contributes to the evolution of the Web to become the heart of Internet service as we see today.

Nowadays, the Web becomes the standard used in daily life by everyone with numerous number of websites available. It was estimated that there were about 767 millions websites available in October 2013 [21] – much bigger than 250 thousands Web 1.0 sites in 1996. Moreover, Web 2.0 sites are not only used for specific purposes of scientific research and military, but for all purposes of our daily life, from working, studying to communicating and entertaining. For example, people can now visit the Web to search for some specific information via Google Search, retrieve information and knowledge by reading articles on Wikipedia, entertain music and movies by watching videos on Youtube, or join social activities by signing in to Facebook. In this way, Web 2.0 truly revolutionizes the Web – we see rich multimedia contents (e.g. videos and animations) everywhere on the Web and these contents are used by users for all purposes of daily lives and works.

In our work, we are inspired by these two developments of Web 2.0: the popularity of rich multimedia contents and the introduction of the Social Web. Figure 2.3 illustrates in more details the main developments of Web 2.0 in comparison with Web 1.0 that we choose to study. As can be seen from the figure, in the first case, we found in the literature the recent development of the ultimate achievement of visual media on the Web – online 3D graphics. Although hypertext continues to be the mainstream navigation paradigm, 3D graphics introduce a new navigation model to browse the Web using 3D interactions (see the right part of figure 2.3). 3D interactions offer users additional capabilities to experience the Web such as navigating around the virtual 3D worlds, or inspecting and manipulating the virtual 3D objects in real time. The second development of Web 2.0 concerned in this work is the use of the participatory feature in the Social Web of Web 2.0 where users are allowed to read, change, and add content to web pages individually or collaboratively. Specifically, we study an emerging paradigm called *Crowdsourcing*, which engages the participations of many users to the creation and edition of website content and

presentation (see the right part of figure 2.3). The interesting point of crowdsourcing method is that it not only employs users to generate website content, but also helps to spread out the website to more visitors. Nowadays, crowdsourcing is one of the key features of Web 2.0 [76].

From this context, the goal of our research work is to adopt *crowdsourcing* to enhance the use of online 3D content. There exist two main ideas which inspire our approach. The first one is the use of crowdsourcing to create more web content and the second one is the employment of 3D content and 3D interactions to collect informative user feedback. In the next sections, we will present in more details these ideas in the e-commerce context.

2.2 Crowdsourcing

This section aims at giving a formal definition of crowdsourcing. We begin the section with an overview about the basic concepts of crowdsourcing. Then, we provide a review about the state-of-the-art usages of crowdsourcing in e-commerce websites, that is, the context of our work.

2.2.1 Overview and Definition

The term crowdsourcing was coined by Howe, 2006 [52] as a new solution to exploit human efforts in solving complex problems. Crowdsourcing is short for crowd and outsourcing. It means that we solicit solutions to solve the problems via open calls to large-scale communities. The formal definition of crowdsourcing proposed by Howe was published in Wired magazine, 2006 [52] as follows:

"Crowdsourcing is the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call".

From Howe's definition, crowdsourcing has been used as a technique in which competence and expertise distributed among the members of the crowd are aggregated and exploited in order to produce new content and knowledge, which can be shared and used by different users, as the case of wikipedia. Nowadays, the rapid development of the Participatory Web (Web 2.0) and the increasing numbers of web users all over the world have made sharing information simpler than ever. This has triggered the rapid use of crowdsourcing in the web context. Specifically,

crowdsourcing has become a key feature of Web 2.0 to motivate the participation of end users in the development of the websites. This use of crowdsourcing can be summarized as follows:

- Crowdsourcing is a technique to motivate the contributions of many users in the creation and edition of web content and presentation;
- Crowdsourcing platforms provide tools to observe and meet user needs and preferences.

In order to conceptually model the concept of crowdsourcing, Malone et al., 2009 [70] proposed a conceptual framework, which contains the four building blocks: (1) who is performing the task, (2) why are they doing it, (3) what is the common goal, (4) how is the task performed. The goal of this framework is to provide developers a conceptual model of crowdsourcing paradigm to design the crowdsourcing process for their applications. From this conceptual model, worker incentives have become the principle of crowdsourcing which shows why people are willing to perform crowdsourcing tasks. Currently, there exist two main categories of crowdsourcing incentives, namely (1) entertainment and personal enjoyment, i.e. people participate in the crowdsourcing tasks to get fun or entertain a favorite, and (2) financial reward, i.e. people involve in crowdsourcing in order to get some benefits such as money or discount codes. Figure 2.4 shows two examples of crowdsourcing platforms which work based on these worker incentives. For example, LabelMe allows users to annotate as many image objects as desired to further enrich existing annotated images or offer new ones. In contrast, Amazon Mechanical Turk pays the user for each crowdsourcing task he/she completes.

Figure 2.4: Examples of Crowdsourcing Platforms: LabelMe [13] (left column) and Amazon Mechanical Turk [4] (right column).

We have presented an overview about the basic concepts of crowdsourcing. In the following, we will provide a review about the state-of-the-art usages of crowdsourcing in e-commerce context.

2.2.2 Crowdsourcing in E-commerce

In order to develop a crowdsourcing platform, two main criteria need to be identified. The first one is related to worker incentives (e.g. discount codes in e-commerce context). The second one is the setup of crowdsourcing tasks to assign to users. Little et al., 2010 [68] classify crowdsourcing tasks in the web context into two main categories:

- Decision tasks, which allow users to provide opinions about a web content and presentation (e.g. opinion about an online product such as rating and comparison);
- Creation tasks, which allow users to create, edit, or compose a new web content or presentation.

In the following, we give an analysis about these two categories of crowdsourcing tasks in e-commerce context.

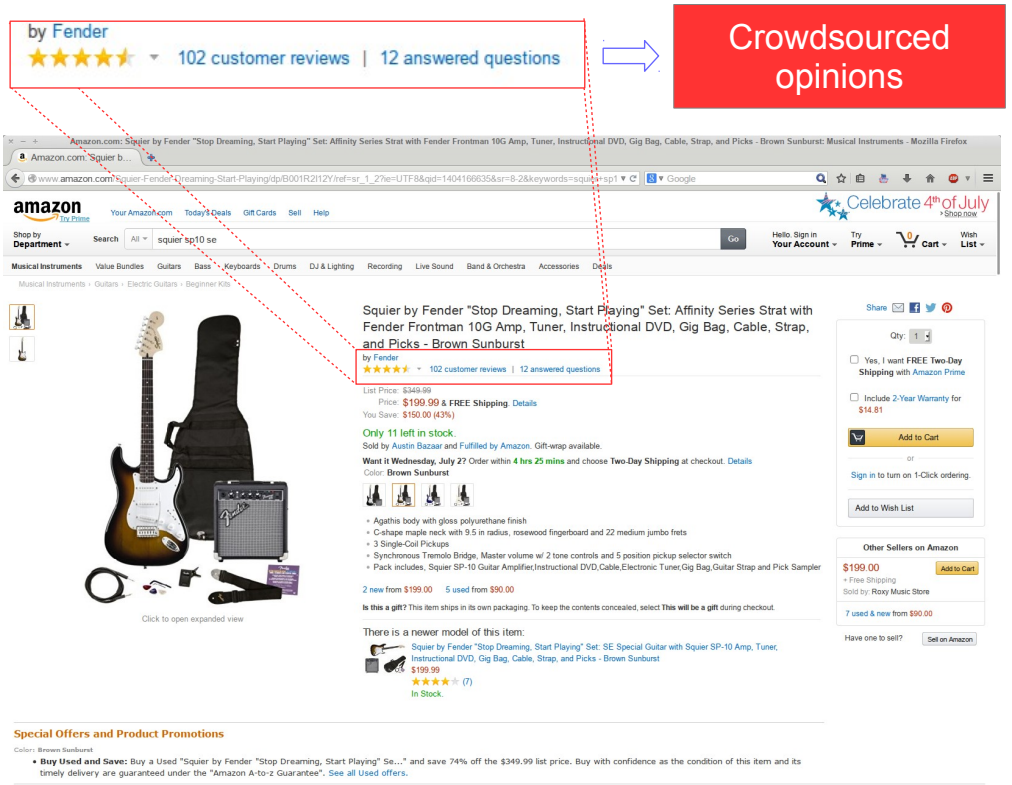

Figure 2.5: Analysis of Crowdsourcing Tasks in Amazon E-commerce Site [3]

Figure 2.5 shows a representative of e-commerce web pages in Amazon site [3]. As can be seen from this figure, we observe that only crowdsourced opinion tasks,

denoted in a zoomable red rectangle, appear in this webpage. These crowdsourced opinion tasks simply ask users to rate the quality of the product (the guitar) from one star (lowest) to five star (highest). Moreover, one can notice that these crowdsourced opinion tasks are placed in a separated context from the product description of the webpage.

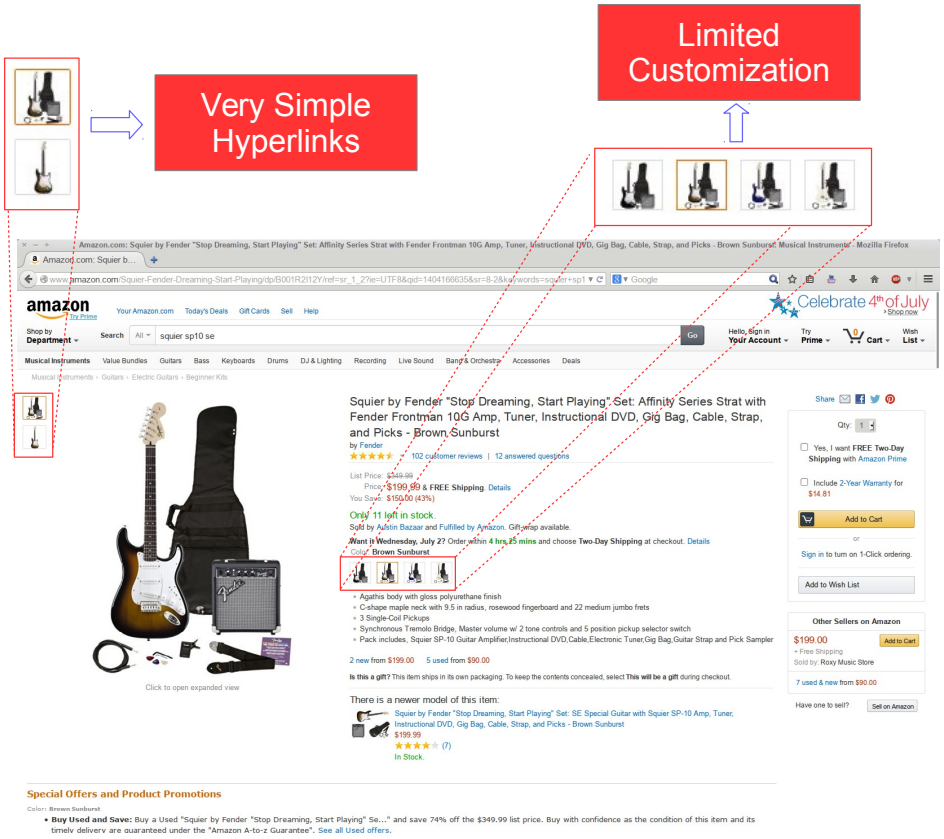

Figure 2.6: Limited Interactions: Swapping Images and Color Changes in Amazon E-commerce Site [3].

Figure 2.6 provides a further analysis of interactions in this e-commerce webpage. As can be seen from this figure, we found that interactions are currently limited to very simple hyperlinks in order to just swap images (see left part of the figure) or change color of the product images (see right part of the figure).

From this observation and analysis, our first research objective is to fill this shortage of crowdsourcing usage by **adopting crowdsourcing to create more web content**, i.e. to generate more crowdsourced creation tasks. In the following section, we will provide our second research objective, that is, employing 3D content and 3D interactions to collect informative user feedback.

2.3 3D Graphics on the Web

This section presents the state-of-the-art 3D techniques and technologies we choose to use and provides our second research objective. We begin the section with an overview about 3D graphics on the Web. Then we show our choice to study 3D web applications using the mainstream 3D viewers (VRML/X3D viewers) and the emerging plugin-free rendering approach. After that, we present 3D interaction techniques supported in these mainstream 3D viewers for the web context. We end the section with a survey of perspective uses of online 3D graphics and provide our second research idea of collecting user feedback from 3D content and 3D interactions.

2.3.1 Overview and Objectives

In recent years, advanced developments of web technologies, network bandwidth, processing powers and abundant memory make it possible to build interactive 3D graphics that can be experienced through the Web. The ability to visualize and manipulate 3D content in real time seems to be the next evolution of the Web for a wide number of application areas such as e-commerce [37], education and training [39, 54], architecture design, tourism [84], virtual museums and virtual communities [67, 66]. Websites distributing 3D content, also called 3D websites, are increasingly employed on the Web nowadays. Chittaro and Ranon, 2007 [38] divided the 3D websites into two categories:

- Sites that display interactive 3D models of objects embedded into web pages [1, 7], such as e-commerce sites allowing customers to examine and inspect 3D models of the product;
- Sites that are mainly based on 3D virtual environments (3D VEs), which are displayed inside the web browser, such as social networking sites (e.g. Second Life [16]) allowing users to take part in virtual social activities.

In the first case, interactive 3D objects are visualized and manipulated on the web pages similarly to other traditional web data such as texts, images and videos. The information structure and user interaction model of web pages are still based on the hypertext model, but are complemented with 3D visualizations and 3D interactions. In the second case, a virtual world or a 3D virtual environment displayable inside a web browser is provided to users so that they can navigate and perform the virtual actions. With 3D VEs, the information structure is a 3D space and user interaction

model is through movements and actions in this space such as walking around, moving forward or pointing left and right.

Behr et al., 2009 [28] presented that right after the first 2D HTML pages went online, people are thrilled with the idea of having 3D content on the Internet. Chittaro and Ranon, 2007 [38] demonstrated that the use of 3D graphics on the Web may have many advantages for users.

- 3D objects/scenes are more intuitive and interactive than the traditional media such as images or video. With 3D data, the user can grasp detail information on the visualized object from any preferred viewpoint, or render with different rendering parameters (like color, wireframe or light).
- 3D interfaces allow users to experience fun and realistic feelings. Eventually, users' emotional and social needs can be satisfied. For example, in a virtual city, the user's avatar can walk around the virtual buildings or even can communicate with other virtual pedestrians.

The main objective of online 3D graphics is to increase the effectiveness, usefulness and usability of the websites by adding value in interacting with 3D visualizations, or in providing users a virtual experience close to the real world one [38]. For example, in e-commerce sites, 3D models allow customers to visually inspect, manipulate, try and customize the products before buying; or in virtual museums, 3D VEs present not only 3D visual reproduction of museum items, but also convey their cultural information to the visitors (e.g. religions and arts) [38].

We have presented that the main goal of online 3D graphics is to provide users additional 3D interaction capabilities to browse the Web. In order to reach this goal, there has been a large number of systems and proposals to render 3D graphics on the Web [28]. In the next section, we will introduce the state-of-the-art 3D rendering methods we use in this work.

2.3.2 Visualizing 3D Graphics

The history of online 3D graphics began with the introduction of VRML (Virtual Reality Modeling Language), that is the language for building and delivering 3D web content [20], at the Second International Conference on the World Wide Web in 1994 in Geneva. The first specification of VRML (VRML 1.0) was published in 1995, and after one year, its improvement was published as VRML 2.0 in 1996. In 1997, VRML

became an ISO standard file format for representing interactive 3D objects and 3D VEs on the Web with the name of VRML97. Despite its popularity, VRML97 faces the problem of complexity and incompatibility to be rendered and accessed in the web browsers since VRML encoding is not understood by most programming languages, and thus each VRML viewer has to handle its own lexer and parser to interpret and process VRML files. VRML97 is finally superceded by a new ISO standard, called eXtensible 3D Graphics (X3D), or VRML 3.0 [23] in 2004.

Besides VRML/X3D, there exists other standards to deliver 3D content such as MPEG-4 Part 11 [77], Adobe Flash Version 10 [2], and Sun Java3D [17]. MPEG-4 Part 11 or Binary Format for Scenes (BIFS) is a MPEG-4 scene description protocol for composing MPEG-4 objects, describing their interactions and animating 2D and 3D MPEG-4 objects. Although there is a 3D subset in MPEG-4 standard, none of the current major MPEG-4 players supports 3D content [28]. Due to this, MPEG-4 part 11 is now still unpopular in 3D web community. Another standard to deliver 3D web content is Adobe Flash Version 10. Basically, in its version 10, Flash attempts to include 3D transformations and 3D objects for 3D composite and Graphical User Interface (GUI) effects. This inclusion of Flash has attracted its 2D users in PaperVision project since Flash has produced very impressive effects on 2D rendering pipeline [28]. However, in 3D rendering pipeline, Flash Version 10 is limited to only simple 3D transformations and has not yet exposed any perspectives to render complex virtual 3D objects and virtual 3D VEs on the web pages. Besides MPEG-4 Part 11 and Adobe Flash Version 10, there exist many other technologies to render online 3D graphics such as Sun Java3D [17] or Microsoft Silverlight [14]. In our work, we chose to use VRML/X3D since this technical choice does not restrict the contributions of our proposed approach. In the following, we will first give in more details basic concepts and definitions of these languages.

2.3.2.1 Modeling Languages

VRML

VRML documents are text files that describe interactive 3D objects and 3D VEs. Each VRML file is a collection of objects called *Nodes*. Nodes can be something physically like sphere, cylinder and cones, or non-physically like viewpoints and transformations. VRML defines 54 different types of nodes, including geometry primitives, appearance properties, sound and video, and nodes for animation and interactivity. Some nodes are container nodes or grouping nodes which contain other nodes (e.g. Group or Transform nodes). Each node in VRML contains *Fields* which present node properties and hold the data of the node. VRML defines 20 different

types of fields that can be used to store different types of data from single integers to arrays of 3D rotations [38]. VRML arranges nodes in hierarchical structures called *Scene Graphs*, each scene graph is a directed acyclic graph to connect VRML nodes. Besides nodes and fields to render 3D worlds, VRML also defines mechanisms (e.g. DEF/USE) and prototypes (e.g. VRML PROTOs) that allow developers to develop the VRML language. For example, the mechanism DEF/USE allows to re-use declared geometries and VRML Protos enables developers to create new VRML nodes.

X3D

X3D is a direct successor of VRML with major improvements. For example, X3D supports multiple data encoding formats (VRML, XML and Binary encodings), in contrast to only-VRML encoding in VRML. These encodings allow X3D content to be represented and stored with better data compression; to be rendered and transferred over the Internet with more brower compatibility and lighter transmission. Another example of X3D improvement is that X3D divides the language into *Components and Profiles*, each component corresponds to a functional area, and each profile corresponds to a classification of applications and devices. This division makes X3D easier and lighter to be implemented by developers than VRML since in VRML each single viewer has to handle the whole VRML specification. Along with this, X3D data is also compatible with more advanced 3D graphics techniques such as programmable shaders and multiple texturing rather than VRML data. Thanks to these major improvements compared to VRML specification, X3D is now the mainstream web modeling language of online 3D graphics and is recommended. X3D also enables the direct conversion from VRML data to X3D data using VRML/X3D converter tools.

We have showed our choice to use VRML/X3D languages. However, X3D is just a specification that allows to describe 3D graphics in X3D formats. In order to render and access X3D web content, it requires the use of the so-called X3D viewers, or X3D rendering engines. In the following, we will present the two main approaches to render and access X3D web data.

2.3.2.2 Rendering Methods

To render and access X3D web content, there are two main approaches: the use of 3D web browser plugins (e.g. Octaga [15] and Flux [10] players), and the use of web browser built-in rendering engines that are plugin-free (e.g. X3DOM [24]). The plugin-based approach, however, disturbs users with the problems of security and

browser incompatibility. Moreover, users have to download and install the plugins by themselves. In order to overcome these problems, Kronos Group [22] introduces a JavaScript API, called WebGL (Web Graphics Library), for rendering interactive 3D graphics natively inside web browser without the use of plug-ins. Nowadays, WebGL is officially enabled in most popular web browsers such as Mozilla Firefox, Google Chrome and Apple Safari.

Recently, we have seen the next evolution of the Web's Hyper Text Markup Language (HTML), called HTML5 [11]. HTML5 is designed to deliver all the online web content (animations, movies, 3D graphics, etc) without requiring additional plugins. For example, we may watch a live video on a HTML5 web page without installing any flash plugins. These developments of HTML5 and WebGL API [22] create a foundation for plugin-free 3D web applications where X3D web data [23] can be rendered and used natively in the Web browsers similarly to other multimedia data (e.g. images, animations and videos).

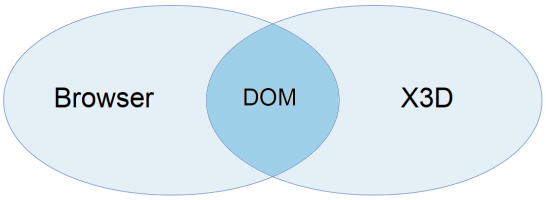

Figure 2.7: Integrated X3DOM Model [28].

In this context, we choose to use the plugin-free approach since it overcomes the problems of security and browser incompatibility in plugin-based approach. We found in the literature an emerging X3D rendering framework developed by Fraunhofer IGD group, called X3DOM [24] to implement this plugin-free rendering approach. The main idea of X3DOM is to directly integrate X3D nodes into HTML5 on top of WebGL (see figure 2.7). Specifically, X3DOM provides a runtime system, based on WebGL APIs, so that X3D scenes encoded as declarative HTML5 tags can be rendered and manipulated as HTML5 DOM elements. It can be said that X3DOM provides a practical scenario in which X3D web data [23] can be natively rendered in web browsers and be accessed by users without any plugin installation. Moreover, X3DOM renders X3D scenes as DOM elements which makes X3D data as a part of first class web data similarly to other web content such as texts, images and videos.

We have presented our choice to use the mainstream 3D web viewing applications (VRML/X3D) and the plugin-free solution to render and manipulate X3D data (X3DOM rendering framework). We now introduce the main 3D interaction techniques supported in these 3D systems in the next section.

2.3.3 Interaction Techniques

In this section, we study a new web navigation model – 3D interactions which emerges thanks to the development of 3D graphics on the Web.

People spend their lives in a 3D world and develop skills for manipulating, selecting and navigating around 3D objects [62]. 3D interactions in computer graphics aim to provide users the interaction with 3D visualizations, that is close to natural skills the users learn in their real life. It can be said that 3D interactions mainly contribute to the difference between 3D graphics and other traditional media such as images and videos. During the development of 3D computer graphics, there have been many types of 3D interaction techniques proposed in the literature. For example, in CAD (Computer-Aided Design), 3D graphics can be manipulated using 3D joysticks or shutter glasses; or in an immersive virtual environment, a virtual reality world can be experienced using head-mounted display. These 3D interaction techniques, however, are specific for some domains (e.g. CAD or visualization), or for immersive virtual environments. In the web context, 3D interaction techniques are mainly based on common, general-purpose hardware for interactions such as mouse and keyboard. This is since mouse and keyboard are the most popular input devices used to browse the Web nowadays. The next parapraphs provide in details mouse-based 3D interaction techniques used on the Web.

Mouse-based 3D interactions can be characterized in terms of three fundamental interaction operations, namely *Rotate*, *Pan* and *Zoom*. These operations allow the primary movements of the camera to move and rotate 3D objects in a 3D space according to the actions created by end users. Rotate/Pan/Zoom are now the standard 3D interactions used in, not only 3D web applications, but also in almost every 3D modeling environment (e.g. Autodesk's 3ds Max, Maya, or Blender) [62]. This is since Rotate/Pan/Zoom commands are easy to use for users and well compatible with a pointing device such as a mouse. The following provides in details the definitions of these operations.

Rotate

3D rotation allows to rotate the camera around an axis in the 3D space, called the axis of rotation. Figure 2.8 gives an example of a 3D coordinate system xyz. In 2D, the axis of rotation is always perpendicular to the xy-plane, i.e. the z-axis. However, in 3D, the axis of rotation can have any spatial orientation, i.e. can be an arbitrary axis. Figure 2.8 illustrates example of 3D rotations around the three principal axes, namely x-axis, y-axis and z-axis from left to right respectively. In more details, x-axis rotation rotates the camera an α-angle counterclockwise the x-axis

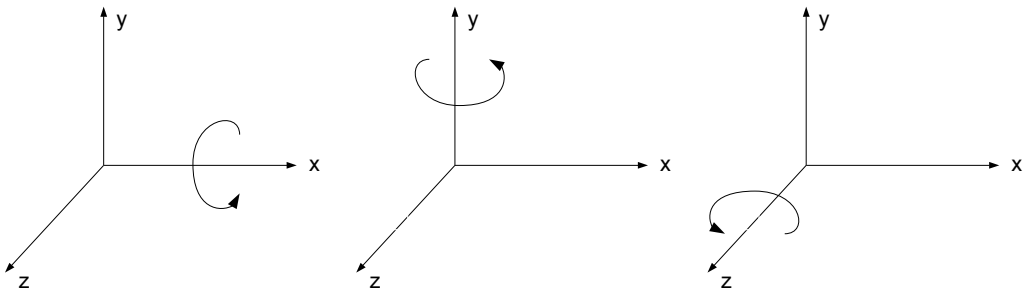

Figure 2.8: Example of 3D Rotations around the three principal axes – x-axis (left column), y-axis (middle column), and z-axis (right column).

of the 3D coordinate system xyz (left column); y-axis rotation rotates the camera a β-angle counterclockwise the y-axis of the 3D coordinate system xyz (middle column); and z-axis rotation rotates the camera a γ-angle counterclockwise the z-axis of the 3D coordinate system xyz (right column) respectively. A 3D rotation around an arbitrary axis can be identified by combining the rotations about these three principal x, y, z-axes.

Rotate is sometimes referred to as *Tumble or Sweep*, which performs 3D rotation by orbiting the camera around a reference point in any direction. Specifically, it sweeps the camera around horizontally and vertically on a virtual spherical track, keeping it focused at the same reference point for generating 3D rotation [62].

Pan

The Pan command moves the camera from one side to another side of the 3D scene. In the context of online 3D interactions, Pan refers to the task of moving the camera from one side to another side along the horizontal axis or along the vertical axis on the scene. For example, with the 3D coordinate system presented in the figure 2.8, the tasks of moving the camera along x-axis between left and right, and along y-axis between up and down are considered as Pan commands.

Zoom

The Zoom command involves changing the focal length of the virtual camera to make the object appear closer (also called Zoom-in) or further away (also called Zoom-out) in the scene. Zoom is one of the most familiar and frequently-used user interactions. In the context of online 3D interactions, Zoom refers to the task of translating the camera along the front-to-back axis to make it appear closer or further away in the scene. For example, with the 3D coordinate system presented in the figure 2.8, the tasks of translating the camera forward and backward the z-

axis to make the object closer to further away in the scene are considered as Zoom commands.

Zoom is sometimes referred to as *Dolly*, which performs the Zoom operation by moving the camera forward and backward the subject. In more details, Dolly-in means to step towards the subject with the camera – the subject appears closer, and Dolly-out means to step backwards the subject with the camera – the subject appears further.

Besides these standard interactions, most VRML/X3D viewers implement additional interaction techniques. For example, *Look Around* technique, which changes the orientation of the camera but keeps it at a fixed position, is also a very popular interaction technique supported in VRML/X3D viewers. In order to provide these 3D interactions to users via 2D input devices such as mouse and keyboard, VRML/X3D viewers identify the navigation modes, each maps a 3D interaction command (Rotate/Pan/Zoom) to a type of mouse buttons, or to a specific keyboard button. For example, in X3DOM viewer, the user needs to drag the mouse while holding the left mouse button pressed to rotate; drag the mouse while holding the middle mouse button pressed to pan; and drag the mouse while holding the right mouse button pressed to zoom.

From user point of view, 3D interactions are used to perform online 3D tasks. A summary of these 3D tasks can be found in the survey about 3D interaction techniques done by Jankowski and Hachet, 2012 [62] or in the taskonomy of 3D web use proposed by Jankowski, 2011 [57]. In this part, we present the most common 3D tasks on the Web, namely *Navigation*, *Selection* and *Manipulation*. The next paragraph provides more details on these tasks.

Navigation

Interactive 3D environments usually represent more space than can be viewed from a single viewpoint [62]. 3D navigation refers to the task of moving the viewpoint through 3D environment. It allows users to get around in the 3D environment in order to obtain different views of the scene. Web users can use 3D navigation to explore virtual 3D environments or inspect virtual 3D objects. 3D navigation on the Web is commonly performed using the standard 3D interactions (Rotate, Pan and Zoom).

Selection and Manipulation

Other common online 3D tasks are object selection and its direct manipulation. In more details, 3D selection refers to the task of choosing a 3D object, and direct

3D manipulation refers to the task of specifying object position, orientation, and scale. Interaction techniques for 3D manipulation include object translation, object rotation, and object scaling. One common method for selecting 3D objects in VEs is to position a mouse cursor over the given object and click on a mouse button. The technique is based on raying casting that uses the ray from the eye point through the pixel currently selected by the mouse pointer to find the first intersection point between the virtual ray and the surface of the target or its approximated surface such as bounding box [71].

Although this raying casting technique is commonly used for selection task in desktop 3D VEs, it is heavily CPU-consuming for JavaScript-based VRML/X3D viewers on the web [29]. This is because the intersection point between the virtual ray and the surface of the target is retrieved by traversing the scene-graph, coarsely checking bounding boxes, and testing all triangles [29]. The whole process is high computational for JavaScript-based rendering framework such as X3DOM [24]. To overcome this limitation, Behr et al., 2010 [29], the founders of X3DOM framework, proposed a render-buffer-based approach to find the intersection point. Specifically, the picking buffer is implemented by first rendering the scene into a framebuffer object so that the world coordinates of the 3D points on the surface of the 3D shape are encoded into the RGB channels, and the alpha channel contains the object ID to reference the rendered shape. Then, the values located under the mouse pointer on the canvas indicated by the user are retrieved and mapped to the rendered framebuffer to obtain the picked 3D object. This technique has been shown by Behr et al., 2010 [29] to improve the performance of 3D selection task on the Web rather than the raying casting technique. In this context, we also use this buffer-picking technique proposed by Behr et al., 2010 [29] for our research work.

We have presented the state-of-the-art 3D web technologies (VRML/X3D, X3DOM) and mouse-based interaction techniques we study in this work. In the next section, we will give a survey about perspective uses of online 3D graphics and provide our second research objective, that is the employment of 3D content and 3D interactions to collect informative user feedback.

2.3.4 Using 3D Graphics

This part provides a survey of perspective uses of online 3D graphics. The goal is to outline the advantages that 3D graphics offer to different web application domains. The section starts with an introduction about the interface guidelines that a 3D website should have. Then it details perspective application domains of 3D websites which motivates this research work.

2.3.4.1 Interface Guidelines

Based on previous works on web usability [74, 75, 64] as well as observations on the use of 3D graphics on popular 3D games, Jankowski and Hachet, 2012 [62] provided the guidelines that the interface of a 3D website should have as follows:

- 3D websites should explain themselves: 3D web pages should be self-evident, obvious, and self-explanatory. Users should be able to learn what the web pages are and how to use them without spending any effort thinking about it. This targets most of the web use, that is, not to waste user time;
- Text: since text is easy to read and to convey information, a 3D web page should support readable and explanatory texts to users;
- Navigation: a web page should help to minimize the number of navigation steps for users to accomplish their tasks, and simplify their interactions in 3D environments (e.g. keep movements planar, or use collision detection);
- Help: should be embedded in the interfaces of 3D websites so that users can ask for some explanation;
- Keep user habits: when browsing the web, users tend to follow something that they are familiar with. The interface of 3D web pages should use existing web conventions rather than mainly proposing new ones.

The interface guidelines provided by Jankowski and Hachet, 2012 [62] shows that a 3D website should be designed in a close relation with traditional web of hypertext and hypermedia. For example, it should be easy to use (e.g. via text reading, and simple interactions), be self-evident and obvious so that not to waste user time and preserve user habits that they are already familiar in the conventional web. This interface guidelines of Jankowski and Hachet outlines the need to develop tools and systems which ease the use of 3D graphics on the Web in a close relation with hypertext and hypermedia.

We have presented the interface guidelines that 3D web pages should meet. We now present the perspective application domains of 3D websites and provide our second research objective.

2.3.4.2 Application Domains

In this section, we will provide an analysis of advantages that 3D graphics propose for various application domains on the Web. Chittaro and Ranon, 2007 [38] presented a survey about the main application domains of 3D websites. In this part, we first recall Chittaro and Ranon's survey and adjust it with the up-to-date state of online 3D graphics on the Web. Our aim is to provide the readers a clearer view of perspective uses of online 3D content and highlight the capabilities to record a lot of user feedback from this use of 3D content and 3D interactions that our work targets.

Education and Training

3D objects and 3D VEs offer the ability to reproduce the real world or to create imaginary worlds that can be used to help users in their learning process. For example, when users interact with virtual 3D objects, or experience realistic 3D VEs, they are provided with realistic representation of subjects and phenomena, that they can use to analyze the same subject from different points of view. This overcomes the limitation of traditional education methods of paperwork and 2D graphics where users have to imagine in their mind how the concepts look like and how they may interact with them. Chittaro and Ranon also emphasized the online phase in the use of 3D graphics for education. The authors indicated that delivering educational 3D VEs through the Web allows one to reach potentially large numbers of learners worldwide, at any time [39]. Examples of using online 3D graphics in education include 3D reconstructions of parts of the human body in medical education; 3D simulators of the parts of the aircraft in mechanical education; or vocabulary lessons in foreign language education [38].

Virtual Communities

3D virtual communities on the Web allow a large number of users to build and inhabit together within a 3D virtual space. Users in a virtual community can interact with each other in order to pursue mutual interests and goals. They may also collaboratively engage in many other acitivities such as the construction of large scale spaces (e.g. buildings, towns) or joining the social activities (e.g. participating in a virtual ceremony). Recently, the advent of the Internet and the development of the Social Web (Web 2.0) increasingly improve the number of 3D virtual communities and their users. For example, Second Life [16], the popular social networking site on the Web nowadays, has 36 million users over the last 10 years, and an average of about 400,000 new registrations are still created monthly. In Second Life, users,

called residents, are allowed to do various activities, such as explore the world, meet other residents, socialize and participate in individual and group activivies.

Architecture and Virtual Cities

Virtual cities are simply the virtual reproduction of the real world places, where users may be allowed to move inside 3D models of a building, or sometimes may be able to communicate with other users (e.g. using chats). The use of virtual cities may be useful for tourists such as helping them to gain an overview about the real city they may prefer to visit, or supporting them with detailed guides. Another application of virtual cities is to help architects in improving the plan, design and management of the real cities [38]. A popular type of virtual cities is the Ancient City, which presents the heritage of different cities all over the world. For example, visiting a Thai Ancient City [18], visitors can experience the atmosphere of Thai livings including the history, cultures, religions, arts and customs from dawn until the present.

Virtual Museums

Virtual museums allow to provide online collections of cultural information to visitors. These collections may be useful if the digital representation of physical items contain enough detail to meet users' needs and interests. With collections such as photographs or manuscripts, their cultural information can be conveyed effectively to users via 2D images [38]. However, with some collections such as sculptures and statues, 2D images may not contain enough cultural information to support users's visit since a lot of spatial information and relationship on 3D shapes of the sculptures and statues are lost via their representation as 2D images. In these cases, using 3D models can better enrich the museum experience for web visitors.

E-commerce

Recently, we have seen the emergence of 3D e-commerce sites that attempt to bring 3D interfaces to users. Some of those websites belong to the first category of 3D websites, i.e. they use hypermedia model for information presentation and interaction, but allow the integration of interactive 3D models of the products on the product page so that users can visually inspect, examine and customize the products in real time before purchasing. For example, at Fiat [7], users are allowed to rotate an online car before buying or in FittingBox, users can try on virtual glasses in real time before eventual buying [8]. The second type of 3D e-commerce sites are virtual stores where users are allowed to explore the virtual representation of a store similarly to their shopping actions in real life. For example, at Ardzan [5], users can choose an avatar and use it to experience a realistic 3D virtual fashion store or a

3D supermarket store. The use of these 3D interfaces, as discussed by Chittaro and Ranon, 2007 [38] may provide the following advantages to users:

- They are closer to the real-world shopping experience, and thus more familiar to the customers;
- They support customers' natural shopping actions such as looking at different parts of the 3D product or walking around a virtual store;
- They can satisfy emotional and social needs of customers, since users can freely inspect parts of the products or explore an immersive virtual environment by themselves or with other customers and sellers.

On the today web, 3D e-commerce is a very interesting use of online 3D content. Indeed, 3D products provide users with new 3D interaction capabilities (e.g. virtual touch screen interactions) as well as allow them to manipulate a lot of rich 3D product configuration (e.g. changing color, size, or material of the product). This use of online 3D products not only makes users enjoy new functionalities and new degrees of freedom, but also helps the shop owner to know more about customers by collecting informative feedback from their 3D interactions.

From this survey, we observe that the use of online 3D content and 3D interactions provide a lot of implicit user feedback. This potential benefit of online 3D content inspires our second research objective, that is, **employing 3D content and 3D interactions to collect implicit crowdsourcing feedback**.

2.4 Chapter Summary

This chapter presents the foundation and motivation of this thesis. We introduce the context we choose to study – 3D interactions as new web navigation model and crowdsourcing as a key feature of Web 2.0 to engage web user participation. From this context, we present our two research objectives. The first one is to use crowdsourcing to create more web content, and the second one is to employ 3D content and 3D interactions to collect informative crowdsourcing feedback. Given these foundation and motivation, our work proposes to use crowdsourcing to: (1) simplify 3D interactions using implicit crowdsourced user feedback, and (2) create semantic links between text and 3D visualization to enhance 3D browsing around online 3D products. In the next chapter, we will present in more details our first proposal.

Chapter 3

Easing 3D Interactions with Virtual 3D Models

Contents

This chapter studies the first problem of using online 3D graphics, that is, 3D interactions are cumbersome with numerous degrees of freedom. We first identify in details the problem, then study the state of the art methods and related works in the literature to solve the problem. Finally, we present our proposed method using *crowdsourcing* to ease interactions with online virtual 3D models.

3.1 Motivation

We have presented in the chapter 2 that the main use of 3D graphics on the Web is to provide users with additional capabilities to interact with 3D content in real time. This support of online 3D interactions to web users, however, poses specific issues.

First, 3D interactions are cumbersome to apprehend as they provide numerous degrees of freedom. Specifically, 3D interactions allow users to interact with 3D visualizations by controlling virtual viewpoints. Each viewpoint control often involves six degrees of freedom (6DOF): three dimensions for translation, and three for rotation. Some 3D applications also provide more degrees of freedom for scaling and shearing. The problem here is that 3D interactions require users to control too many parameters of viewpoint movement that they may easily get frustrated in learning how to interact in 3D. For example, new-to-3D users, who may be domain experts, but have marginal knowledge about interaction techniques, can find it difficult to manipulate a 3D visualization. It is common for them to face the trouble of "where my object is in the scene", or to take them too long to get familiar with the supported interactions. These problems often cause the results of users leaving the site since they often do not want to waste their time and effort on learning how to interact with the websites, but prefer consuming the website content directly (see the interface guidelines presented by Jankowski and Hachet, 2012 [62]). These problems lead to a demand to simplify 3D interactions so that users can manipulate 3D content in less interaction efforts. This need is particularly important for the web context, where most users are not experts in 3D visualizations and interactions.

Second, besides cumbersome 3D interactions, complex 3D objects often require powerful computations. Delays in rendering or time lags in transmitting due to the limited bandwidth and big latency can limit users from interacting with virtual 3D models easily. For example, assuming that the user is manipulating an online 3D model with average size of 15MB and having a good Internet connection of 300KB/s, he/she may have to wait around 1 minute for the content to be rendered and transmitted via the Internet. The amount of time is too long for web browsing since as pointed out in the book "Speed up your site", published by King, 2003 [63] that users will wait for a maximum time of *8.6* seconds to see the loaded content of the web page, otherwise they will leave the site and switch to another web content. This problem shows that it is necessary to support users in accessing online 3D content in less time.

Third, along with the mentioned problems, 3D interactions such as Rotate-Pan-Zoom are provided to users via 2D input devices (e.g. mouse or keyboard) by using the so-called navigation modes – each assigns a mouse button or a keyboard button to one of the 3D operations (e.g. Rotate, Pan, or Zoom). In order to perform 3D interactions using their mouse, users need to accomplish a movement by shifting back and forth among these navigation modes. For example, with X3DOM viewer, users need to switch back and forward among dragging the left mouse button for rotate event, dragging the middile mouse button for pan event and dragging the right mouse button for room event. This accomplishment requires users too much interaction effort to browse the 3D web content since they may have to shift back and forth many times between different navigation modes to inspect different parts of the 3D objects. This is usually not preferred by most web users since they often want to perform simple interactions with the website and focus on browsing the website content (see the interface guidelines proposed by Jankowski and Hachet, 2012 [62]). This problem indicates that it can be helpful to simplify access to the 3D content using simple mouse-button clicks rather than using only the mapping navigation modes.

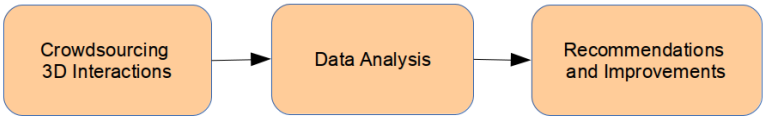

Figure 3.1: General Approach.

From these problems, we are aware of the importance to enhance 3D interactions with online 3D graphics. We see that it is necessary to develop tools and systems that are capable of supporting online 3D interactions with virtual 3D models so that users can access meaningful 3D content in less time and interaction effort, and with simple mouse events. In order to meet this goal, we propose a new paradigm to ease online interactions with virtual 3D models based on crowdsourcing (see figure 3.1). In particular, we collect 3D interactions created by previous users during their crowdsourcing tasks (first column), then analyze these collected statistical data to determine informative regions (called ROIs – Regions of Interest) within a 3D object (second column). A set of 3D poses, called *Recommended Views* is computed and used for recommendations to the next generations of users (last column). The recommended poses aim at simplifying the 3D navigation, that is, reducing the number of pan, zoom and rotate events the user may need to reach his/her desired regions. The main idea of our method is that by analyzing the interaction traces of past users, we can interactively help subsequent users to improve their 3D web browsing, that is, to reduce 3D browsing time and simplify 3D interactions. We propose an implementation of our proposed paradigm using modern 3D web techniques and technologies such as X3D, HTML5, X3DOM and WebGL (see chapter 2). By using

an experimental user study, we will show that our system helps users in saving time and interactions for retrieving the meaningful content of the 3D object, and also our proposed interface allows users to perform online 3D interactions through simple mouse clicks.

3.2 State of the Art

Our proposed approach is related to *Adaptive 3D Interaction Support Techniques*, which can be classified into three categories: (1) Point of Interest (POI) Techniques, (2) Adaptive Hypermedia Extension, and (3) Monitoring User Interactions. In the following, we will present in more details each of these categories.

3.2.1 Point of Interest (POI) Techniques

Mackinlay et al., 1990 [69] introduced a technique called Point of Interest (POI) Movement that provides user a rapid controlled movement of the virtual viewpoint towards a point of interest on the 3D object. In more details, the technique first asks user to indicate a 3D point that he/she is interested in. Based on this indication, it computes a motion function that approaches the POI asymptotically along the ray from the viewpoint to the POI. This function is then used to move the viewpoint logarithmically toward the POI. The result is rapid motion over distances that slows as the viewpoint approaches the POI. During its movement, the viewpoint is also oriented to face the surface containing the POI, by using the surface normal at this POI. Point of Interest Movement technique is also referred to as "Go To" technique that allows the camera to jump to the interesting position specified by the user.

Igarashi et al., 1998 [55] proposed a Path Drawing technique for 3D space navigation, which is an extension of the POI technique. Specifically, the technique allows user to draw a desired walkthrough path directly on the screen using a free stroke. After that, the system calculates a moving path in the 3D world by projecting the stroke onto the walking surface, then moves the camera along this path to reach the target (the endpoint of the stroke). If during the movement, the user wants to reach another target, he/she can draw a new stroke to modify the camera path. Path Drawing technique was evaluated to be slower than POI Movement technique, but preferred more by users, particularly for tactile devices [62].

The main advantage of POI Movement technique is that it is easy to use in 3D applications with the means of 2D pointing device such as the mouse since users

can directly pick the interesting point using the mouse. However, the limitation of POI technique is that it requires users to be familiar with 3D navigation in order to specify which 3D part he/she is interested in. Due to this limitation, POI technique does not work well for new-in-3D users, who often face the problem of "where my object is in the scene" as they just manipulate the 3D object out of its field of view. In contrast, our work offers users simplified 3D interactions using simple mouse click buttons. Users are not required to be familiar with 3D navigation to be capable of using our proposed recommendations deduced from crowdsourced interactions.

3.2.2 Adaptive Hypermedia Extension

In this technique, adaptive 3D interactions are extended from adaptive hypermedia techniques. Specifically, the method restricts user's freedom while interacting with a virtual 3D space in order to (1) present relevant, interesting and compelling 3D locations or 3D objects; (2) provide users the best views of the scenes or of the 3D objects; (3) create paths that are easy to learn and avoid the disorientation of the user; (4) avoid the problem of users getting lost in the virtual space [62].

Hanson and Wernert, 1997 [50] proposed a paradigm to create Constrained Navigation for facilitating 3D navigation using 2D input devices such as mouse or keyboard. The authors divided the navigation space into the two main components: The Constraint Surface defines the range of positional values that the camera can have, and the Camera Model Field (CMF) describes the ideal viewing parameters for each point on the constraint surface. Each time the user controls the motion through the constraint surface, the system determines the corresponding values in the CMF and presents them to the user. In 1999, Wernert and Hanson [86] introduced a taxomomy of the interaction support techniques based on the use of CMF. In particular, the authors developed a design which incorporates not only the task-based geometric constraints on the user's location, gaze, and viewing parameters, but also a personal "guide" that serves two important functions: keeping the user oriented in the navigation space, and pointing to the interesting parts of the 3D objects/scenes as they are approached. If the user is active, these guide's cues may be ignored by continuing in motion, but if the user stops, the gaze shifts automatically toward whatever the guide was interested in.

Hughes et al., 2002 [53] demonstrated that by changing the way CMF viewing information is used during user interaction, an extension from adaptive hypermedia to adaptive 3D navigation supports can be implemented. In this spirit, the authors proposed a mechanism for computing the so-called *ideal viewpoints* in the 3D environments, based on the analysis of the user model. Each ideal viewpoint contains

the ideal viewing parameters inherited from CMF viewing information (e.g. camera position, orientation or focal length) at the location of the 3D environment that may be interesting for the users. During their interaction with 3D environments, the user navigation is constrained or directed with additional information so that the user can reach the suggested ideal viewpoints. Hughes et al., 2002 [53] and Chittaro and Ranon, 2007 [38] classified the adaptive 3D navigation support techniques inherited from adaptive hypermedia as the follows:

- Direct Guidance: This approach provides users a strict line order through the navigation space. The idea is that instead of giving the user options to navigate, the system selects the so-called "best choice", and gives it to the user. In hypermedia systems, this strict line is presented to users as a set of hyperlinks via the interface, usually labeled as "next" button. For 3D environments, direct guidance is defined as a set of pre-determined sequence of viewpoints that the system constructs and provides to the user. Hughes et al., 2002 [53] implemented a direct guidance as a path through the 3D environment that encompassess all ideal viewpoints, and then automatically moves the user's viewpoint along this path;

- Hiding: This technique restricts the number of navigation options to a limited subset. Irrelevant paths are hidden, leaving the users to choose only from the parts of the webpage that are consistent with their task. In hypertext systems, hiding technique is presented to the user by simply disabling links to inappropriate pages. In 3D environments, hiding technique is done by restricting the possible positions or orientations of the camera. For example, if the user needs to discover a configuration of the 3D objects, he/she may be restricted to walking through the scene. Specifically, the user still has the direct control over the X and Y position of the camera, but the Z value is fixed by keeping the viewpoint position at a fixed distance from a surface in the 3D environment. The total 3D navigation options in this technique is often the number of camera positions multiplied by the available camera orientations;

- Sorting: This method alters the order in which navigation decisions are presented to the user. In hypertext systems, sorting technique is implemented by storing the links into a list arranged by relevance. Sorting method cannot be directly transferred to the context of 3D environments since preserving the designed shape of a 3D object or the designed structure of a virtual world is essential for the fidelity of 3D models. Hughes et al., 2002 [53] proposed to implement 3D sorting by ordering the ideal viewpoints and directing the user to move forward the ordered ideal viewpoints. However, the user still has the possibility to override the system decisions to explore other parts of the 3D environment;

- Annotation: This approach displays additional information on navigation options. Annotation technique can be used to provide the user the better indications of the content, or the statements of its relevance. In hypertext systems, annotation methods include changing the color of links, or placing additional icons near the links. In 3D environments, annotation can be used easily. For example, Hughes et al., 2002 [53] introduced the use of the ideal viewing vector to control the direction of a splotlight for traveling through the scene. When the user moves through the 3D environment, the flashlight will shine on the relevant objects, and hence helps users in reaching the interesting parts of the 3D environment.

The main advantage of adaptive 3D interactions extended from hypermedia is that it overcomes the problem of users lost in virtual 3D space. This is since the system guides users to follow a set of pre-defined ideal virtual viewpoints. However, the limitation of this approach is that it depends on expertise of each individual user in 3D interactions and visualizations in order to compute ideal viewpoints. Due to this, produced adaptive 3D interactions may not work well for non-expert users who may be domain experts but have marginal knowledge about 3D interactions. In contrast, our work employs the expertise of the crowd in 3D interactions in order to compute recommendations for helping many common users in accessing 3D content.

3.2.3 Monitoring User Interactions

This technique computes adaptive 3D interactions based on monitoring user behaviors. The main idea of this technique is that user interaction patterns are learnt from their web visits. The deduced patterns are then used to anticipate users' needs and preferences in upcoming 3D navigation.

Example of previous work using this technique was done by Celentano and Pittarello, 2004 [35]. The authors proposed to capture 3D user interactions by using explicit sensors (e.g. proximity sensors) or triggers. Collected usage data are then compared with previous patterns of interaction stored in the user profile, in order to infer the so-called recurring patterns, which is the repeating patterns of interaction presenting the preferential uses of the system from the user. The authors defined a pattern of interaction as the sequences of activities which the user performs in some specific situation during the interactive execution of a task [25]. In the upcoming 3D user behaviors, whenever the system detects that the user is entering a recurrent pattern of interaction, it modifies the interaction properties of that pattern or performs some activities of that pattern on the behalf of the user [38]. The authors proposed to implement the model of interaction pattern using a

finite-state machine (FSM). The implementation process is based on the basis of an initial user profile as the starting point of the FSM, and the collected usage data to modify the interaction behaviors of the user.

The main advantage of this technique is that deduced interaction patterns present users' preferential usages to the 3D system. However, monitoring 3D interactions of each individual user shows the limitation that proposed interaction patterns are difficult to be used in a participatory and sharing environment such as an online social networking site (e.g. Second Life). This is since inferred interaction patterns are specific for each individual user rather than presenting the common usage of a group of users. In contrast, our work computes adaptive 3D interactions based on monitoring interaction traces of the crowd rather than of every individual user. By analyzing 3D interaction traces of the crowd, we produce simplified 3D interactions which can be used and shared for different users.

We have presented a review of the state of the art methods and related works in the literature to enhance online 3D interactions. Based on this revision, our method addresses the limitations of previous works by: (1) adapting Celentano and Pittarello's work to the context of 3D object browsing, and (2) monitoring interactions of the crowd to collect implicit user interaction traces. In the next section, we will first demonstrate our choice to use crowdsourcing for the context of the 3D web.

3.3 Crowdsourcing and Web3D

3.3.1 Why Crowdsourcing

We have presented in section 3.2 that most previous works compute adaptive 3D interactions by analyzing information traces of each individual user rather than of a group of users. Although these approaches do help users in interacting with online 3D content, they are still limited to the cases of new-in-3D users who are not familiar with 3D interaction techniques. In the web context, this limitation becomes more serious since there exists a numerous number of web users, who may be domain experts, but have marginal knowledge about 3D interaction techniques, and hence find it difficult to interact with 3D visualizations. Indeed, more approach has to be taken into consideration to support many common, non-expert users in visiting 3D websites.

One possible approach for these situations is to build an automatic system which generates the so-called the best views of the 3D objects – each best view presents an informative 3D region of the 3D object, then propose these views to the users as recommendations. Many research works have been proposed in the literature following this approach. For example, Vázquez et al., 2001 [83] proposed a measure, called viewpoint entropy, to compute the best view of the 3D shape. Viewpoint entropy can be interpreted as the amount of information of a scene which can be seen from a point. The authors proposed to use the projected area of all the visible triangles of the scene as the probability distribution of the amount of visible information in the scene to compute entropy. The best viewpoint of the 3D shape is the one at which the maximum entropy is obtained, i.e. at this viewpoint, the maximum information of the scene is visible to the users. Inspired by the concept of salient regions in an image that visually attract human attention, Lee et al., 2005 [65] introduced the idea of mesh saliency as a measure of regional importance for graphics meshes. The authors proposed to detect saliency on 3D mesh surfaces using a centered-surround operator on Gaussian-weighted mean curvatures. Then, the best view is computed as the one capturing the most salient attributes of the 3D objects, i.e. maximize visible saliency.

The works proposed by Vázquez et al., 2001 [83] and Lee et al., 2005 [65] are examples of various research works in the literature following the automatic approach to compute the best view of the 3D object. In the web context, this approach, however, is still limited since the computed best views do not guarantee to be the target of interest that the users are reaching. In other words, preferential uses of the systems from the user have not yet well-explored in this approach in order to suit well with the context of the websites where taken into account users' needs and preferences are essential. In our work, we propose to use *crowdsourcing* to overcome this problem of automatic approach. Our proposal shows two main points. First, crowdsourcing inherently poses the common preferential uses that many users have on a multimedia content. Second, crowdsourcing does not depend on the situation of each individual user since the crowdsourced data come from the collaboration of many users. These benefits of crowdsourcing are applied in online multimedia applications with two main uses. The first one is the use of crowdsourcing to perform online tasks that are difficult for automatic machines or that are requested by the enterprises who are interested in a task solved by humans (e.g. annotating the best textual description for an image containing Google's logo). The second use of crowdsourcing is to exploit the so-called collective intelligence of the crowd through their online collaboration. This approach triggers the wisdom of the crowd to identify interesting multimedia content by monitoring the behaviors of the crowd during their web visits.

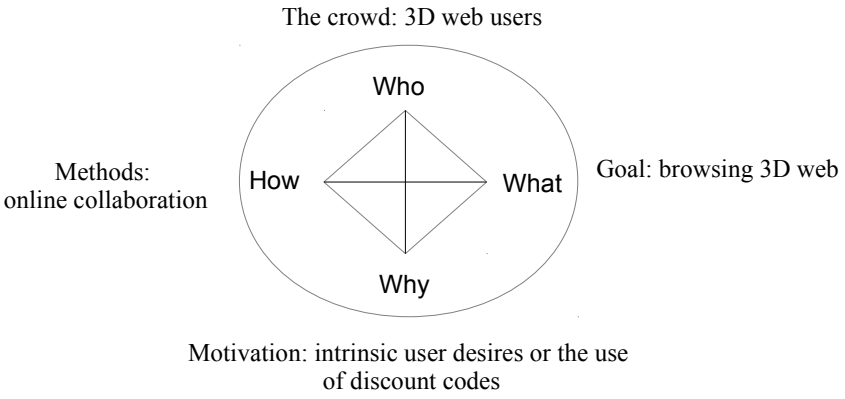

Figure 3.2: Adapted Conceptual Model of Crowdsourcing for the 3D Web.

In order to understand how crowdsourcing works in the context of the 3D web, in the following, we introduce a conceptual model of crowdsourcing we adapt from the work proposed by Malone et al., 2009 [70] for the case of the 3D web (see figure 3.2). As can be seen from the figure, the concept of crowdsourcing in the 3D web can be defined by four building blocks as follows:

- Who is the crowd: 3D web users who perform crowdsourcing tasks;
- Why is the crowd performing these tasks: this can be due to intrinsic desires to interact with online 3D content or the use of some promotion credits such as discount codes;
- What is the goal of the crowd: visiting 3D websites as a part of web browsing, so that to serve different purposes of life;
- Methods to perform the crowdsourcing task: the crowd collaborates together.

This conceptual model of crowdsourcing allows to identify what crowdsourcing is in the context of the 3D web. In the next step, our work wants to apply this concept of crowdsourcing in the 3D web and the two main uses of crowdsourcing in online multimedia applications to enhance of use of online 3D multimedia content. Specifically, in the first case, we propose to monitor 3D interactions of the crowd to identify interesting 3D regions of the 3D objects for simplifying online 3D interactions. This proposal will be presented in details in this chapter. In the second case, we propose to use crowdsourcing to perform an online annotation task that links a textual content with the corresponding 3D visualization for enhancing 3D browsing. This proposal will be presented in details in the next chapter of this thesis.

We have demonstrated our choice to adopt crowdsourcing to ease online 3D interactions. The next section gives in details our first use of crowdsourcing, that is, to monitor crowd behaviors for identifying informative 3D multimedia content.

3.3.2 Monitoring Crowd Interactions

This section outlines the previous research works that monitor the wisdom of the crowd to identify an interesting media content. We recall these research works and extend for our research.

Figure 3.3: Zoom and Pan viewing behaviors (denoted as red rectangles – the first row), detected ROIs within a video via crowdsourced zoom and pan actions (presented as heat maps with brightness of the pixels – the second row), and retargeted video frames suggested to subsequent users (third row) [34].

In the field of videos, Carlier et al., 2010 [34] proposed a paradigm to detect Regions of Interest (ROIs) within a video through crowdsourcing. The authors crowdsource the implicit viewing behaviors (zoom and scroll actions) from a large number of users using a zoom and pan interface. For example, in the first row of figure 3.3, the authors use red rectangles to denote viewports selected by users. One can notice that some users stay focused on the whiteboard, while others follow the speaker's motion. Viewports selected by users in each viewing session are stored in a database. Collected data are used to infer ROIs containing informative regions within the video. At the second row of figure 3.3, ROIs are presented as heat maps with brightness of the pixels – the brighter the color is, the more user interest there is. Detected ROIs are then grouped into shots and automatically reframed to generate the retargeted video for being replayed to subsequent users who would prefer a less interactive viewing experience (third row of the figure 3.3). By using a user study with 48 participants, the automatic retargeted video is proved to be comparable quality to one handcrafted by an expert user.

The work proposed by Carlier et al., 2010 [34] demonstrates two points. First, since users are the final consumers of the content, they are naturally the best possible "ROI detectors". The authors show that determined ROIs within the video are equal to the ones handcrafted by the experts of the field. Second, the results show that viewing behaviors of the crowd can be converged to an informative media content. By this work, the authors show this inference for the case of video contents. In our work, we want to extend the works done by Carlier et al., 2010 [34] targeting 3D content. In more details, we propose to crowdsource 3D viewing behaviors of the crowd to identify the ROIs within a 3D object. We first recall these two results from Carlier's works, that is, the crowd can be the best possible "ROI detectors" and the viewing behaviors of the crowd can converge to an interesting multimedia content, then extend to the context of online 3D graphics.

We have presented our motivation to use crowdsourcing for the context of 3D web. In the following, we will present our proposal to use crowdsourcing for solving the first problem of online 3D graphics, that is to simplify online 3D interactions.

3.4 Easing 3D Interactions with Crowdsourcing

We propose a pipeline to monitor 3D viewing behaviors of the crowd and generate recommendations and improvements to subsequent users by analyzing collected user traces. We then quantify by a user study the increase in performance of users given recommendations produced from crowdsourced interactions.

3.4.1 Proposed Pineline

Figure 3.4: 3D Models (left and middle columns) and example of ROI, called the stamp (last column).

In order to demonstrate our proposed pipeline, we first assume that we have some 3D models and each contains a ROI which users may want to see. For example, in

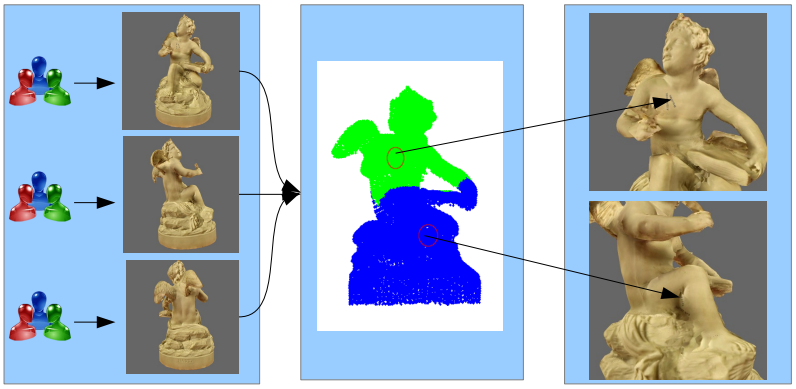

Figure 3.5: Proposed Pipeline: Crowdsourced 3D Viewing Interactions (first column), an example of two detected ROIs denoted as Red Circles (second column), Generated Recommendations (last column).

figure 3.4, we have two 3D models – a statue and a vase (left and middle columns) and each model contains a ROI – namely the stamp (third column). Each stamp presents the manufacture date of the model that users may be interested in. The following details our proposed pipeline which analyzes the interactions of the crowd to detect these stamps as ROIs.

The pipeline we propose includes three main parts: capturing 3D viewing interactions of the crowd, identifying ROIs by analyzing collected data, and generating recommendations to subsequent users based on detected ROIs (see figure 3.5). Basically, our pipeline can be defined as a learning process which allows to infer and generate additional meaningful visualization of 3D graphics, that is the interactive 3D recommended views (last column of the figure 3.5). A detail of the proposed pipeline is presented as follows:

- In the first part, we collect 3D viewing interactions from a set of initial users (first column). Collected data include *visible 3D points, camera position* and *normal vectors at visible 3D points* for each frame viewed by users. We define those visible 3D points as possible user look-at points during their 3D web browsing. After handling the gathering process, we send data back to the server and store them into a database. We also call this step the crowdsourcing part (see first column of figure 3.5).

- In the second step, we analyze the gathered data to determine ROIs, presenting the most informative regions of the 3D object (second column of figure 3.5). We identify ROIs as the parts on the surface of the 3D object which appear the most through crowd navigation. These 3D parts can be characterized as the 3D regions on the virtual 3D model that contain high density of the

user look-at points, i.e. of the 3D points logged from our crowdsourcing part. To determine 3D ROIs, we use a clustering technique to cluster the collected 3D points around the ROIs within the 3D object. The cluster containing the biggest number of logged 3D points is considered as the most informative region of the 3D object.

- After identifying the ROIs from 3D viewing traces of the crowd, in the last step (third column of figure 3.5), we compute the interactive 3D poses, each corresponding to a detected ROI, called *Recommended Views*, such that the ROI is at the center of the viewport. For example, the third column of figure 3.5 presents examples of two recommended views, each positions a stamp as an ROI at the center of the viewport. These computed recommendations are then proposed to subsequent users to ease their 3D browsing, that is to reduce 3D browsing time and to simplify 3D interactions.

In our proposed pipeline, we use the inference that if the ROIs are looked at more often than other regions, then the density of their logged 3D points will be much higher than that of other regions. In the second column of figure 3.5, we show two examples of detected ROIs marked by two red circles: each region contains a stamp (one on the chest and one on the knee of the statue), produced from this technique.

We have presented the main ideas of our proposed paradigm based on crowdsourcing to ease online 3D interactions. In the next section, we will introduce a system architecture which can be used to implement this proposed approach.

3.4.2 System Architecture

From technical point of view, our system acts as an extension for the existing 3D web application. Figure 3.6 presents the logic view of our system. As can be seen from the figure, our system receives 3D model and interaction traces as inputs to detect ROIs and compute recommended views. Having computed recommendations, our system produces simplified 3D interactions as outputs. Figure 3.7 shows the architecture design to implement our system. On the client side, we develop a *Listener* component which is embedded into the client side renderer and is used to capture 3D viewing interactions of the web users. We then send collected data back to the server and store them into *User Traces* database.

On the server side, the *Communication Module* acts as a communicator among the existing 3D web application, client requests and our system core (including ROI Detector and Recommendation Generator). With respect to the 3D web application

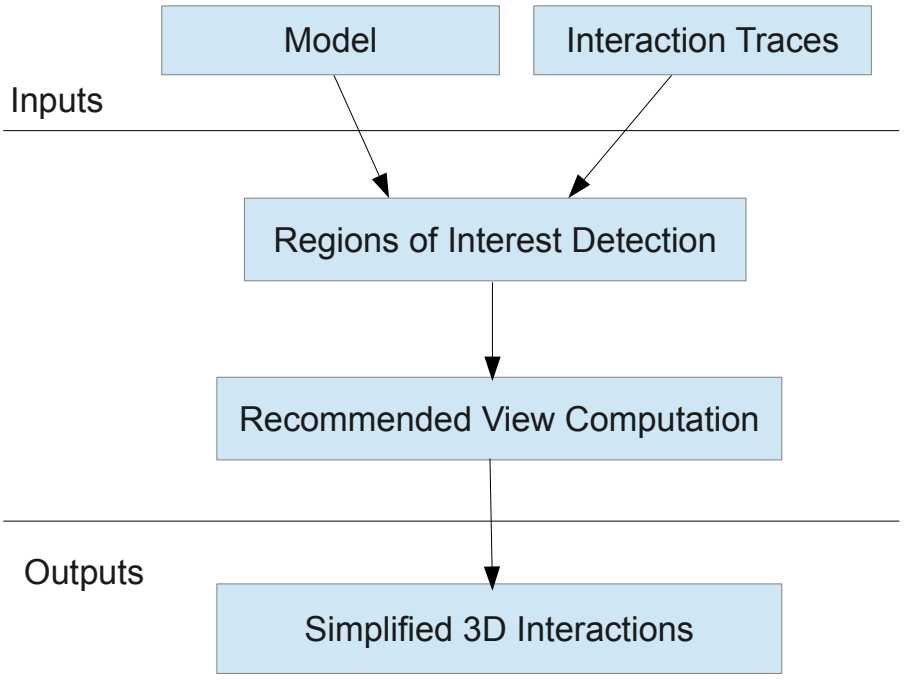

Figure 3.6: Logic View of Our System.

and users, our system acts as a web service, in which each time the user explicitly asks for a recommendation (via our clickable recommendation buttons), the *Communication Module* forwards user request to the system core to obtain the corresponding 3D recommended view. This recommended view is then presented as an additional meaningful 3D visualization to the user.

The main capabilities and functionalities of our system are developed in the *system core*, which contains the *ROI Detector* and the *Recommendation Generator*. Specifically, the *system core* is used to analyze collected user interactions and generate recommendations. The *ROI Detector* queries input data from user traces database, then uses a clustering technique to detect the ROIs which contain high density of the logged 3D points. The determined ROIs are then used as the inputs for the *Recommendation Generator* component in order to compute interactive recommended 3D views.

We have presented the logic view of our system according to the pipeline. We now show an implementation based on the state-of-the-art technical choices.

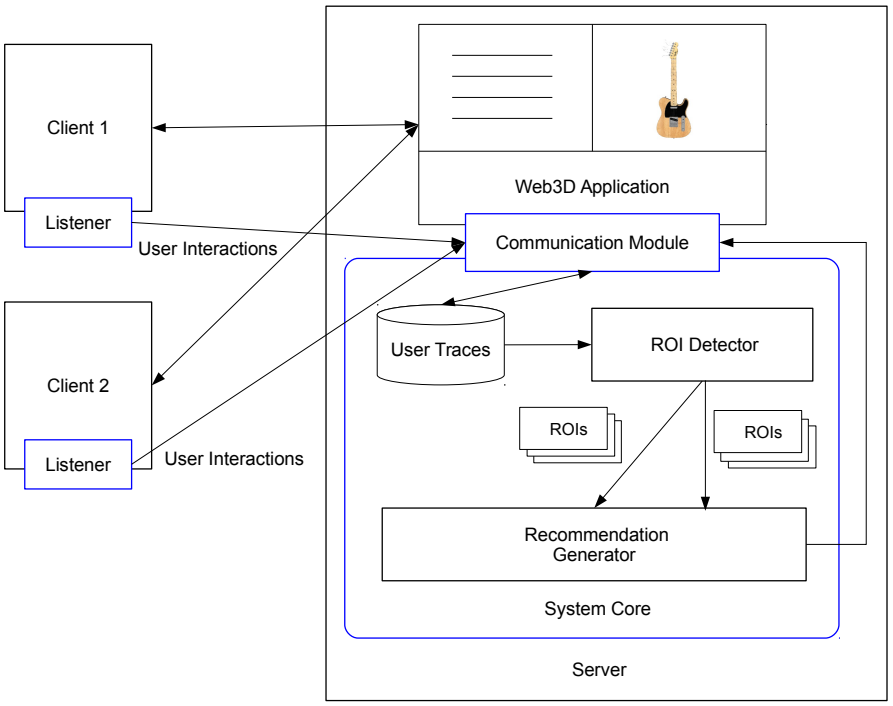

Figure 3.7: System Architecture.

3.4.3 Implementation

As presented in chapter 2, we choose to use plugin-free 3D web applications using X3D encoding format [23] and X3DOM rendering framework [24]. This choice is based on the study that X3D is the mainstream 3D web data on the Web nowadays and X3DOM framework provides a plugin-free rendering solution where X3D data are rendered and accessed as HTML5 DOM elements. In the following, we will present in details an implementation of the proposed pipeline using these technical choices.

Part 1: Collecting User Traces

In order to acquire input data for our system, we use a 3D picking buffer technique proposed by Behr et al., 2010 [29] to collect visible 3D points on each frame viewed by users. In more details, the technique renders the scene into a texture attached to a framebuffer object: the world coordinates of 3D points corresponding to 2D points on the canvas are encoded into the RGB (Red-Green-Blue) channels. Each time user views a frame, those 3D coordinates are retrieved and sent back to the server as collected user look-at points.

In each frame, we extract the camera position from the current view matrix and compute the distances between the camera and the visible points. We also use a shader to capture the normal vector at each visible 3D point.

Part 2: Regions of Interest Detection

After the crowdsourcing part, we have a set of 3D points on the surface of 3D objects, presenting user look-at points. With each point, we store its world coordinates in a 3-dimentional Euclidean space R^3. As presented in the proposed pipeline, we define ROIs as the highest density regions of these collected 3D points. To detect ROIs from these collected 3D points, we use a Mean-Shift density peak/mode detection [41] to cluster collected visible points around the ROIs. In this work, we choose to use Mean-Shift for its robustness. First, Mean-Shift has been proved to be a robust clustering technique that can handle data in the presence of outliers [41]. Since our crowdsourced data come from a large number of users, we do expect outliers in 3D interactions of the crowd. Second, Mean-Shift allows to cluster visible 3D points with an unknown ROIs. This is a nice benefit of using Mean-Shift in comparison with K-Means since we do not expect to know in advance the number of ROIs created by 3D interactions of the crowd. Given these reasons, the following presents how we apply Mean-Shift to detect ROIs from crowdsourced 3D points.

The basic Mean-Shift algorithm treats the points in the d-dimensional feature space R^d as a probability density function where dense regions in the feature space correspond to the local maxima or modes. Based on this idea, for each data point, Mean-Shift performs a gradient ascent procedure on the local estimated density until convergence. The stationary points obtained via the gradient ascent procedure represent the modes of the density function. The set of all data points that converge to the same mode defines a cluster. The output of Mean-Shift is a set of the clusters and the associated modes [44].

Given n collected 3D points $x_i, i = 1, ..., n$, in order to apply Mean-Shift clustering technique, we need to choose a Mean-Shift kernel K and determine how to select its kernel bandwidth h. There exists many types of Mean-Shift kernels. The two popular commonly used ones are *Flat kernel* and *Gaussian kernel* [36]. With respect to the Flat kernel, Mean-Shift iteratively moves the kernel to the mean position of all the points within its window radius defined by its bandwidth h. With respect to the Gaussian kernel, a similar procedure is implemented except that every point is first assigned a weight that decays exponentially as the distance from the center of the kernel increases. The main difference between the two kernels is that Gaussian kernel gives us a way to calculate a weighted mean. In our work, since we target to use our platform with real 3D objects on the 3D websites (e.g. e-commerce sites or virtual museums), we choose to use *Flat kernel* K, where we can simply determine

ROIs as the most crowded 3D points from logged data. The chosen *Flat kernel* K at the data point x is defined as follows [41]:

$$K(x) = \begin{cases} 1 & \text{if } \|x\| \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

Having the shape of the Mean-Shift kernel, the next requirement is to determine how to select the kernel bandwidth h. Generally, the bandwidth parameter needs to be no greater than the bounding box size of the data points and can be taken proportional to the bounding box size. In this work, we use this technique to select the bandwidth for the chosen Flat kernel $K(x)$. Specifically, we take the bandwidth value which is proportional to the bounding box size and small enough to ensure the convergence of Mean-Shift procedure. With these two input parameters (Mean-Shift kernel and its bandwidth), the kernel density estimate function $f(x)$ at the data point x is defined as follows [41]:

$$f(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h}) \tag{3.2}$$

The Mean-Shift procedure starts from a random data point, then converges to the stationary points of the kernel density function $f(x)$, called the *modes*. The output of the Mean-Shift algorithm includes the locations of the cluster centers (the modes), the clusters with their members (the 3D points), and the cluster size (the number of 3D points in the cluster). We take biggest clusters in the output of Mean-Shift procedure and set them as detected ROIs.

Part 3: Recommended View Generation

We compute recommendations such that the ROI is at the center of the viewport. The method is to set the camera position and orientation so that it looks towards the ROI center. To do so, we first compute the ROI center as the point of the collected data which is nearest to the cluster center (the mode). This is since each cluster center in our Mean-Shift procedure with the Flat kernel is also the centroid, i.e. the mean position of all the points within the cluster. We then extract the logged normal vector at ROI center and set it as the camera look-at vector. After that, we get the median of the collected distances between the ROI center and the camera. Using this median, the ROI center and the normal vector, we calculate the camera position. Having camera position and look-at point, we compute the camera orientation, then define each 3D recommended view using the associated camera position and orientation. Each time, there is a request for recommendation from

users, our system automatically moves the viewpoint to the recommeded view by setting the camera position and orientation to the new computed ones.

3.4.4 Simplified User Interface

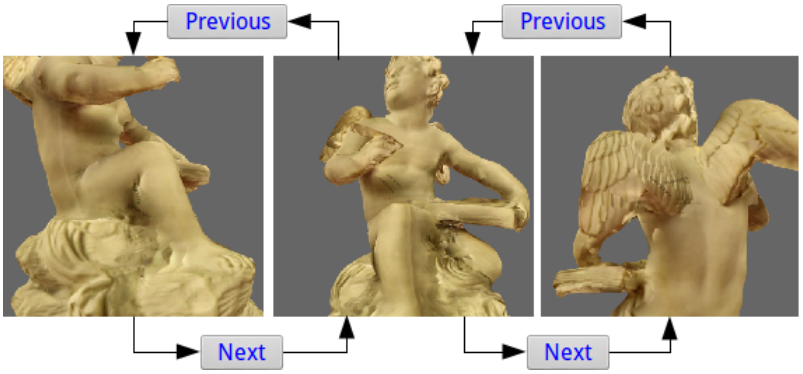

Figure 3.8: Simplified 3D User Interface.

After having the recommendations deduced from crowdsourced 3D viewing behaviors, we need to determine how to present these recommendations to users via user interface. We found in the literature a navigation tool supported in most VRML/X3D viewers, called a *viewpoint menu* [62]. This menu offers users a choice of viewpoints recommended by the author of the 3D website so that to ease user navigation with 3D objects – the user can access a viewpoint by simply selecting a menu item [62]. This viewpoint menu has become an important navigation tool in VRML/X3D web applications. However, it still shows the limitation that most contained viewpoints are static – each viewpoint is suggested by the author of the website and simple presents the image of an interesting 3D region. Jankowski, 2012 [58] performed an evalution of static 3D views versus animated 3D views in 3D web user interfaces. The results show that all users clearly preferred navigating in 3D using a menu with animated viewpoints than with static ones [62].

In our work, we follow this idea of using a viewpoint menu to simplify 3D user interface, allowing users to access an online 3D content via simple mouse-button clicks (see figure 3.8). Specifically, we propose a set of clickable buttons (Recommendation, Original, Next, Previous), positioned nearby the 3D rendering area. The user can explicitly ask for a recommendation from our system by simply clicking on these proposed buttons. Each time getting a request from the user, our system provides him/her an *interactive animated 3D view* – the *Recommended View*, presenting as the thumbnail to the user with the ROI at the center of the viewport (see figure 3.8). Our proposed Recommended Views allow users to capture the ROIs easily.

Figure 3.8 gives three examples of recommended thumbnails, presenting three ROIs on the statue, that are the stamps, on the knee (left column), on the chest (middle column), and on the wing (right column) of the statue, respectively.

Our simplified 3D user interface improves two main points in comparison with the static viewpoint menu supported in most VRML/X3D viewers. First, our proposed recommended views are interactive and animated – users can interact freely with the recommended 3D views using mouse interactions. These interactive and animated 3D views were evaluated by Jankowski's work, 2010 [58] to be more preferred by users than static 3D views. Second, our recommended 3D views are deduced from crowdsourced 3D viewing behaviors of web users rather than from the expert advice (the website author). It means that our recommendation poses more preferential uses of the 3D content from users than in the case of the static viewpoint menu.

We have presented our proposed pipeline and introduced a possible implementation. We now describe an experimental setup of the user study and provide an analysis of the results in order to evaluate our method.

3.4.5 Experimental Results

This section presents a user study and an analysis of the experimental results to evaluate the proposed approach.

Set-up of the Experiments

In this project, the user study aims to quantify the increase in performance of users given recommendations produced from crowdsourced 3D interactions. Since our work targets web applications having online 3D content to browse like e-commerce or virtual museums, we assumed that the context of the user study was in a *virtual museum* where users preferred to inspect some ancient statues. We chose two 3D models from 3D-COFORM project [1] (see figure 3.4) for the user study, called *ManStatue* and *VaseWhite*. *ManStatue* is a rather complex model whereas *VaseWhite* is a simple model, presenting a lot of symmetry. We assumed that each statue contains a *stamp*, presenting the manufacture date. Knowing the manufacture date of the stamp, users can get some knowledge about the history of the statue that they may be interested in. We then designed the protocol of the user study as follows:

In the preparation step, we first explained the interface of the user study with available interactions to users and let them try it by themselves on an additional 3D model. The interface allowed users to zoom, pan and rotate around 3D objects.

After that, we constructed a two-step user study, namely (1) crowdsourcing part and (2) pipeline evaluation part.

In the first step, we need to collect interactions of the first sample of users in order to generate recommended 3D views. We gave the users 6 models of the ancient statues, called *ManStatue*s (MS1, MS2 and MS3) and *VaseWhite*s (VW1, VW2 and VW3) respectively. Each statue has a stamp containing its manufacture date. Specifically the stamps are on the knee of *MS1*, on the chest of *MS2*, on the left wing of *MS3*, on the white part of *VW1*, on the yellow part of *VW2* and on the top of *VW3*. We told users what we expected from them, that is to browse the virtual 3D models to locate the stamps (each stamp is determined as a ROI). We then showed users what the stamp looks like on a 3D model. At this point, we want users to fully understand what is expected from them. Finally, we allowed users to participate in the real test with the six test models, in a random order to avoid some undesirable side effects. We recorded the time and interaction taken by users to complete every task, that is to successfully locate each stamp.

In the second step, we followed the exact same protocol as for the first step, except that the interface is supported with the computed recommendations deduced from the first step of the user study. The goal of this step is to allow us to properly evalute the pipeline, that is to quantify if recommendations deduced from the interactions of the crowd in the first step of the user study reduce 3D browsing time and ease 3D interactions. For this, we modified the interface used in the first step of the user study by adding a recommended button, nearby the 3D rendering area. Users could press the proposed button to explicitly ask for a recommendation. The system also supports the next and previous buttons so that users can browse between the different recommended views. We also recorded the time and interaction taken by users to complete every task like in the first step of the user study.

This user study was performed on a total number of 28 users, 18 of them being part of the crowdsourcing step and 10 others being part of evaluating the pipeline; 17 users were male, 11 were female, aged from 23 to 35, mostly from the university community.

We have presented the protocol of our user study. We now provide an analysis of the results in order to evaluate our method.

Effectiveness of Recommendations

Table 3.1 shows the average time taken by the users to locate the stamps on the 3D models at each step of the user study. As can be seen from this table, in the first step of the user study, it generally took the users around one minute to reach the

	MS1	*MS2*	*MS3*	*VW1*	*VW2*	*VW3*
Step 1	76	48	67	53	50	54
Step 2	11	9	8	7	8	6

Table 3.1: Average time (in seconds) taken by users to complete tasks, for each step of the user study.

	MS1	*MS2*	*MS3*	*VW1*	*VW2*	*VW3*
Step 1	192	108	153	128	126	133
Step 2	29	22	20	17	21	15

Table 3.2: Average number of mouse events used by users to complete tasks, for each step of the user study.

ROI (the stamp) using a traditional 3D user interface. In contrast, in the second step of the user study, where recommendations are added to the 3D user interface, users were taken only around nine seconds to reach such ROI. The following provides an analysis of this difference.

Previous studies [64, 74, 75] show that in the web context, users do not prefer to spend time on learning what the websites are and how to interact with them – they often prefer to be capable of directly consuming the content of the websites. King, 2003 [63] in his book about website optimization, named "Speed up your site" has shown that web users will wait for a maximum time of *8.6* seconds to see a loaded content. These research findings rise a notification that users will not wait too long to determine an interesting 3D content while browsing the 3D web. It means that in the first step of our user study, it seems too long for web users to spend around one minute (60 seconds) on manipulating the 3D interactions to find a meaningful 3D region. In contrast, in the second step of the user study, it seems reasonable to take web users around 9 seconds to interact with the virtual 3D models for locating a ROI. This result presents that our recommendations improve 3D web browsing, that is to reduce their 3D browsing time.

Table 3.2 shows the average number of mouse events (including mouseClick, mouseUp, mouseDown and mouseDrag events) used by users to complete tasks for each step for the user study. As can be seen from the table 3.2, in the second step of the user study, our recommendations allow users to use less interaction effort (less mouse events) to accomplish the task of locating the ROI than in the first step of the user study where having no recommendations. This result shows that our recommendations help to simplify 3D interactions, that is, to satisfy user desires of not spending too much time on learning how to interact with 3D websites, as indicated by previous research works [64, 74, 75].

Convergence of the Crowd

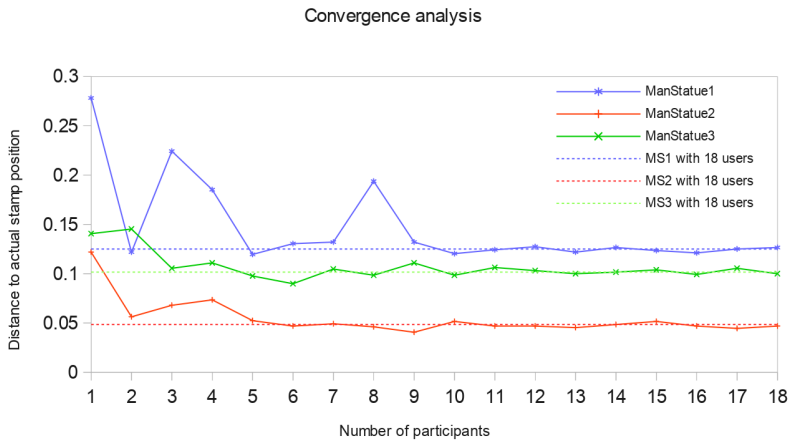

Figure 3.9: Convergence Analysis of the Crowd Interactions.

Since we have presented earlier that our work is inspired by the works done by Carlier et al., 2010 [34], that is, 3D viewing behaviors of the crowd can converge to an interesting 3D region. We then need to identify a measure that allows us to evaluate the convergence of 3D viewing behaviors of the crowd. As we have presented in tables 3.1 and 3.2 that our recommendations do help users in reducing 3D browsing time and using less 3D interaction effort to complete the assigned tasks. These results show that our method succeeds in determining the locations of ROIs (the stamps) by analyzing the interaction traces of the crowd. Having this findings, we still need to evaluate if the quality of recommendations reaches a stable value when increasing the number of users in the crowd. To reach this goal, our method is to compute the correctness of recommendations, depending on the number of users taken to compute the recommendations (see figure 3.9). In order to compute the correctness of a recommendation, one should notice that in our method the correctness of a recommendation is linearly correlated with the correctness of its center since we compute recommendations such that the ROIs are at the center of the viewport. For this, we compute the correctness of recommendation center and use that to evaluate the correctness of the recommendation. The correctness of recommendation center is defined as the distance between the position of the ROI and the actual position of the stamp. By performing this computation on different numbers of users taken to compute recommendations, we have a diagram as shown in figure 3.9 which can be used to analyze the convergence of the crowd interactions. Before analyzing the results, we first present a detail of this computation as follows:

We select n random users out of 18 users, then compute the recommendations using only the interactions of those n users. The output of our method is a 3D point (corresponding to the mode of the biggest cluster out of the Mean-Shift algorithm).

We can then compute the distance between this 3D point and the actual 3D position of the stamp (see figure 3.9). By taking a high number of permutations and averaging the distances we get a point on the diagram.

As can be seen from this figure, the results show that the correctness of the recommendations, i.e. the correctness of ROI positions, fluctuates at around several test users (from 2 to 8 users), then reaches a stable value at approximately 10 users where we can get the same precision for the recommendations as we have with 18 users (represented by the horizontal lines). Using this result and the ones presented in the tables 3.1 and 3.2, we can conclude that 3D viewing behaviors of the crowd can converge to an interesting 3D region.

Handling Transferred Data

	Avg bw with JSON	*Avg bw with compressed data*
ManStatue	60KB/s	24KB/s
VaseWhite	90KB/s	26KB/s

Table 3.3: Average Bandwidth (Avg bw) to transfer logged 3D Points.

In order to evaluate performance of our proposed pipeline, it is important to handle the amount of crowdsourced data transferred from client to server. To reach this goal, we use a network bandwidth monitoring program, namely *bmon* [1] to measure the average bandwidth used by our system to transfer collected 3D points during our experiments.

Table 3.3 shows the results of our measurement. As can be seen from the table, the results show that with the use of JSON standard to encode 3D data, our system uses an average bandwidth of 60KB/s for manstatue models and 90KB/s for vasewhite models. On the other hand, with the use of a compression algorithm such as LZW (Lempel–Ziv–Welch) [85] to compress transferred data, our system uses an average bandwidth of 24KB/s for manstatues and 26KB/s for vasewhites.

From these results, we see that our system should work well for LAN connections where bandwidth is over 100Mb/s. However, for more popular Internet connections (e.g. with ADSL network where upload bandwidth is 1Mb/s), an optimization is required for the case of JSON standard. In this spirit, one possible solution is to use binary encoding format to encode 3D points. With this solution, we can reduce the average bandwidth for manstatues to 17KB/s and for vasewhites to 25KB/s. This reduction should ensure the use of our pipeline in both LAN and Internet.

[1] https://github.com/tgraf/bmon

Complexity of Proposed Pipeline

As presented previously, our proposed pipeline consists in three main phases: (1) crowdsourcing 3D interactions, (2) data analysis, and (3) recommended view generation. In the crowdsourcing phase, the complexity of our approach relies on handling the amount of logged data transferred via the network. We currently use JSON standard to encode 3D points and perform our experiments with LAN connections. For this, the complexity of our proposed approach in the crowdsourcing part retrieves a constant value. This is since our system works well with LAN connections using JSON standard.

In the analysis phase, the complexity of our proposed pipeline relies on the complexity of the Mean-Shift algorithm. Specifically, our Mean-Shift algorithm starts from a random data point and performs a gradient ascent procedure which iteratively moves the kernel to the mean position of all the points within its window radius. This procedure is repeated to all unvisited data points. For each iteration of Mean-Shift, the computational cost is $O(n^2)$, where n is the number of collected 3D points [36]. From this computation, we get the complexity of our approach in the analysis phase is $O(Tn^2)$, where T is number of iterations in the Mean-Shift algorithm. In the last phase, the complexity of our proposed pipeline has a constant value since we simply respond the computed RVs from server to client according to user requests.

Comparision with Previous Experiments

One similar set of experiments was performed by Levoy et al., 2002 [79]. In more details, the authors proposed a new 3D model acquisition pipeline that allows users to freely rotate an object by hand and see a continuously-updated model as the object is scanned. By using this setup, the proposed system captured instant feedback from users about the presence of holes and the amount of surface that has been covered in the scanning process. This real-time feedback is then used to ease the upcoming process to scan objects in less time and cost. The main advantage of this setup is that it exploits freely 3D interactions from users to detect holes and covered surfaces during scanning process. One can notice that this setup does not take into consideration the interest of users towards the manipulated 3D object. This is a noticable limitation since user interests are often useful for the generation of 3D content used in the web context. In contrast, our setup tackles this issue by detecting ROIs from the interactions traces of the crowd in order to generate recommendations for upcoming web users.

3.5 Summary and Perspectives

In this chapter, we propose a new paradigm to enhance online 3D interactions with virtual 3D models using *crowdsourcing*. Our proposed pipeline consists of analyzing 3D user interactions to identify Regions of Interest (ROIs), and generating recommendations to subsequent users. With an user study, we validate our method by quantifying the effectiveness of recommendations, that is to reduce 3D browsing and simplify 3D interactions, and qualifying that 3D viewing behaviors of the crowd can converge to an interesting 3D region.

This work can be improved in several directions. The first one is to define the optimal number of recommended views, i.e. the optimal number of ROIs within the 3D object to be presented to users. This is interesting since presenting to users so many recommendations may cause undesirable side effects (e.g. users may be tired of clicking on too many buttons to ask for recommendations). The second one is to identify the accurate moment to present the recommendations to users. This is helpful since Baccot et al., 2009 [27] shows that timing ad banners on e-commerce sites has an impact to the level of user interests during their web visits. Another direction is to add learning time of users to get familiar with 3D interactions supported in the websites. Having this learning time of users, one can measure the convergence of the crowd not only in term of correctness of recommendations, but also in term of time. Moreover, this experiment should be undertaken with a set of regular users to avoid the bias due to experiments taken by students in university community.

One may notice that in the setup of our experiments, we use simple 3D models and a small set of users to evaluate the recommendations. This is since our experiments were mostly performed in university community with volunteer participants. A setup on real websites with a large number of web users and various types of 3D objects could produce more valuable recommendations as well as help to improve the evaluation of our proposed method (e.g. in which time and with which types of 3D objects, the crowd could converge easily and accurately, or how the age and gender of users may affect the quality of recommendations). Another point is that in an experiment, crowdsourcing does converge rapidly, verifying this fact that on a large number of models for a non biased crowd should be an interesting perspective.

In our work, crowdsourcing would converge to the most frequent 3D user interactions. Due to this, rare user profiles like novice or very expert users may not find our recommendations helpful. One direction to overcome this limitation of our proposed approach is to construct a user profile for each individual user. Having this con-

structed user profile, for rare users, our system could support their web browsing by proposing them personalized interactions. Meanwhile for common users, our system provides them crowdsourced recommendations due to the most commonly used 3D interactions.

We have presented our first contribution of using crowdsourcing to simplify 3D interactions, in the next chapter, we will present our second study to semantically link text and 3D visualization for aiding new users in browsing 3D web content.

Chapter 4

Linking Text and 3D for Enhancing 3D Browsing

Contents

This chapter studies the second problem of using online 3D graphics where 3D interactions are too different from the mainstream web navigation model – hypertext. We begin the chapter with a detail description of the problem. Then, we study the state of the art methods and related works in the literature to solve the problem. Finally, we present our proposed method using *crowdsourcing* to link online text with 3D visualization for enhancing 3D browsing.

4.1 Motivation

We have presented in the chapter 2 that the ability to visualize and manipulate 3D content in real time seems to be the next evolution of the Web for a wide number of application areas such as e-commerce, education and training, architecture design, virtual museums and virtual communities. The use of online 3D graphics in these application domains does not mean to substitute traditional web content of texts, images and videos, but rather acts as a complement for it. The Web is now a platform where hypertext, hypermedia, and 3D graphics are simultaneously available to users.

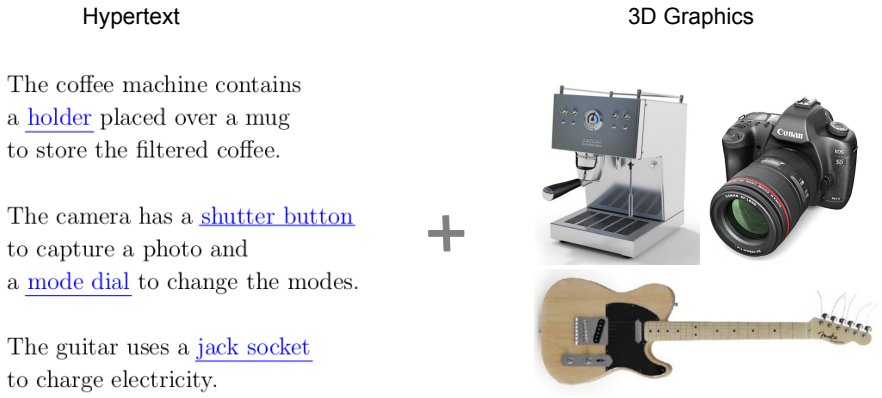

Figure 4.1: Navigation Mediums on the Web: Hypertext (left) and 3D Graphics (right) on the same web page.

This use of online 3D graphics, however, poses specific issues: 3D graphics contain purely rich visual information of the concepts (e.g. the 3D visualizations of the coffee machine, the camera, and the guitar in the right column of the figure 4.1). On the other hand, traditional websites mainly contain descriptive information (text) with hyperlinks as navigation means (e.g. the textual descriptions of the coffee machine, the camera, and the guitar in the left column of the figure 4.1). The problem is that viewing and interacting with the websites that use two very different mediums (hypertext and 3D graphics) can be complicated for users: they need to handle text browsing to look for general information (e.g. reading through the whole textual descriptions of the coffee machine, the camera and the guitar in the left column of the figure 4.1), and text searching for more specific information. On the other hand, users also need to manage how to interact with 3D visualizations (e.g. navigating around the 3D space or examining and manipulating virtual 3D objects) in order to gain a better understanding of the data. This separation of interactions between the two modalities (text and 3D graphics) requires too much effort from users to browse the site.

For example, in figure 4.1, the user can use one of the hyperlinks (e.g. shutter button) to access more specific information about the shutter button of the camera (left column). In this case, user is usually forwarded to another document which contains further textual information of the shutter button. If user wants to inspect the 3D model of the camera and identify how the shutter button looks like in real life, he/she will need to come back to the current web page and use the supported 3D interactions (e.g. Rotate/Zoom/Pan) to locate the feature. The problem in this situation is that the same information about the shutter button of the camera is presented to user separately in two very different mediums (descriptive text and 3D visualization). Users, hence, have to manage a shift back and forward between hypertext and 3D graphics in order to gain a better understanding of the data.

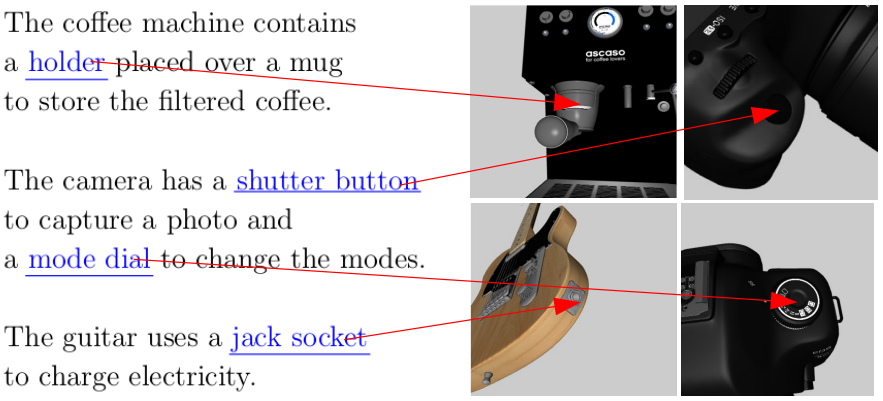

Figure 4.2: Linking Text and 3D: a semantic link, denoted as red arrows, associates a textual name with its *possibly visible* position on the 3D model (e.g. the jack socket of the guitar).

These problems raise a new challenge to establish a link between hypertext and 3D graphics to be used on the same webpage. However, the question for this challenge is that how this link should look like and how to technically implement it. In our work, we recall the idea of hypermedia, that is, texts are linked with images for web browsing, and extend it for the context of 3D websites. Specifically, we first define this link as semantic associations between textual descriptions and 3D visualizations, then determine the task of creating this link as an *online 3D annotation task*. Figure 4.2 shows examples of how these links look like – they are presented as red arrows where users are allowed to select a text for readily locating the associated 3D visualization on the 3D model (figure 4.2). Semantic associations aim to help gathering knowledge about a 3D object and ease browsing its 3D model. In order to create such semantic associations between textual descriptions and 3D visualizations, we propose to use *crowdsourcing*. The next paragraph details the reasons why we choose crowdsourcing as the proposed approach.

As presented in chapter 3 that the recent rapid developments of the Participatory Web (Web 2.0) and the increasing numbers of web users all over the world have triggered the rise of crowdsourcing in online multimedia applications with two main uses. The first use of crowdsourcing has been presented in the chapter 3, that is, the monitoring of the viewing behaviors of the crowd can converge to an interesting multimedia content. The second use of crowdsourcing in online multimedia applications is to perform online annotation tasks that are difficult for automatic machines or that are requested by the enterprises who are interested in a task solved by humans.

Salek et al., 2013 [81] demonstrated that tasks such as annotating an image with the best textual description, or determining whether a specified object is located in an image are inherently demanding on the Web. For instance, Google can request web users to perform the task of finding a best description for an image containing Google's logo, or Facebook can request its users to contribute to the task of locating the most famous person in an image of many people. These types of tasks can be performed easily by humans but are difficult for automatic machines since they require not only image processing capabilities, but also a deep understanding of the image, including social, cultural or geographical knowledge. In contrast, crowdsourcing proposes to be an effective solution to perform these types of annotation tasks in online image applications. Moreover, crowdsourcing marketplaces, such as Amazon's Mechanical Turk [33] or Microworkers [51], are bringing together requesters who are interested in a task solved by humans, and workers who are willing to perform the requested tasks for a payment.

In this work, we are motivated by the rise of crowdsouring to perform online annotation tasks in the fields of images and videos. These types of tasks are generally easy for humans but difficult for automatic machines. Similarly, annotating online textual name and 3D visualization is also not an easy task for automatic machines, although it can be performed in some cases like the work proposed by Russell et al., 2013 [80]. This is since matching a textual description with a proper 3D location requires not only 3D shape analysis capabilities, but also a deep understanding of the shape features such as their semantic meaning or their contextual name (e.g. used in scientific research or in daily life).

Moreover, previous research works show that crowdsourcing is an effective way to perform online annotation tasks in the fields of images and videos. For example, Salek et al., 2013 [81] used crowdsourcing to perform object localization task, that is an image annotation task to find the location of a target object in an image. Empirical evaluations showed that crowdsourced data are successfully used to estimate the true locations and to rank participants by their ability. Finin et al., 2010 [48] proposed to use both Amazon Mechanical Turk [33] and CrowdFlower [6] to annotate the so-called named entities for Twitter status updates – each named

entity is an atomic element in text belonging to predefined categories such as the names of persons, organizations or locations. The experimental results showed that the crowdsourced annotations of named entities are useful in domains such as Facebook and Twitter. This success of crowdsourcing to perform online annotation tasks in the fields of images and videos also inspires our choice to use crowdsourcing for annotating semantic association between text and 3D.

4.2 State of the Art

Our proposed approach is related to *Semantic Link Creation Technique*, which can be classified into three categories: (1) Virtual Hyperlink Integration, (2) Hypertextualized VEs Creation, and (3) Textual 3D Annotation. In the following, we will present in more details each of these categories.

4.2.1 Virtual Hyperlink Integration

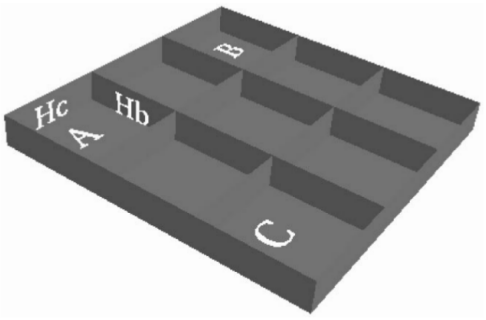

Figure 4.3: Examples of virtual hyperlinks in 3D VEs – H_b and H_c: H_b allows users to teleport from room A to room B, and H_c allows users to teleport from room A to room C in the virtual building [78].

Hyperlinks in 3D VEs, also called virtual hyperlinks, allow users to teleport from places to places in a virtual 3D space. The interaction structure is similar to the case of hyperlink in texts and images, that is, parts of a virtual space are connected together in a non-linear order. For example, in figure 4.3, there is a virtual building that contains three rooms A, B, and C. The system allows users to move instantaneously (to teleport) from room A to room B using the virtual hyperlink H_b and from room A to room C using the virtual hyperlink H_c [78].

Ruddle et al., 2000 [78] studied the effects of virtual hyperlinks on navigation in 3D environments. The authors demonstrated that virtual hyperlinks propose the same advantages as the case of hypertext such as reducing navigation time and allowing the design of flexible layouts. However, the use of virtual hyperlinks often creates cognitive problems for users such as they lose their sense of location and direction after being teleported by virtual hyperlinks. Moreover, most 3D systems do not support a tool to keep track of history to return to previous locations before following virtual hyperlinks [47]. This makes the use of virtual hyperlinks become difficult for users.

Eno et al., 2010 [47] studied the linking behavior in a 3D environment. The authors examined explicit landmark links and implicit avatar pick links in Second Life [16]. The goal is to determine if linking patterns in virtual worlds corresponded to linking behavior in the traditional web of hypertext and hypermedia. Landmarks, which are inventory items containing a description, a target location, and an optional location, are shared by users to link locations in Second Life. Avatar profile picks, which allow to add a list up to 10 favorite locations, are linked to favorite places that the avatars can add to their profiles. Landmarks are similar to shareable bookmarks in traditional web, which direct users to new locations, creating a link from one space to another space. While picks are tied to the avatar rather than specific locations. The authors found out that although link graph in 3D environments is more sparse than in the traditional web, the underlying structure is similar. Moreover, the results from the user study showed that linking is valued by users, that is, they followed the links in the Second Life environment, and making linking easier can help to provide a richer user experience.

The main advantage of virtual hyperlink technique is that it inherites the benefits of hypertext for 3D VEs such as reducing navigation time and number of 3D interactions. However, the limitation of this approach is that it makes users get lost in virtual space since they are teleported instantaneously from one 3D place to another 3D place. In contrast, our method does not cause users to face this problem since the recommendation is displayed to users via a continuous viewpoint movement from the current viewpoint to the recommended one.

4.2.2 Hypertextualized VEs Creation

In order to associate hypertext and 3D graphics, Jankowski, 2011 [56] proposed to develop integrated web information spaces called “Hypertextualized Virtual Environments” (HiVEs), where hypertext, hypermedia and 3D media are simultaneously available and linked. The main idea behind HiVEs is that users can gain a lot more

online information in a space that combines the capabilities of both hypertext and 3D graphics. The author demonstrated that there should be a design of user interface that supports both the independent and integrated exploration and comprehension of HiVEs. The design should pair interactive 3D graphics with well-established user interface conventions of the traditional web so that users can find it easy to complete the web tasks [59].

Inspired by the idea of having such a HiVE web information space, Jankowski, 2011 [57] introduced a so-called "taskonomy of 3D web use" – a taxonomy of web tasks and 3D tasks, which identifies the main browsing tasks the users may have when visiting the HiVE. Examples of such tasks are Locate on Page and Go to Page for web tasks, and Navigation and Manipulation for online 3D tasks. The authors demonstrated that a user interface designed for HiVEs should support the main tasks defined in this taskonomy of 3D web use, so that users can still browse text for descriptive information as well as use 3D interactions to inspect the virtual 3D objects for better understanding of the data [62].

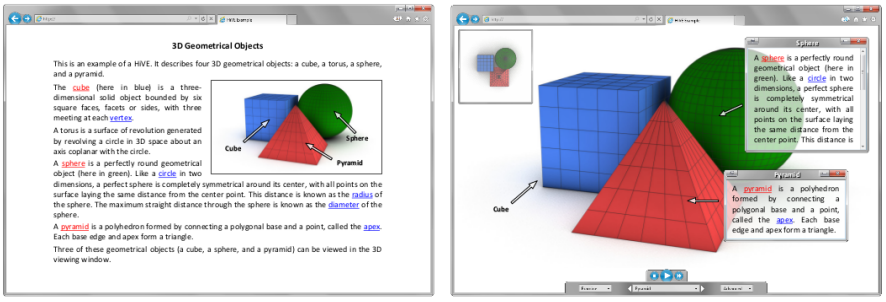

Figure 4.4: Dual-mode User Interface for HiVEs proposed by Jankowski and Decker, 2013 [61]: Hypertex Mode (left) and 3D Mode (right).

Based on the constructed taskonomy, Jankowski and Decker, 2012 and 2013 [60, 61, 62] introduced a so-called Dual-mode User Interface for HiVEs that has two modes between which a user can switch anytime: the *Hypertext Mode* and the *3D Mode* (figure 4.4). In the Hypertext Mode, the hypertext is the primary information carrier and 3D graphics are embedded into the webpage to provide additional visual information and 3D interactions (see the left column of the figure 4.4). On the other hand, in the 3D mode, the situation is reversed in which 3D graphics are the main information carrier and the hypertextual annotations are immersed into the 3D scene (see the right column of the figure 4.4). The authors indicated that this Dual-mode User Interface is capable of highlighting the informative information (in the hypertext mode) or the visual information (in 3D mode) depending on the choice made by the users. With the experimental user studies, the authors showed that users performed the web tasks better with the Dual-mode User Interface for HiVEs

rather than alternatives such as Hypertext User Interface Only or 3D User Interface Only [61].

Hypertextualized VEs technique shows the advantage that it allows users to easily switch between different mediums (hypertext, hypermedia and 3D media). However, the limitation of this idea is that it is hard to be practically implemented since it is not easy to change the current information infrastructure of the Web to the new one. In contrast, our work employs crowdsourcing as a practical solution to generate the semantic link while preserving the same information infrastructure of the Web (with of the use of hypertext as the mainstream web navigation model).

4.2.3 Textual 3D Annotation

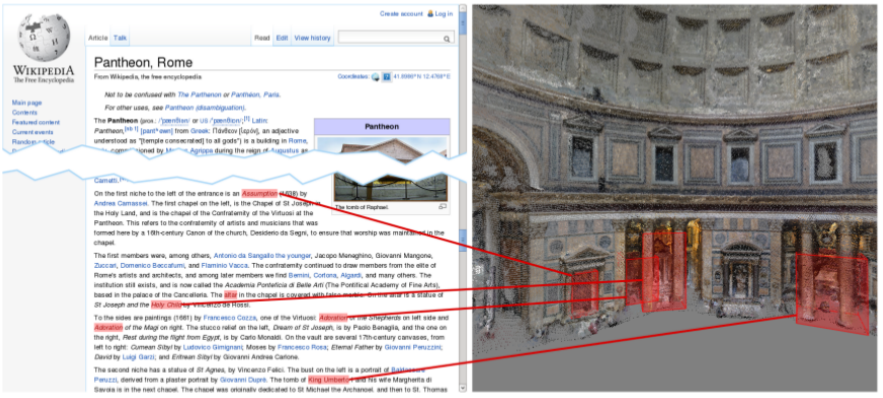

Figure 4.5: An Automatic Approach to link Online Text and 3D Visualization [80].

The method allows to semantically associate a metadata (a textual description) with a part of 3D objects. An automatic approach to perform textual 3D annotation was done by Russell et al., 2013 [80] (see figure 4.5). The authors proposed an algorithm to generate interactive visualizations (presented as red arrows in the figure 4.5), that link text sources with photorealistic 3D models to enhance 3D browsing. Specifically, the algorithm consists of three steps. Firstly, it downloads a set of images from Flickr [9] using a given reference text, then automatically reconstructs a 3D model from the downloaded Flickr photos using the freely available visualSFM package [19] and the patch-based multiview stereo (PMVS) algorithm, proposed by Furukawa and Ponce, 2010 [49]. Having the 3D reconstruction, in the second step, the algorithm identifies the correspondences between regions on a 3D model to textual descriptions/names. Specifically, it finds all possible noun phrases in the given website as candidate textual names, then determine possible 3D locations of these candidate textual names on the 3D model via query expansion [40]. The

method involves image search and text matching verification for each possible noun phrase to find text that actually names objects and link it to the reconstructed 3D geometry.

Having the correspondences between texts and 3D visualizations, in the last step, the algorithm filters a large number of false detections by training a classifier over features obtained from the text and the output of query expansion. The proposed user interface added with computed semantic associations (see figure 4.5) enables coordinated browsing of the text with visualization, that is texts can be selected to move the camera to the corresponding 3D visualizations, and 3D bounding boxes provide anchors back to the texts describing them. By the use of experiments on multiple sites, Russell et al., 2013 [80] showed that the proposed semantic associations provide innovative navigation experiences for users when browsing the websites having both the texts and the 3D models.

The automatic approach proposed by Russell et al., 2013 [80] uses the idea of mining text and photo co-occurences across the Internet as well as applies the algorithms in the domain of 3D photography for 3D reconstruction and identification of correspondences between text and 3D locations. The interesting point in this approach is that it produces a very intuitive interaction between text and 3D visualization. However, this approach also shows certain limitations. First, it requires huge repositories of sharing images (e.g. Flickr) to acquire input data. Second, it involves the 3D reconstruction from images to generate semantic links. Third, the process to associate text and 3D visualization is static. In contrast, our work creates manual textual 3D annotation using *crowdsourcing* and addresses the limitations of Russell et al's work. First, we adapt Russell et al's work to the context of given 3D objects (like in the case of e-commerce) with no 3D reconstruction from images involved. Second, our generated semantic links are adaptive to the expertise of the crowd in 3D interactions. Specifically, we get more accurate links for easy features than for technical ones. Moreover, we show that an integration of crowd feedback and opinions did improve the quality of semantic links and recommendations, particularly for technical features. In the next section, we will present in more details our crowdsourcing method to create semantic links.

4.3 Associating Text and 3D via Crowdsourcing

We propose an approach using *crowdsourcing* to generate semantic associations between texts and 3D visualizations. We then quantify by a user study the increase in performance of users given produced links both in terms of efficiency and correctness.

4.3.1 Proposed Approach

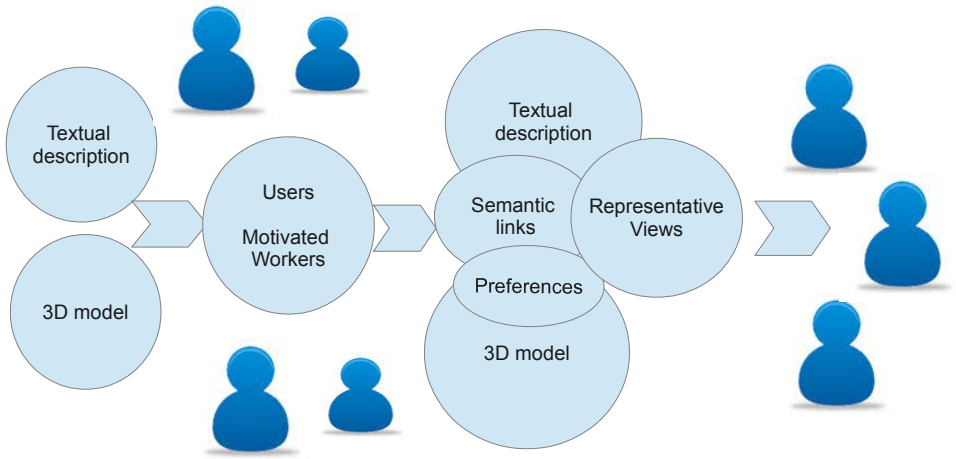

Figure 4.6: Our idea to use Crowdsourcing.

Figure 4.6 presents our idea to use crowdsourcing for building semantic associations between text and 3D visualization. In more details, having inputs of textual description and 3D model (left part of the figure), we employ the participations of motivated workers in the crowd (middle part of the figure) in order to generate semantic links and access user needs and preferences (right part of the figure). Having semantic links and user preferences, we compute representative 3D views and suggest to upcoming users.

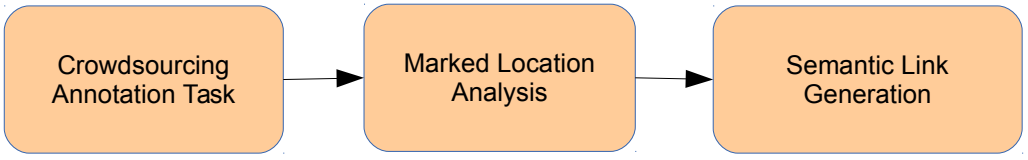

Figure 4.7: Approach Overview.

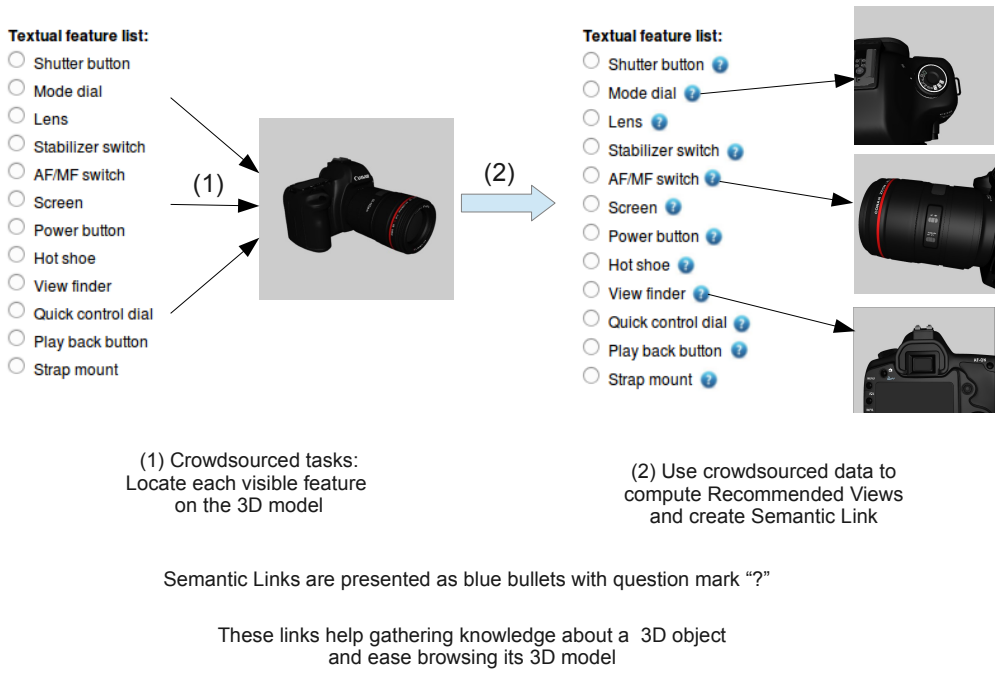

Figure 4.8: Proposed *Crowdsourcing* Approach to build Semantic Association.

Figure 4.7 gives an overview about our crowdsourcing approach which is divided into three main parts: (1) crowdsourcing the annotation task of locating a textual feature on the 3D model (left column), (2) analyzing crowdsourced data to identify the location of corresponding 3D visualization (middle column), and (3) generating semantic link, that allows users to readily locate a 3D visualization associated with a textual content (last column). A detail is presented as follows:

- In the first part (see left part of figure 4.8), we assign users the explicit annotation task in which they can browse the 3D model to locate a given textual feature. The task indicates that when users find the location of corresponding 3D visualization, they should spot it by a red dot, also called *marked point*. Our system then collects the 3D locations of marked points from many users and store them into a database.

- In the second step (see right part of figure 4.8), we analyze the crowdsourced locations of marked points to determine the location of 3D visualization associated with a textual content. Our method is to aggregate the answers from multiple users in order to determine the so-called *convergent* marked-point, that is, the point from the collected data which presents the most recognized location of 3D visualization marked by the crowd. Then we define this convergent marked-point as the detected location of the corresponding 3D visualization.

- In the last part (see right part of figure 4.8), we generate semantic links between textual features and detected locations of 3D visualizations. This includes two sub-parts. First, we present the 3D visualization as an interactive 3D view, called recommended 3D view, such that the detected location is at the center of the viewport. Then, we associate this recommended 3D view with the corresponding textual name and stores them into the database. The produced semantic associations are then suggested to subsequent users so that they can readily locate the 3D visualization associated with a textual content. We also propose to collect user feedbacks and opinions on the relevance of semantic links so that to assess and evaluate the quality of the proposed semantic association.

We have presented the main ideas of our proposed approach using crowdsourcing to build semantic association between text and 3D graphics. In the next section, we will introduce a system architecture which can be used to implement this proposed approach.

4.3.2 Implementation

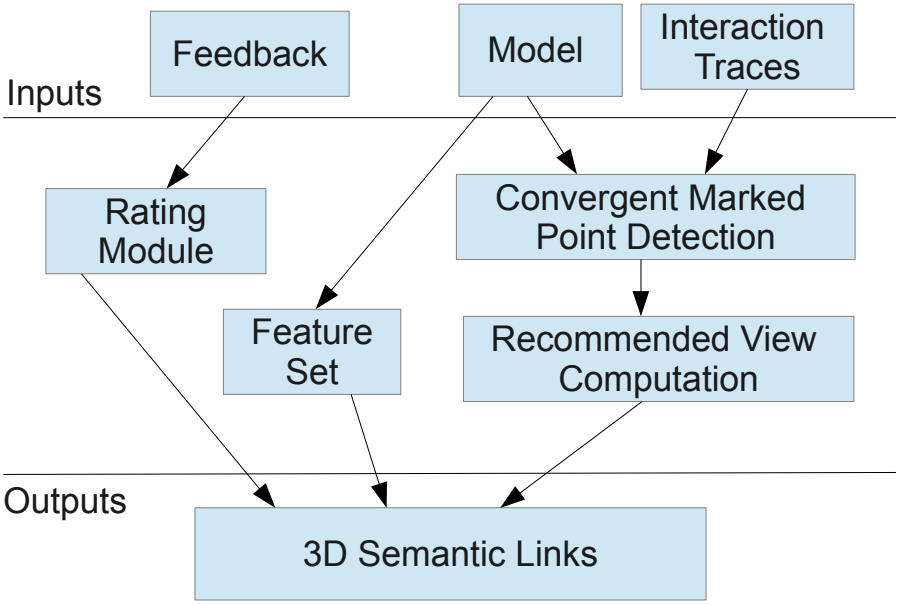

Figure 4.9: Logic View of Our System.

From technical point of view, our system acts as an extension for the existing 3D web application. Figure 4.9 presents the logic view of our system. As can be seen from the figure, our system receives 3D model and interaction traces as inputs to

detect convergent marked point and compute recommended views. Having computed recommendations, our system queries feature set from the 3D model and produces 3D semantic links as outputs. As can be seen from the left part of the figure, we develop a rating model to integrate user feedback into the computation of 3D semantic links. Figure 4.10 shows the architecture design to implement our system. On the client side, we develop a *Listener* component which is embedded into the client side renderer and is used to capture user interactions and feedbacks. We then send collected data back to the server and store them into the databases User Traces and Feedbacks.

On the server side, the *Communication Module* acts as a communicator among the existing 3D web application, client requests and our system core (Location Detector, Semantic Link Generator, and Rating Module). With respect to the 3D web application and users, our system acts as a web service, in which each time the user explicitly selects a semantic association, the *Communication Module* forwards user request to the system core to obtain the corresponding 3D visualization, presented as recommended 3D view.

The main capabilities and functionalities of our system are developed in the *system core*, which contains the *Location Detector*, the *Semantic Link Generator*, and the *Rating Module*. The *Location Detector* queries input data from User Traces database, then analyzes to detect the location of 3D visualizations associated with textual contents. The *Semantic Link Generator* uses detected locations of 3D visualizations to compute the recommended 3D views such that the convergent marked point is at the center of the viewport. Having the recommended views, the *Semantic Link Generator* queries the associated textual names from Feature Set database, then generates the semantic associations. The *Rating Module* queries the data from Feedbacks database to process the feedbacks and opinions on the relevance of semantic associations from the crowd, then sends these feedbacks to the *Semantic Link Generator*. The *Semantic Link Generator* integrates the feedbacks as a part of recommendations to be presented to users.

We have presented the logic view of our system according to the proposed approach. We now show in more details how we implement the system.

In order to detect the location of 3D visualization associated with a textual content, we compute the convergent marked point as the point from the collected data which is closest to the median of marked-points (each coordinates considered separately). We chose Median since it is the robust position estimator of marked-points from the 'crowd', outlier corresponding to unconcentrated, uncompetant or malicious users are not disturbing the convergent marked-point. Having the detected location of 3D visualization, we calculate camera position and orientation so that

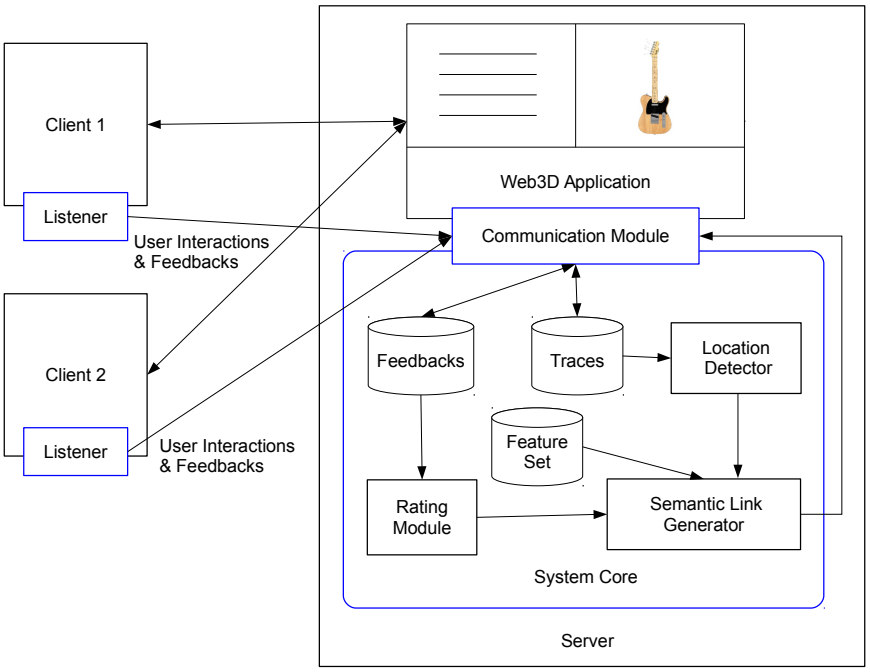

Figure 4.10: System Architecture.

its look-at point is convergent marked-point. Using recommended 3D views and associated textual names, we then generate the semantic links. In our method, the quality of semantic links is correlated with the dispersion of marked-points.

Similar to chapter 3, we also use plugin-free rendering solution (VRML/X3D, X3DOM) to implement our approach. Concerning the dabatase system, there exist three main models to store data. The first one is Hierarchical Database where parent nodes have more intrinsic importance than its child nodes. The second one is Graph Database where objects are arbitrarily related with no intrinsic importance. The third one is Relational Database where data tables are related by primary key. In our work, we adopt Relational Database structure since it is convenient to store and manipulate collected data.

We have presented our proposed approach and introduced a possible implementation. We now describe an experimental setup of the user study and provide an analysis of the results in order to evaluate our method.

4.3.3 Set-up of the Experiments

Participants. Participants were volunteers and the experiments took place in the presence of an organizer. We avoided remote online experiments to insure the accuracy of recorded time. A total of 47 male and 35 female participants aged from 19 to 40 (mean 27), mostly from the university community took part in the experiment. None of the users participated in more than one part of the test.

Figure 4.11: Models of the User Study.

Models. Since we target the e-commerce use case, we chose models of everyday life in our user study. The models are of different complexity, aesthetics, and require various knowledge from daily life to technical aspect or specific domain. More specifically, we use 6 models, including two cameras, two guitars and two coffee machines (see figure 4.11). Products have been chosen for having both technical features and aesthetical relevance. For each model a list of four features defined by a textual description is given.

Database. While users were completing their tasks of locating a textual feature on 3D models, we collected their traces and stored them into our databases. The data collected consisted in:

- The amount of time it took the user to find each feature's visualization in 3D.
- The world coordinates of each 3D *marked-point* on the surface of 3D object, representing the feature's recognized position. World coordinates of normal vector and of camera position at each marked point are also logged.
- Time and events (zoom/pan/rotate/double click, etc) created by users.
- "I don't know" events if users cannot locate the feature.
- Level of user knowledge about each product.

We used these collected data to evaluate our crowdsourcing approach, that is to quantify the increase in performance of users given produced links both in terms of efficiency and correctness.

4.3.4 Protocol of the User Study

Since our work targets web applications having online 3D content to browse like e-commerce or virtual museums, we conducted the experiments on the 3D product pages of *e-commerce* sites. Initially, we considered product pages that simply contain basic texts and a browsable 3D model of the product (see figure 4.12). The goal of the user study is to use crowdsourcing to generate semantic associations added to these product pages. The enhanced product pages are then suggested to subsequent users for quantifying the helpfulness of semantic links.

The user study consists in 4 different parts, each of which evaluating a different aspect of our work:

- Part 1: We gave users outsourcing tasks in order to generate semantic links;
- Part 2: Outsourcing tasks with recommendations deduced from part 1 were provided to users to evaluate quality of crowdsourced semantic links;
- Part 3: Outsourcing tasks with recommendations and rating scores from part 2 were given to users to evaluate the influence of crowd opinions on recommendations;
- Part 4: Outsourcing tasks were given to users to assess user preferences to the enriched interface of semantic links compared to the initial one (3D product along with its textual description).

A detail of the four conducted parts of the experiments is presented as follows:

Part 1: Association (see figure 4.12): For each model, the user selects features one by one and tries to locate the feature on the 3D object. If he/she is able to find it, he/she can double-click on it and a red dot marks the location on the object. We call this red dot the *marked-point*. If the user is not able to locate a feature, he/she can click the "I don't know" button and go on to the next feature. We then ask users to estimate their degree of familiarity with the product, ranging from 1 (novice user) to 5 (expert user).

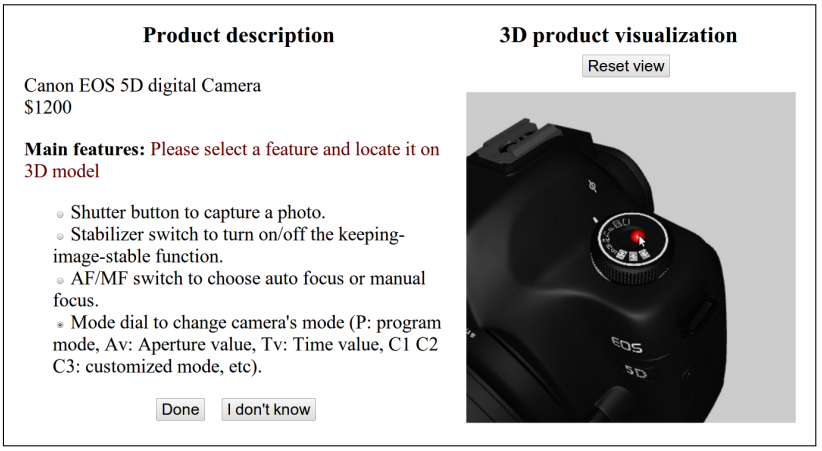

Figure 4.12: The interface to collect user traces: users are asked to select a textual description, then locate it on the 3D model (part 1).

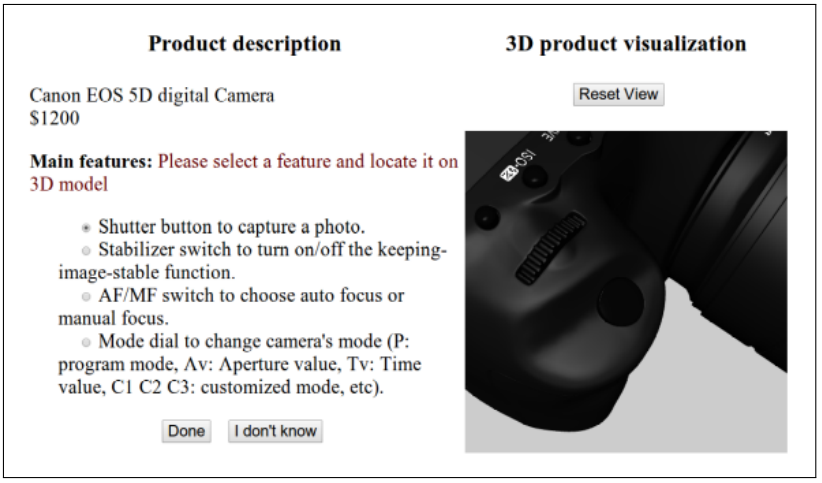

Figure 4.13: Recommendation is given automatically to the user when he/she selects a textual description (part 2).

Part 2: Evaluation (see figure 4.13). In this step, users have to do the same operations as in Part 1, except that each time users choose a feature, a recommended view is automatically displayed to suggest the corresponding 3D visualization of the feature. Then, for each feature, we ask the user if the recommended view was helpful or not (figure 4.14).

Part 3: Helpfulness Evaluation (see figure 4.15). Users are in the same conditions as in Part 2 except that they are given the results of the helpfulness evaluation of the recommended views from Part 2. We still ask them to do the same operations as in Part 1.

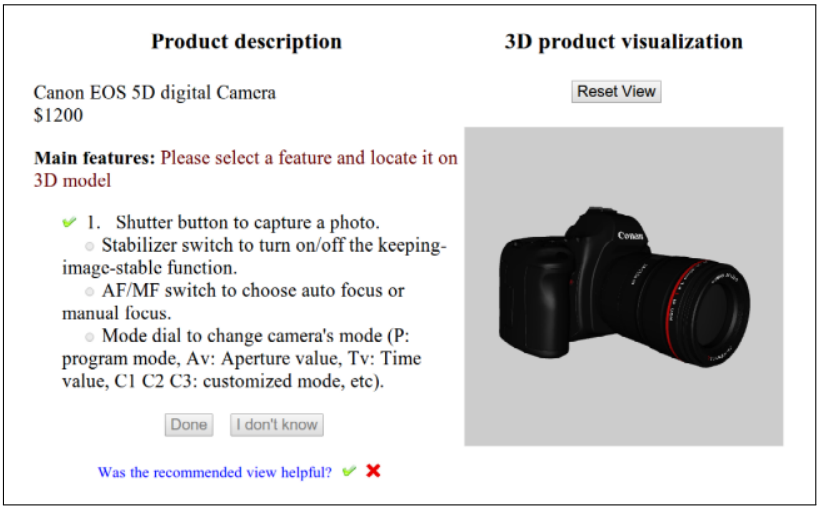

Figure 4.14: Users are asked to evaluate the helpfulness of Recommended View at the end of each task (part 2).

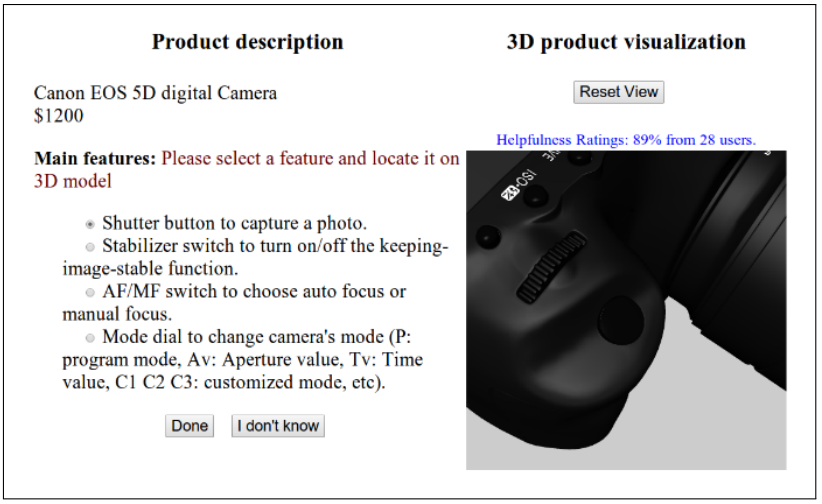

Figure 4.15: Recommended View is integrated with Helpfulness Score when he/she selects a textual description (part 3).

Part 4: Novel Interface Evaluation (see figure 4.16). In this step, users are asked to interact with two different interfaces: one is just a 3D product along with its textual description, and the other one, though very similar, associates a semantic link on each textual feature of the 3D model. We then asked users which of the interface they would prefer in an e-commerce scenario.

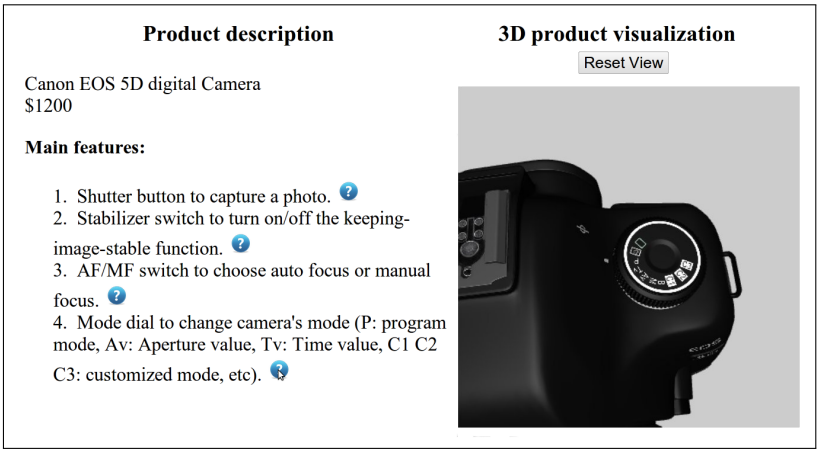

Figure 4.16: The final recommended interface with semantic links shown as blue bullets with question mark is proposed to users (part 4).

4.3.5 Experimental Results

In this section we analyse the data collected from the four parts of the experiment that are described in the previous section.

Quality of answers and recommendations

Figure 4.17 shows examples of the recommendations we get from our system. Those recommendations are generated by analyzing user traces from Part 1-3 of the experiment. Each of these views is associated to one feature from the textual description. The views should show the filter-holder of the coffee machine, the jack socket of the electric guitar and the stabilizer switch of the photo camera. The coffee machine is a simple and popular object and almost everybody knows what a filter-holder is, which explains why the recommendation is precise in all parts of the experiment. The guitar though is a more specific object, and the user needs to know some technical details about electric guitars to be able to locate the jack socket. For such objects, we observe that the recommendation generated from crowdsourced associations is rather accurate and opinions on helpfulness of recommendation helped to improve the accuracy of recommendation. Finally, the stabilizer switch of the camera is difficult to find and some users mis-identified it. In this case, we see that opinions on helpfulness of recommendation could help to correct the position of the stabilizer switch, which results in a different recommendation compared to the one generated from crowdsourced associations. These three features are examples out of the 24 features (6 3D objects with 4 features each) that we studied in our experiments. By analyzing the variance among answers, and the

Figure 4.17: The original views (left column – part1). Recommendations generated from crowdsourced associations (middle column – part2). Recommendations improved by providing opinion on the helpfulness (right column – part3).

difference between answers depending on the user (self estimated) expertise, we can group the recommendations obtained into three classes as follows:

- **Easy features** have lots of good answers, both by experts and others users; have small variance; account for 18 out of 24 features with good recommendations.
- **Technical features** have lots of "I don't know" answers, mostly by non-expert users; have large variance; account for 4 out of 24 features with rather accurate recommendations.
- **Hard features** have lots of wrong answers from non-expert users; have large variance; account for 2 out of 24 features with inaccurate recommendations.

Efficiency to execute the task

Figure 4.18 presents the distribution of the average time taken by users to locate the features on the 3D models. On this figure, we focus on the 3D models that appear on figure 4.17: a coffee machine, an electric guitar and a camera. We plot the average time taken by user to execute the provided task (correctly or not). For

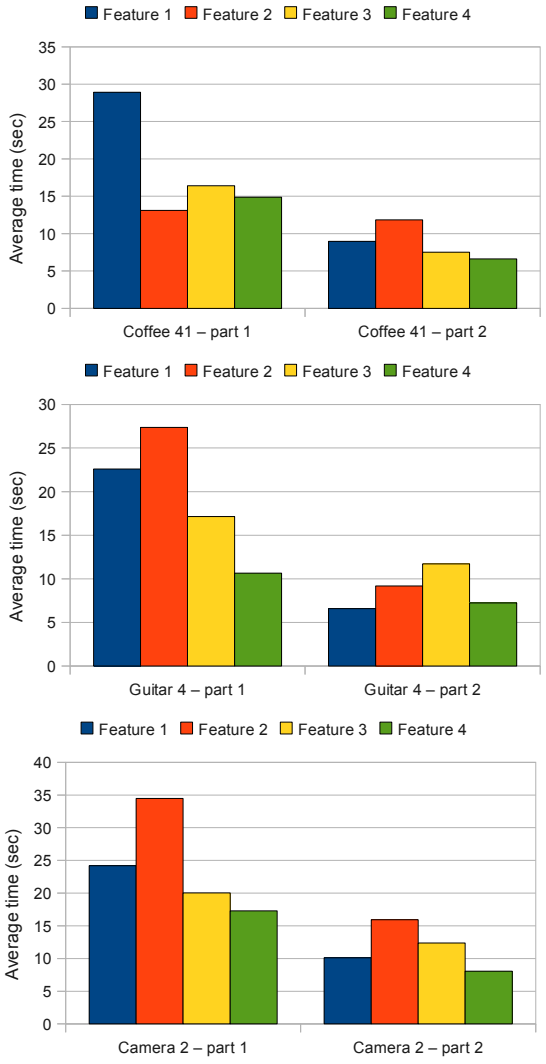

Figure 4.18: Average time to locate the features in 3D.

	Right	Do not know	Wrong
Part 1	75%	12%	13%
Part 2	80%	12%	8%

Table 4.1: Percentage of right answers, "I don't know" (DNK) answers, and wrong answers to the tasks for Parts 1 and 2 of the experiments.

each product (e.g. the camera) there are two sets of results corresponding to the average time of users from Part 1 and Part 2 of the experiment. As can be seen from this graph that the users from Part 2, who were given recommendations based on traces of users from Part 1, are quicker to perform the task than users from Part 1. Table 4.1 shows the percentage of right and wrong answers, as well as "I don't know" answers for both Parts 1 and 2 of the user study. We can see when looking at this table, that users not only perform the task more quickly, but also that they

are more efficient. The percentage of right answers is higher in Part 2 than in Part 1, reaching now the value of 80%.

	Easy	Technical	Hard
Helpfulness	85%	54%	25%

Table 4.2: Percentage of users from Part 2 thinking the recommendation was helpful, for three features each characteristic of one class.

Influence of the comments from previous users

Having presented the three types of features (easy, technical, and hard) and shown that recommendations help users being more efficient in completing the task, we now want to establish the interest of asking users to evaluate a recommendation's helpfulness.

Table 4.2 presents the percentage of users from Part 2 who found the recommendations (generating from traces of users from Part 1) helpful to complete the task. We show these percentages for three particular features, each represents for one of the three classes we defined earlier : easy, technical and hard. The easy feature is the "filter holder" from one of the coffee machines, the technical feature is a "Jack socket" from one of the electric guitars and the hard feature is the "Stabilizer switch" from one of the cameras. We could sum up the results from Table 4.2 by saying that users from Part 2 find the recommendation for the easy feature very helpful, the recommendation for the technical feature moderately helpful and the recommendation for the hard feature not helpful. It is interesting to observe the influence of such information (how helpful users find on the recommendations) on others users, and that is the purpose of Part 3 of the user study.

Feature class	Part 1		Part 2		Part 3	
	Wrong	DNK	Wrong	DNK	Wrong	DNK
Easy	10%	0%	0%	0%	0%	0%
Technical	15%	40%	0%	50%	0%	35%
Hard	35%	35%	43%	35%	35%	50%

Table 4.3: Percentage of wrong answers and "DNK" answers on three representative features for Part 1, Part 2 and Part 3 of the user study.

Table 4.3 presents the percentage of wrong answers and "I don't know" answers (the percentage of right answers can be easily deduced from these numbers) on the three features previously mentioned, for the three first parts of the user study.

For the easy feature, 90% of the users from Part 1 accurately locate the feature on the 3D model which results on a good recommendation. Therefore when presented

with the recommendation, all users answer correctly to the task of locating the feature.

For the technical feature, users from Part 1 have more trouble identifying the feature on the 3D model. 40% of users acknowledge they do not know the answer, and 15% of the users locate the feature at the wrong place. This means there are still 45% of users that identify the feature's location and it enables our system to produce a meaningful recommendation. The recommendation, when provided to users from Part 2, indeed prevents them from locating the feature at the wrong position, but increases the number of users acknowledging they do not know where the feature stands on the 3D model. This result shows that users somehow trust the recommendations enough to prevent them from answering badly, but do not trust it completely to prevent them from answering that they do not know where the feature is located. Users from Part 3 however, because they also have information about the helpfulness of the recommendation, are more encline to trust the recommendations and we observe that 65% of them get the correct localization for the feature which is a great improvement compared to users from Part 1 (45% of good answers) and Part 2 (50% of right answers).

For the hard feature, 35% of users do not know where to locate the feature and 35% of the users locate it wrongly. The generated recommendation is therefore really bad, and users from Part 2 perform even worse. 43% of them position the feature at the wrong place. The helpfulness measure being very low (25%, see Table 4.2). 50% of users from Part 3 prefer to answer they do not know the place of the feature.

Interface

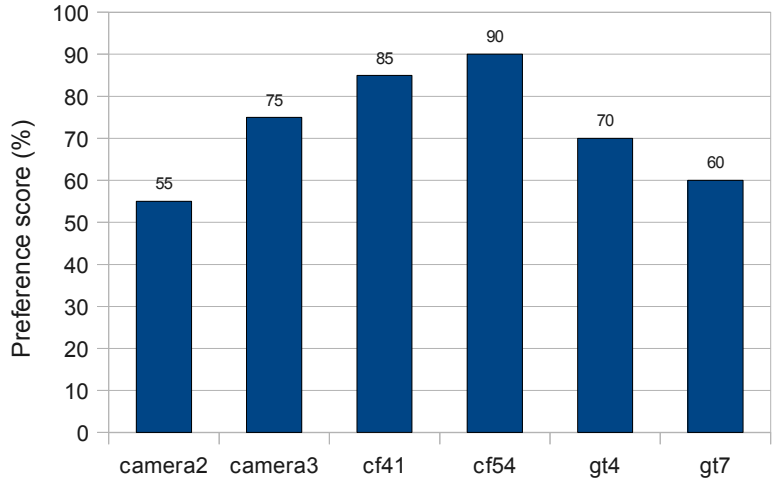

Figure 4.19: Evaluation of user preference to the recommended interface.

Finally we asked the last group of users (from Part 4) to compare two interfaces and decide which one they prefer. We presented them, for each 3D model, with two different interfaces. Both interfaces include a 3D visualization of the product and a textual description of the product. But one of the interfaces supports semantic links. In more details, for each feature in the textual description, a button is added and can be clicked to automatically change the view of the product, displaying the recommendations generated thanks to the traces of users from Part 1. Figure 4.16 shows an example of this recommended interface, and on this example the user has just clicked on the 4th feature's button. Figure 4.19 presents the results of this study. Users tend to prefer the recommended interface for all the products, even for "camera2" in which 2 recommendations are in low quality.

4.3.6 Interpretation

The conducted experiment and the user study have shown two particular points: first, we show that the semantic associations can be derived from crowdsourcing. Second, the proposed semantic association has proved useful in two ways: qualitatively, users have appreciated it, quantitatively, it has improved their performances.

We show that the lack of expertise for the crowdsource may be overcome: we have implicit and explicit clues to assess the quality of a recommendation. Implicit clues come from the distribution of answers from multiple users. Convergent answers, that is, a set of samples well approximated by their average/mean value, correspond to features easy to identify whereas features difficult to detect lead to few or sparsly distributed answers. Explicit feedback on how helpful a recommendation was, has also been collected and given to following users to help them assess if a recommendation could be trusted. By getting opinion of users on the derived association, we manage to improve the quality of their answers.

The proposed association between textual content and 3D interactive object has been appreciated by 85% of users (that is, they said it was helpful). Of course, one can argue that feedback from previous users when performing the same task is expected to be helpful. However, we may wonder if changing poses (or viewpoint) by clicking on the textual description rather than using only the classical mouse interaction is disturbing. It clearly appears that inducing a motion on the 3D object by clicking on the text is not disturbing since the text and viewpoint are semantically linked. Moreover, in the last experiment, users were simply asked to compare a 3D content and independant text and most preferred our proposed enhanced version. We thus showed that the proposed semantic association is useful to users.

Quantitatively, the proposed semantic association has also improved performances of users. In terms of efficiency, the time required to achieve a task is much shorter when using recommended views. In terms of correctness, accuracy is improved: more right answers are given in part 2 and 3 than in part 1. This is true in particular for easy features, which most of them are, but not necessarily for technical or hard features. However, for technical or hard features, we have shown that by including user feedback on the helpfulness of the recommendation, we were able to decrease the number of wrong answers compared to the original test (no recommendations).

Finally, we have seen that the use of crowdsourcing in this context has been valuable. First, the identification of easy, technical and hard features has been possible through the analysis of user traces (in particular, the dispersion of the answers, and the difference between expert and non expert users). This grouping of features has been important since the results were very dependent on the nature of the feature. Second, opinion on the recommended views has proved a very valuable information: for hard features, giving a feedback on the helpfulness of the recommended view did decrease the number of wrong answers (more people did 'admit' they did not know). In some sense, it tells us that if the crowdsourced information is not perfect, and certainly worst than an expert advice, a further iteration of crowdsourcing can derive a valuable confidence measure on this information.

One concern in our approach is that if it is possible to use our proposed pipeline in a large scale environment with uncontrolled crowds. From our experiments, we show some clues to answer this question. First, we have opportunities to get customers to work for us in e-commerce domain where many e-shopping sites with different types of crowds can be used. Second, in part 3 of our experiments, we show that the feedback from the previous users in the crowd did improve the quality of recommendations. This improvement is adaptive to the expertise of the crowd and converges to the recommendation equally to the one created by the experts. Having these clues, a setup on real e-commerce sites would nicely evaluate the use of our approach in such large scale environments.

4.4 Summary and Perspectives

In this chapter, we proposed a new approach using *crowdsourcing* to build semantic association between text and 3D graphics. The links allow users to readily locate 3D visualization associated with a textual content. With an experimental user study, we evaluated that the proposed semantic association reduced 3D browsing time and was useful to subsequent users, both for correctness and efficiency. Moreover, the results from the user study showed that the proposed semantic association is appreciated by users, that is, a majority of users assess that recommendations were helpful for them, and browsing 3D objects using both mouse interactions and the proposed links is preferred compared to having only mouse interactions.

This work can be improved in several directions. The first one is to deepen analysis on the crowd's inputs, such as studying when and in what ways the workers may give up a question or submit wrong answers. The second one is to study whether and how the initial viewpoints may affect the results. Those methodologies aim to dig more efficiently and effectively information from the crowd inputs so that to produce more meaningful and accurate recommendations.

Similarly to the previous chapter, another improvement of this work is to setup experiments in order to verify the use of our system on a less biased and larger crowd as well as test the robustness of our method. A similar approach in chapter 3 which employs the correctness of recommendations as well as convergence time of the crowd could be added to such experiments. Another direction of our work is to consider automatic guiding of users when performing the annotation task of linking text and 3D visualization. For example, an automatic guiding tool could guide users to a set of personalized 3D interactions if it triggers that users are getting lost to browse the 3D model. Using this guiding, we could reduce outliers in the 3D interactions of the crowd to generate semantic links.

Chapter 5

Conclusion and Perspectives

Contents

5.1 Conclusion

The Web evolved from a read-only text-based system to the currently rich and interactive Web that supports 2D graphics, audio, and videos. This evolution continues to the ultimate achievement of visual media – 3D graphics, in which information is expressed through 3D visualization in order to reproduce the real world objects or to represent the imaginary worlds. The use of online 3D graphics on the Web does not mean to substitute traditional web content of texts, images and videos, but rather acts as a complement for it. The Web is now a platform where hypertext, hypermedia, and 3D graphics are simultaneously available to users. This use of online 3D graphics, however, poses two main issues:

First, since 3D interactions are cumbersome with numerous degrees of freedom, 3D browsing can be inefficient. Specifically, users are required to control too many parameters of viewpoint movement to perform online 3D interactions that they can easily get frustrated. For example, new-to-3D users, who can be domain experts, but have marginal knowledge about interaction techniques, can find it difficult to interact with 3D visualizations. It is common for them to face the problem of "where my

object is in the scene" or spend too much time on getting familiar with 3D interaction techniques. Inadequate support to user interactions with 3D graphics in these cases can result in users leaving the site.

Second, 3D graphics contain purely rich visual information of the concepts. On the other hand, traditional websites mainly contain descriptive information (text) with hyperlinks as navigation means. The problem is that using these two very different mediums on the website to express the same information can generate complications for users: they need to deal with text browsing to look for general information and text searching for more detail information. On the other hand, they also need to handle how to interact with 3D visualizations to gain better understanding of the data (e.g. navigating in a 3D space, or inspecting and examining virtual 3D objects from different points of view). This separation of interactions between the two main modalities (text and 3D graphics) requires users too much effort to browse the site.

This thesis introduced novel methods using *crowdsourcing* to tackle these issues of using online 3D graphics. Our proposed techniques reduce 3D browsing and simplify 3D interactions. We illustrated this research work as in the following.

We presented the foundations of this thesis in chapter 2. We introduced the concept of hypertext as the key web navigation paradigm and provided the context we chose to study in this work – 3D interactions as new web navigation model and the use of crowdsourcing to develop 3D websites. We then presented the main 3D techniques and technologies we used for our research work and perspective uses of online 3D graphics that our work targeted.

Chapter 3 described the first contribution of this work that tackled the first mentioned problem of using online 3D graphics. We proposed a new paradigm to ease online 3D interactions with virtual 3D models based on *crowdsourcing*. We detailed our proposal in a pipeline that consists of analyzing 3D user interactions to identify Regions of Interest (ROIs), and generating recommendations to subsequent users. We proposed an implementation of our pipeline using plugin-free 3D web techniques and technologies (HTML5, WebGL, X3DOM) and introduced a simplified 3D user interface using our proposed recommendations. With an experimental user study, we evaluated that the recommendations both reduce 3D browsing time and simplify 3D interactions. Morever, the results from the user study also showed that 3D viewing behaviors of the crowd can converge to an interesting 3D content. Finally, we outlined several future directions to improve this work.

Chapter 4 described the second contribution of this work that tackled the second mentioned problem of using online 3D graphics. We proposed a new approach using

crowdsourcing to build semantic association between text and 3D for enhancing 3D browsing – a semantic association links a meaningful part of the 3D model with its textual name. These links allow users to readily locate the 3D visualization associated with a textual content. We defined the task of associating online textual name and 3D visualization as *online 3D annotation task* and proposed to use crowdsourcingn to perform this 3D annotation task. We chose crowdsourcing since it is an effective method to perform online annotation tasks in multimedia applications on the Web (e.g. online image and video applications). With an experimental user study, we evaluated that the proposed semantic associations reduce 3D browsing time and are useful to subsequent users, both for correctness and efficiency. Moreover, the results from the user study showed that the proposed semantic association is appreciated by users, that is, a majority of users assess that recommendations were helpful for them, and browsing 3D objects using both mouse interactions and the proposed links is preferred compared to having only mouse interactions. Finally, we outlined several future directions to improve this work.

5.2 Perspectives

As mentioned earlier, the use of online 3D graphics provides a new web navigation model – 3D interactions, and crowdsourcing emerges thanks to the popularity of the Participatory Web (Web 2.0). In this work, we have proposed a new direction where crowdsourcing is used to enhance 3D browsing, that is, to ease 3D browsing time and simplify 3D interactions. Although we have shown in the two contributions that the collective participation of web users in the crowdsourced tasks does produce valuable helps to enhance the use of online 3D graphics, we believe that we have merely scratched the surface in this area of research, and that many interesting and exciting directions still lie ahead. We have identified several key areas, but not limited, for future research of this work, as outlined in the followings.

5.2.1 Short-term Perspectives

The proposed methods can be improved in several directions as summarized in the following:

Deepening Crowdsourcing Usage

One future direction of this work is to deepen analysis on the crowd's inputs so as to dig more efficiently and effectively information from the crowd. For example,

one scenario is to study when and in what ways the workers may give up a question or submit wrong answers; or whether and how the initial setup of 3D user interface may affect the results.

Another avenue of this work is to widen crowdsourced 3D tasks to be provided to the web users. For example, in an e-commerce setup, users can be asked to rank the features by order of importance, regarding an eventual buying of the product. This information can then be used to create an automatic description of a 3D product that would emphasize on the more popular product features.

Another future direction is to deepen analysis on the crowdsourced 3D viewing behaviors. For example, by learning from their web visits, 3D interaction patterns of the crowd can be deduced – a 3D interaction pattern is a sequence of 3D tasks which the crowd performs on the behalf of a group of users who share the common needs or interests. These deduced patterns can then be used to anticipate users' needs and preferences in upcoming navigations and interactions. A finite-state machine can be a technical solution used to model 3D interaction patterns following this approach.

Development in Real Life Scenarios

Currently, we validated our methods via user studies with participants mostly from the university community. A setup on real websites with real web users could greatly improve the results of our proposed methods. For example, in one part of our experiments, we have shown that opinions on helpfulness of recommendations did help to improve the quality of recommendations and increase the efficiency for users to complete the 3D tasks. Thus, the set-up of this scenario on real e-commerce sites can produce valuable information for subsequent customers who would prefer to get references from previous customers.

Another avenue to use and validate the proposed methods is to create a real setup on crowdsourcing marketplaces, such as Microworkers [51], thanks to the use of promotion credits or discount codes. For example, the proposed methods can be established on the Microworkers platform with an e-commerce company as the Employer, who is interested in a task solved by humans, and the Internet users as the Workers, who are willing to perform the requested tasks for a payment. The paid tasks assigned to the Workers are difficult tasks for automatic machines but simple for humans. For example, by using an automatic system, we can calculate a set of best views – each contains an informative 3D region of the 3D objects, then propose these computed views to web users as a set of thumbnails. The paid task provided to web users is to label the best textual descriptions for each product feature presented in the computed thumbnails. This annotation task can be easy for humans but difficult for automatic machines as it requires a deep understanding of

the product features as well as additional knowledge about technical characteristis and aesthetical design of the product features. Using this information, the proposed system analyzes the crowdsourced tasks selected by the users and deduces the recommended thumbnails containing the best descriptions of the product features. The Employer (the e-commerce company) can then use such crowdsourced descriptions and automatic recommended thumbnails to advertise the 3D products to subsequent customers. From this example, we open a new direction that crowdsourcing method can be combined with automatic approaches to enhance the use of online 3D graphics on the Web.

5.2.2 Long-term Perspectives

This work can be extended in several directions as summarized in the following:

Smart Graphics Framework

As mentioned earlier, the use of online 3D graphics offers users various online 3D tasks to perform. Jankowski, 2011 [57] grouped these tasks into a so-called "taskonomy of 3D web use" – a taxonomy of web tasks and 3D tasks, which identifies the main browsing tasks the users may have when browsing the 3D web. However, performing these 3D tasks can be difficult for users as 3D interactions are cumbersome and 3D rendering and transmitting require powerful computations as well as high-quality network capabilities. To overcome these limitations, an interesting direction to enhance the use of online 3D graphics is to develop a smart graphics framework which enables the so-called *smart graphics* to interact with user environment in order to tailor user interaction context. Specifically, smart graphics are graphical components of the website which know how users can interact with them and how to aid users in their interactions. For example, when the user looks at a virtual camera inside a training class on his/her personal computer (e.g. the laptop) or inside a shopping activity on his/her smart phone, the smart graphics do not offer the same features and functionalities to the user. In the first case, the smart graphics would provide users basic features and tutorials so that the user could learn how to manipulate a camera (e.g. he/she could learn how to press a virtual shutter button to take a photo). On the other hand, in the second case, the smart graphics would provide users the descriptions and visualizations of the best product features as well as additional information such as the price and promotion policies (e.g. free shipping). In this case, the user could be allowed to zoom in on interesting features of the products to inspect them from different points of view.

A possible method to develop smart graphics framework is to define graphical components using the concept of smart components and smart objects. Specifically, each graphical component is considered as a smart graphics component that can adapt its behaviors to individual users based on the knowledge of user environment model (e.g. action model, domain model and user model). For example, we can determine a smart graphics component as an agent related to its virtual representation – called an avatar. Smart graphics avatar is used to communicate with user environment and smart graphics agent encompasses core functionalities of the smart graphics component. Each time, the user uses the smart graphics system, smart graphics avatar perceives the values of user environment model (e.g. terminals, network capabilities, user interests and preferences, etc) and forwards them to the smart graphics agent. Based on these forwarded knowledge, the smart graphics agent uses the adaptation algorithms to calculate the adapted visual displays and behaviors of the smart graphics and create the so-called adapted avatar which is then offered to the user to support their interaction context. From this example, we open a direction where smart graphics framework allows to provide intelligent visual displays and behaviors of graphical components based on the knowledge of user environment model regarding all types of graphics on the Web such as images, videos or 3D graphics. The construction of such smart graphics framework seems to suit well with the current Web where images, videos and 3D graphics are simultaneously available to users. Having the development of this smart graphics framework, we can then setup more useful set-ups of smart graphics on 3D e-commerce sites as presented in the next paragraph.

More Set-ups on 3D E-commerce

3D e-commerce is a very interesting perspective domain to use online 3D graphics. In 3D e-commerce, users are allowed to examine online 3D products from every angle: they are offerred the ability to perceive the most realistic 3D view of the products they are looking for; or even are allowed to experience the virtual shopping actions close to their real life shopping behaviors. For example, in La Redoute – the Virtual Shopping Site [12], the user can choose an avatar and create his/her own look for the so-called virtual try-on. Specifically, the system asks the user to mention height, weight, and answer some questions about the preferred features of the product (e.g. color), then creates a virtual 3D model that allows him/her to try-on the virtual clothes. Although this process provides the user the similar real worlds shopping actions, it is simply a reproduction of the virtual features indicated by the user on his/her chosen avatar. This type of system has not yet taken into account the possibilies to consult users the interesting virtual features of the products so that they can try on in real time. In contrast, by using the mentioned smart graphics framework, users can be offered the best look of the product they are finding on an e-commerce site. For example, they can be suggested and tried on in real time

the style of the clothes which are most suited with their age, gender, body size or interests and preferences.

Studying More Interaction Techniques

In this context, the interaction techniques are limited to mouse and keyboard. Another possible avenue is to extend the use of crowdsourcing for various 3D interaction techniques. For example, our system may crowdsource 3D user interactions used on the different input devices such as touch inputs, 3D joysticks or head-mounted displays. Tactile input devices are used in many types of smart phones nowadays. A set-up to crowdsource context-aware 3D viewing interactions on tactile input devices may be interesting. For instance, by analyzing crowdsourced 3D interactions on a specific tactile input device, our system can determine interesting 3D content for different types of display modes – each display modes corresponds to a mode of user interface displayed to users (e.g. portrait mode or landscape mode in smart phones or tablets). Then, suitable adaptive representation of 3D objects with highlighted ROIs can be inferrred and displayed to users according to the switch task of display modes they have created. In this way, we open a new direction that 3D viewing interactions of the crowds can be used to modify and improve the generations of 3D content to be used on the web applications.

We have outlined several key directions to improve and extend this work. We hope that the contributions and the open issues would inspire the researchers in these domains for further improvements.

Bibliography

[1] 3D-COFORM Project. http://www.3dcoform.eu/x3dom. accessed June, 2013.

[2] Adobe Flash 10. http://www.adobe.com/products/flashplayer/. accessed January, 2014.

[3] Amazon: E-commerce Site. http://www.amazon.com/. accessed July, 2014.

[4] Amazon Mechanical Turk: Crowdsourcing Platform. https://www.mturk.com/mturk/welcome. accessed July, 2014.

[5] Ardzan: 3D Virtual Internet. http://www.ardzan.com/. accessed January, 2014.

[6] CrowdFlower: A Crowdsourcing Platform. http://crowdflower.com/. accessed January, 2014.

[7] Fiat: E-commerce Site. http://www.fiat.ie. accessed June, 2013.

[8] FittingBox Company. http://www.fittingbox.com/. accessed January, 2014.

[9] Flickr: Photo Sharing Site. http://www.flickr.com/. accessed January, 2014.

[10] Flux Player. http://www.web3d.org/products/detail/flux-player. accessed June, 2013.

[11] HTML5 Specification. http://www.w3.org/TR/html5. accessed June, 2013.

[12] La Redoute: 3D Shopping Site. http://www.laredoute.fr/. accessed January, 2014.

[13] LabelMe: Crowdsourcing Platform. http://labelme.csail.mit.edu/Release3.0/. accessed July, 2014.

[14] Microsoft Silverlight. http://www.microsoft.com/silverlight/. accessed July, 2014.

[15] Octaga Player. http://www.web3d.org/products/detail/octaga. accessed June, 2013.

[16] Second Life: Social Networking Site. http://secondlife.com. accessed June, 2013.

[17] Sun Java3D: Media Framework. https://java3d.java.net/. accessed January, 2014.

[18] Thai Ancient City. http://www.ancientcity.com/. accessed January, 2014.

[19] VisualSFM: A visual structure from motion system. http://homes.cs.washington.edu/~ccwu/vsfm/. accessed January, 2014.

[20] VRML Specification. http://www.web3d.org/x3d/specifications/vrml. accessed June, 2013.

[21] Web Server Survey. http://news.netcraft.com. accessed December, 2013.

[22] WebGL Technology. http://www.khronos.org/webgl. accessed June, 2013.

[23] X3D Specification. http://web3d.org/x3d/specifications. accessed June, 2013.

[24] X3DOM Framework. http://www.x3dom.org. accessed June, 2013.

[25] Arondi, S., Baroni, P., Fogli, D., and Mussio, P. (2002). Supporting co-evolution of users and systems by the recognition of Interaction Patterns. In *Proc. of the Working Conference on Advanced Visual Interfaces*, pages 177–186. ACM.

[26] Baccot, B. (2012). *Soprfiling, an adaptation platform.* PhD thesis, University of Toulouse, France.

[27] Baccot, B., Choudary, O., Grigoras, R., and Charvillat, V. (2009). On the Impact of Sequence and Time in Rich Media Advertising. In *Proc. of the 17th ACM International Conference on Multimedia*, MM '09, pages 849–852, New York, NY, USA. ACM.

[28] Behr, J., Eschler, P., Jung, Y., and Zöllner, M. (2009). X3DOM: a DOM-based HTML5/X3D integration model. In *Proc. of the 14th International Conference on 3D Web Technology*, pages 127–135. ACM.

[29] Behr, J., Jung, Y., Keil, J., Drevensek, T., Zoellner, M., Eschler, P., and Fellner, D. (2010). A scalable architecture for the HTML5/X3D integration model X3DOM. In *Proc. of the 15th International Conference on Web 3D Technology*, pages 185–194. ACM.

[30] Berners-Lee, T. (1999). Weaving the Web: The Past, Present and Future of the World Wide Web by Its Inventor (with M. Fischetti).

[31] Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H. F., and Secret, A. (1994). The world-wide web. *Communications of the ACM*, 37(8):76–82.

[32] Bush, V. (1945). As we may think.

[33] Callison-Burch, C. (2009). Fast, cheap, and creative: evaluating translation quality using Amazon's Mechanical Turk. In *Proc. of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 286–295. Association for Computational Linguistics.

[34] Carlier, A., Charvillat, V., Ooi, W. T., Grigoras, R., and Morin, G. (2010). Crowdsourced automatic zoom and scroll for video retargeting. In *Proc. of the international conference on Multimedia*, pages 201–210. ACM.

[35] Celentano, A. and Pittarello, F. (2004). Observing and adapting user behavior in navigational 3D interfaces. In *Proc. of the working conference on Advanced visual interfaces*, pages 275–282. ACM.

[36] Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(8):790–799.

[37] Chittaro, L. and Ranon, R. (2002). New directions for the design of virtual reality interfaces to e-commerce sites. In *Proc. of the Working Conference on Advanced Visual Interfaces*, pages 308–315. ACM.

[38] Chittaro, L. and Ranon, R. (2007a). Adaptive 3d web sites. In *The adaptive web*, pages 433–462. Springer.

[39] Chittaro, L. and Ranon, R. (2007b). Web3D technologies in learning, education and training: Motivations, issues, opportunities. *Computers & Education*, 49(1):3–18.

[40] Chum, O., Philbin, J., Sivic, J., Isard, M., and Zisserman, A. (2007). Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

[41] Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619.

[42] Conklin, J. (1987). Hypertext: An Introduction and Survey.

[43] Cormode, G. and Krishnamurthy, B. (2008). Key differences between Web 1.0 and Web 2.0. *First Monday*, 13(6).

[44] Derpanis, K. G. (2005). Mean shift clustering. *Lecture Notes. http://www. cse. yorku. ca/˜ kosta/CompVis_Notes/mean_shift. pdf.*

[45] DiNucci, D. (2012). Fragmented Future (1999). *Dostupn é z: http://www. darcyd. com/fragmented_future. pdf*.

[46] Engelbart, D. C. and English, W. K. (1968). A research center for augmenting human intellect. In *Proc. of the December 9-11, 1968, fall joint computer conference, part I*, pages 395–410. ACM.

[47] Eno, J., Gauch, S., and Thompson, C. W. (2010). Linking behavior in a virtual world environment. In *Proc. of the 15th International Conference on Web 3D Technology*, pages 157–164. ACM.

[48] Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., and Dredze, M. (2010). Annotating named entities in Twitter data with crowdsourcing. In *Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88. Association for Computational Linguistics.

[49] Furukawa, Y. and Ponce, J. (2010). Accurate, dense, and robust multiview stereopsis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1362–1376.

[50] Hanson, A. J. and Wernert, E. A. (1997). Constrained 3D navigation with 2D controllers. In *Visualization'97., Proceedings*, pages 175–182. IEEE.

[51] Hirth, M., Hoßfeld, T., and Tran-Gia, P. (2011). Anatomy of a crowdsourcing platform-using the example of microworkers. com. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, pages 322–329. IEEE.

[52] Howe, J. (2006). The rise of crowdsourcing. *Wired magazine*, 14(6):1–4.

[53] Hughes, S., Brusilovsky, P., and Lewis, M. (2002). Adaptive navigation support in 3D e-commerce activities. In *Proc. of Workshop on Recommendation and Personalization in eCommerce at AH*, pages 132–139. Citeseer.

[54] Ieronutti, L. and Chittaro, L. (2007). Employing virtual humans for education and training in X3D/VRML worlds. *Computers & Education*, 49(1):93–109.

[55] Igarashi, T., Kadobayashi, R., Mase, K., and Tanaka, H. (1998). Path drawing for 3D walkthrough. In *Proc. of the 11th annual ACM symposium on User interface software and technology*, pages 173–174. ACM.

[56] Jankowski, J. (2011a). *Hypertextualized Virtual Environments: Dual-Mode User Interface Design*. PhD thesis.

[57] Jankowski, J. (2011b). A taskonomy of 3d web use. In *Proc. of the 16th International Conference on 3D Web Technology*, pages 93–100. ACM.

[58] Jankowski, J. (2012). Evaluation of static vs. animated views in 3d web user interfaces. In *Proc. of the 17th International Conference on 3D Web Technology*, pages 182–182. ACM.

[59] Jankowski, J. and Decker, S. (2009). 2lip: Filling the gap between the current and the three-dimensional web. In *Proc. of the 14th International Conference on 3D Web Technology*, pages 181–190. ACM.

[60] Jankowski, J. and Decker, S. (2012). A dual-mode user interface for accessing 3d content on the world wide web. In *Proc. of the 21st international conference on World Wide Web*, pages 1047–1056. ACM.

[61] Jankowski, J. and Decker, S. (2013). On the Design of a Dual-Mode User Interface for Accessing 3D Content on the World Wide Web. *International Journal of Human-Computer Studies*.

[62] Jankowski, J. and Hachet, M. (2012). A Survey of Interaction Techniques for Interactive 3D Environments. In *Eurographics 2013-State of the Art Reports*, pages 65–93. The Eurographics Association.

[63] King, A. B. (2003). *Speed up your site: web site optimization*. New Riders.

[64] Krug, S. (2009). *Don't make me think: A common sense approach to web usability*. Pearson Education.

[65] Lee, C. H., Varshney, A., and Jacobs, D. W. (2005). Mesh saliency. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 659–666. ACM.

[66] Lee, F. S., Vogel, D., and Limayem, M. (2003). Virtual community informatics: A review and research agenda. *Journal of Information Technology Theory and Application (JITTA)*, 5(1):5.

[67] Lepouras, G. and Vassilakis, C. (2004). Virtual museums for all: employing game technology for edutainment. *Virtual reality*, 8(2):96–106.

[68] Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. (2010). TurKit: Human Computation Algorithms on Mechanical Turk. In *Proc. of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 57–66, New York, NY, USA. ACM.

[69] Mackinlay, J. D., Card, S. K., and Robertson, G. G. (1990). Rapid controlled movement through a virtual 3D workspace. In *ACM SIGGRAPH Computer Graphics*, volume 24, pages 171–176. ACM.

[70] Malone, T., Laubacher, R., and Dellarocas, C. (2009). Harnessing crowds: Mapping the genome of collective intelligence.

[71] Möller, T. and Trumbore, B. (1997). Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*, 2(1):21–28.

[72] Nelson, T. H. (1965). Complex information processing: a file structure for the complex, the changing and the indeterminate. In *Proc. of the 1965 20th national conference*, pages 84–100. ACM.

[73] Nielsen, J. (1995). *Multimedia and hypertext: the Internet and beyond*. Morgan Kaufmann.

[74] Nielsen, J. (2000). *Designing for the Web*. New Riders Publishing.

[75] Nielsen, J. and Loranger, H. (2006). *Prioritizing web usability*. Pearson Education.

[76] O'reilly, T. (2005). What is web 2.0.

[77] Pandzic, I. S. and Forchheimer, R. (2003). *MPEG-4 facial animation: the standard, implementation and applications*. John Wiley & Sons.

[78] Ruddle, R. A., Howes, A., Payne, S. J., and Jones, D. M. (2000). The effects of hyperlinks on navigation in virtual environments. *International Journal of Human-Computer Studies*, 53(4):551–581.

[79] Rusinkiewicz, S., Hall-Holt, O., and Levoy, M. (2002). Real-time 3D model acquisition. In *ACM Transactions on Graphics (TOG)*, volume 21, pages 438–446. ACM.

[80] Russell, B. C., Martin-Brualla, R., Butler, D. J., Seitz, S. M., and Zettlemoyer, L. (2013). 3D Wikipedia: using online text to automatically label and navigate reconstructed geometry. *ACM Transactions on Graphics (TOG)*, 32(6):193.

[81] Salek, M., Bachrach, Y., and Key, P. (2013). Hotspotting–A Probabilistic Graphical Model For Image Object Localization Through Crowdsourcing.

[82] Sutherland, I. E. (1965). The ultimate display. *Multimedia: From Wagner to virtual reality*.

[83] Vázquez, P.-P., Feixas, M., Sbert, M., and Heidrich, W. (2001). Viewpoint Selection using Viewpoint Entropy. In *VMV*, volume 1, pages 273–280.

[84] Wang, Y., Yu, Q., and Fesenmaier, D. R. (2002). Defining the virtual tourist community: implications for tourism marketing. *Tourism management*, 23(4):407–417.

[85] Welch, T. A. (1984). A Technique for High-Performance Data Compression. *Computer*, 17(6):8–19.

[86] Wernert, E. A. and Hanson, A. J. (1999). A framework for assisted exploration with collaboration. In *Visualization'99. Proceedings*, pages 241–529. IEEE.