

# Rendering Interactive Plugin-less 3D Models in a Web Browser

Trisha Malhotra<sup>1</sup>;Kunal Nayyar<sup>2</sup>; Krishna Teja V<sup>3</sup>; Prof. Shilpa Nimbre<sup>4</sup>  
<sup>1</sup>trish6.malhotra@yahoo.in; <sup>2</sup>k4konal@gmail.com; <sup>3</sup>abstact470@gmail.com; <sup>4</sup>shilpa\_gonbare@yahoo.co.in;  
Department of Computer Engineering,  
K.C. College of Engineering & Management Studies & Research,  
Kopri, Thane (E)-400 603, India.

**Abstract** — With the advent of 3D technology, everything in 3D is deemed superior than in 2D. 3D graphics can be difficult, especially 3D in a web browser. 3D in a browser unlocks many potential to how we use a website, from blogging to E-commerce. Instead of viewing images, users can be given the option to view the model or item in 3D, with real textures and lighting effects. This paper presents an approach to integrate 3D models created and designed in 3D modeling softwares like AutoCAD 123D and Blender, with web to view them in web browsers. Instead of using pure WebGL, we combine it with Three.js which is a JavaScript 3D library. All the heavy lifting is done with three.js without sacrificing much flexibility. It is compatible with 3D Max, Maya and Blender making it easier to model and thereby introduce in the web format. Loading the model needs compression and decompression. Depending on the model size and internet bandwidth the latency may vary. Experiments with many 3D models show promising results. Our solution leans upon rendering the 3D graphical data along with real-time interactivity as an important feature, which can revolutionize the web in a subtle yet strong way.

**Keywords** — 3D model; Realtime; Server; Web browser; Pluginless; Graphics Intensive.

## I. INTRODUCTION

Everything today stands digitized. No single day passes when one has not gazed an electronic screen be it a computer or a cell phone. 3D graphics can be difficult, especially 3D in the browser. A step ahead in 2D visualization of real world is 3D. 3D in a browser unlocks much potential on to how we use a website, from blogging to E-commerce. Instead of viewing images of certain products in E-commerce websites, users can be given the option to view the model in 3D, with real textures and lighting effects.

3D graphics can be difficult, especially 3D in the browser. It is compatible with softwares like 3D Max, Maya and Blender making it easier to model and thereby introduce in the web format. To overcome this limitation, this research focuses on utilizing HTML5 and WebGL. Applying such approach, 3D capabilities, can be realized directly in the browser without any need for the additional plugin or extension. Another benefit is, WebGL provides hardware accelerated 3D functionality on the web, resulting significant performance improvement. This research deals with visualization and analysis of 3D objects. Three.js is a library that makes WebGL - 3D in the browser - very easy. While a simple cube in raw WebGL would turn out hundreds of lines of JavaScript and shader code, a Three.js equivalent is only a fraction of that. Frameworks like three.js make it a bit easier, but the official documentation is still under construction and only is supported by the desktop versions of Chrome, Firefox, or Safari. Unfortunately, WebGL does not work on mobile browsers yet, and we need Internet Explorer with version 10 or more. The visualization of 3D content through the web is now

possible thanks to specific formats like X3D, STL and technologies like the very recent but still in development WebGL specification, which makes the GPU controllable by JavaScript, and basic norms like HTML 5. The Web3D concept (i.e. communicating 3D content on the web) is seen as the future of the Web 2.0, and is supported by many organizations like the W3C and the Web3D consortium. Also in most of these applications, 3D data are represented by two main types i) polygonal meshes, which model the surface of the 3D objects by a set of vertices and edges, ii) and facets.

## II. LITERATURE SURVEY

In this digital age, everyone is busy with their lives depending on digital assistance. Web has the answer to all questions. It is omnipresent and no one can escape from it. Technological developments introduced 3D modeling. They are interesting to work with as 3D models give a real life view of all items and things depicted by it. 3D in web browser can be a challenging project to work on. Using WebGL and JavaScript have their own benefits. 3D models created in modeling softwares are integrated to HTML via Canvas element - which is apart of HTML5. WebGL has full integration with all Document Object Model (DOM) interfaces. It is a DOM API and therefore could be used from any DOM-compatible language: JavaScript, Java or Objective C. WebGL is supposed to be direct part of the most common web browsers - no plug-ins necessary. It is integrated in WebKit. WebGL is based on OpenGL ES 2.0 and designed to be very efficient. Both hardware and software support for displaying realistic 3D graphics are improving every

day. Therefore, there are many technologies that exist to present 3D data on the internet.

Jean-Francis Balaguer and Enrico Gobbetti presented 'i3D: A High-Speed 3D Web Browser'[1]. In this paper, they have presented i3D, it is a system that combines the 3D input and high-performance rendering capabilities of high-end virtual reality systems with the data fetching abilities of network browsers. Spaceball needs to be used so that the user can intuitively navigate inside the three-dimensional data, and by selecting 3D objects with the mouse will trigger requests for access to remote media documents that can be text, still images, animations or even other 3D models. It is implemented on top of OpenGL and uses VRML. Currently used at CERN. The user manager is responsible for sensing and analyzing the user's movements and actions. The protocol manager is responsible for the retrieval of media documents from the World Wide Web. The database manager maintains the state of the 3D scenes in order to provide the necessary geometrical information and visual attributes for the user and rendering managers to perform their tasks. Since it requires specific hardware and software like the Spaceball, hence is not for the common public.

Marcos A Rodrigues, Mariza Kormann and Lucy Davison presented 'A Case Study of 3D Technologies in Higher Education: Scanning the Metalwork Collection of Museums Sheffield and its Implications to Teaching and Learning'[2]. This paper describes results from the fast 3D scanning project conducted at Sheffield Hallam University in collaboration with Museums Sheffield. It discusses the steps in the process from scanning and noise removal to 3D post-processing and how the resulting 3D models can be made available on a standard web browser. First the artifacts are scanned and its textures mapped. This results in a 3D model. The 3D model is made interactive by using OpenGL (Open Graphics Library) and JavaScript functions which act as a wrapper (because browser doesn't directly understand OpenGL) in the web browser. The model is then implemented using HTML.

M. Auera, G. Agugiarob, N. Billena, L. Loosa, A. Zipf presented 'Web-based Visualization And Query Of Semantically Segmented Multiresolution 3d Models In The Field Of Cultural Heritage'[3]. This paper presents a database schema to organize not only segmented models but also different Levels-of-Details and other representations of the same entity. As service for the delivery of the segmented models a standardization candidate of the OpenGeospatial Consortium (OGC), the Web3DService (W3DS) has been extended to cope with the new database schema and deliver a web friendly format for WebGL rendering. Then a generic user interface is presented which uses the segments as navigation metaphor to browse and query the semantic segmentation levels and retrieve information from an external database of the German Archaeological Institute (DAI). MapView is set up only by MapActivity. All of these threads are in the control of the Activity life cycle, as MapView uses the file system and network in the background. Before using the map, an api-key for the Map service needs to be applied from Google.

Bartosz Sawicki and Bartosz Chaber presents 'Efficient visualization of 3D models by web browser '[4]. Proposed solution based on progressive mesh streaming is compared with server side rendering approach. In the first solution, servers render 3D scene, compute view and send it as an image to the client. Mesh data are transmitted to the client in the second approach which finally renders it. The main advantage of server side rendering is that mesh doesn't have to be sent over the network, which is especially important when we deal with large scale models. However client side rendering could utilise hardware acceleration, and provide more natural user navigation. Unfortunately large meshes can't be rendered on the client side, because its computational power would be exceeded. Hence progressive mesh algorithms are used.

Guillaume Lamoure, Laurent Chevalieri, Florent Dupontz presented 'Streaming Compressed 3D Data on the Web using JavaScript and WebGL'[5]. This paper is a dedicated to progressive compression algorithm for 3D graphic data with colours. It produces a binary compressed format allowing a progressive decompression with several levels of details. JavaScript half edge data structure allows complex geometrical and topological operations on a 3D mesh. The multi-thread JavaScript / WebGL implementation of the decompression scheme provides 3D data streaming in a web browser.

TABLE I. COMPARISON OF DIFFERENT SURVEY PAPERS

I	Comparison of different survey papers			
	Name	Technology Used	Time cost	Memory cost
1	Web-based Visualization And Query Of Semantically Segmented Multiresolution 3d Models In The Field Of Cultural Heritage	Maya, W3DS, WebGL	Moderate	Very high
2	i3D: A High-Speed 3D Web Browser	OpenGL, VRML	Moderate to Fast	High
3	A Case Study of 3D Technologies in Higher Education: Scanning the Metalwork Collection of Museums Sheffield	OpenGL, Java	Very High	Low
4	Efficient visualization of 3D models by web browser	Progressive Mesh Algorithm	Low	Very high
5	Streaming Compressed 3D Data on the Web using JavaScript and WebGL	JavaScript, WebGL	Low	Low

### III. METHODOLOGY

This section of the paper provides an overview of the method adopted for proposed system. The finished website consists of client side and server side functions. Both are explained in the following section.

#### A. The Server Side

The proposed server at a remote location will contain all the 3D models created using Blender and AutoDesk123D. When the server receives a request for a particular model, it retrieves it from its database and after applying compression algorithms sends the model to the client. The whole process is done in a matter of seconds.

#### B. The Client Side

The Client Side is the public domain of the website which will be available to the end user. Using HTML a minimalistic website is created that will have information about the product and a space for interacting with the 3D model. The 3D model will be rendered using WebGL which is an open source graphics library. JavaScript functions will be used to add a layer of interactivity with the model. The user will be able to zoom, rotate, turn, slide and click the object for different functions and information. The data will be displayed real-time.

#### C. Application development

The models will be developed using Blender and AutoDesk123D. The server side scripting will be done using PHP and the Client side programming will include JavaScript and HTML.

### IV. ADVANTAGES

#### A. Pluginless

The main advantage of this method needs no external plugins in the web browser.

#### B. Improved User Experience

This project provides a unique and interactive way for the users to see and gain knowledge about the product they wish to purchase.

#### C. Compression Techniques Employed

For large models, i.e., more than 20 MB compression is done before sending it from the server to reduce transfer time and charges.

#### D. Speed and Performance

High quality and faster loading times are the advantages of using WebGL and JavaScript.

### V. DISADVANTAGES

#### A. Mobile platform support

Unfortunately the mobile systems haven't been developed enough to support this system

#### B. Handling large objects

Very large 3D models cannot be handled effectively by WebGL.

#### C. Maintaining balance between Quality and Speed

A delicate balance between quality of the model, level of interactivity and loading time has to be achieved through various tweaks and trials.

#### D. Consistency

The whole process should run successfully every time in various heterogeneous systems with the same results.

### VI. RESULTS

Partial results are displayed as follows.

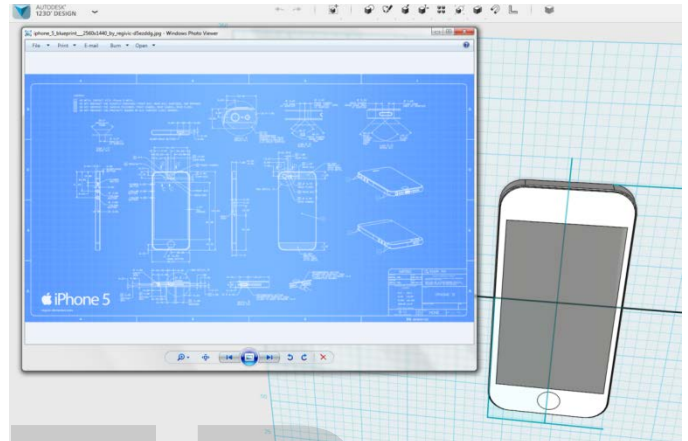


Figure 1: Model 1- iPhone 5, designed in Autodesk 123D. Model outlines the metallic and glass finish. Size of the model is 2.738Mb in STL format.

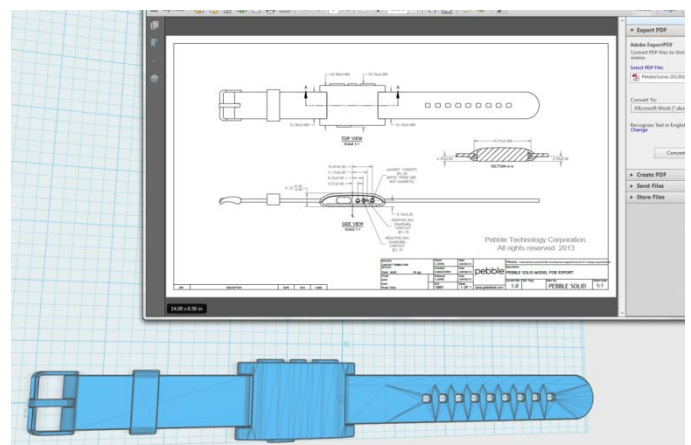


Figure 2: Model 2-Pebble Smart watch designed in Blender. Size of the model is 19.732Mb in STL format.

Here is the integrated model in the site viewed using Google Chrome browser. The GUI is basic as the test was to check how well the model is rendered and how fast it loads on local server.



Currently the task of producing each individual 3D model is an arduous one. Thus software or a unified process of developing 3D models just from specifications or pictures has to be created in order to make this a commercial and feasible

product. We are planning on creating and including many more object models with various file sizes, looks and material finishes. 3D viewing in a browser is thus a very apt concept in today's times and will take the internet user experience to greater heights..

## VII. REFERENCES

- [1] Jean-Francis Balaguer, Enrico Gobetti 'i3D: A High-Speed 3D Web Browser', Center for Advanced Studies, Research and Development in Sardinia.
- [2] Marcos A Rodrigues, Mariza Kormann, Lucy Davison 'A Case Study of 3D Technologies in Higher Education: Scanning the Metalwork Collection of Museums Sheffield and its Implications to Teaching and Learning', ITHET 2011: IEEE 10th International Conference on Information Technology Based Higher Education and Training.
- [3] M. Auera, G. Agugiariob, N. Billena, L. Loosa, A. Zipf 'Web-based Visualization And Query Of Semantically Segmented Multiresolution 3d Models In The Field Of Cultural Heritage' ISPRS Technical Commission V Symposium, 23 – 25 June 2014, Riva del Garda, Italy.
- [4] Bartosz Sawicki, Bartosz Chaber 'Efficient visualization of 3D models by web browser' Institute of Theory of Electrical Engineering, Measurements and Information Systems, Faculty of Electrical Engineering, Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warsaw, Poland.
- [5] Guillaume Lamoure, Laurent Chevaliery, Florent Dupontz 'Streaming Compressed 3D Data on the Web using JavaScript and WebGL', International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, No.1, February 2012.

IJSER