Rimbert Rudisch-Sommer / Christian Elsässer

Jedermann3D: A Web-Based Interface Targeted for the Inexperienced User to Edit 3D Models

108 - Recent Advances in Multimedia Processing, Organization and Visualization beyond Domains and Disciplines

Abstract

3D printers allow easy fabrication of individual customized 3D objects. Yet the customization of such objects requires special 3D software and expertise in using it. The goal of the project Jedermann3D presented in this demonstration paper is to provide customization functionality within a web site without extra installation requirements and to break down the complexity of such a process to a level manageable by everyone. Thus users without special training will be able to create their own individual 3D models ready for printing.

Keywords:

3D User Interface, WebGL, 3D Printing

1. Introduction

3D print is rapidly gaining in popularity. Today the areas of application are mainly in the industrial or health sector, like industrial prototypes, jewelry, or prosthetic dentistry. The individualization of physical objects for use in everyday life, like home decoration, or gifts is however getting increased attention from private users.

Currently it is not economical to purchase a 3D printer for home use. A number of printing services nevertheless exist which provide an upload option for 3D object files and then send the actual object in the mail. The task of creating and editing such a virtual 3D model ready for printing remains a great challenge however, especially for inexperienced users. The operation of the typical 3D software tool requires expert knowledge.

The idea of the project to be presented in this demonstration paper was to create a web-based UI as a service for enabling inexperience users (hence the name “Jedermann” meaning everyone) to easily edit 3D objects, which they can subsequently have printed.

2. Basic Idea and State of the Art

A comparable service – letting end-users edit individual products online for subsequent production and delivery – are photo-books. A number of photo-book online services exist enabling users to upload their images, arrange them on multiple pages, label them and create a special title page. Typically users can also choose from a number of predefined designs for the result.

The UI of those photo-book online services is typically easy to learn and inexperienced users quickly create satisfactory results. The reason lies in the simplicity of the tasks involved – the scaling, rotating, and moving of images and text. Furthermore many users have already worked with office application documents containing images or have created physical photo albums and are well acquainted with the application domain.

This is different to 3D Objects: Only a small number of users are experienced in editing physical (doing pottery, carving wood or stone) or virtual (using 3D software like Blender[1], Cinema 4D[2], ZBrush[3], etc.) 3D objects. For virtual editing, changing the camera position to view the object from different directions can immediately prove challenging for users new to the field because it requires advanced visual thinking. What's more, the operations in three dimensions are more complex than their counterparts in 2D.

Currently only some virtual sculpting tools exist for inexperienced end users. The more advanced come as IOS or Android apps (123DSculpt[4]). One printer manufacturer just recently also published a small web-based application (Dremel 3D[5])

3. Design and Implementation of Jedermann3D

The technology to integrate 3D graphics into websites is WebGL. The Web Graphics Library is an immediate mode 3D-rendering API designed for the web (Khronos 2015). 3D graphics can be rendered directly within current web-browsers[6] and can be manipulated via JavaScript. Those are the prerequisites of providing a UI for 3D manipulation on a webpage without requiring users to install extra software. The WebGL component is however rather young and little experience exists about its stability or performance.

In pursuit of solutions, two goals were set at the project's kickoff:

- Design of a User Interface making it possible to manipulate 3D objects also for inexperienced users and identifying the best set of operations necessary for individual manipulation for printable objects.

---

1 http://www.blender.org/
2 http://www.maxon.net/products/cinema-4d-studio/sculpting.html
3 http://pixologic.com/
4 http://www.123dapp.com/sculptplus
5 https://dremel3d.de/3d-printer-software-apps
6 http://caniuse.com/#feat=webgl

- Evaluation of the WebGL API regarding its functionality and performance. Is it possible to realize the conceptualized UI with it, and furthermore, what are the functional or performance limits?

The basic idea for reducing the complexity of a typical 3D software was to define the sequential steps necessary to customize a given 3D object and ultimately present them to the user in form of a “wizard” (Tidwell 2010). At all times the user should get immediate visual feedback and should be able to go back and forth in the manipulation chain.

The user should start out by selecting one out of a number of predefined objects to start with. The object is placed on a virtual grid. User feedback showed that it is important to give this feedback about the real size of the object. The maximal dimensions were defined to be 10 cm^3.

The position of the camera is changed with mouse movement: scrolling is zooming, moving the mouse with the left button leads to rotating the camera around the object and moving the mouse with right button allows lateral camera movement. This is the way most 3D software implements interaction and has been adopted very quickly also by inexperienced users. Additionally some standard camera positions have been defined like “front view” or “top view” which can easily be triggered by means of constantly visible buttons.

The operations applicable on a single 3D object basically were defined as:

- Changing the dimensions: scaling and skewing
- Changing the topology: tearing and pressing, “stamping”
- Changing the surface: colors, textures, images

Additionally, it should be possible to combine multiple objects to one final result and place it on a socket, where a label consisting of 3D characters can also be attached.

The afore-mentioned wizard should guide the user through these operations step by step. The order of these operations was carefully selected to provide the user the best feedback possible. As an example, one goal was to let users attach uploaded images onto the surface of the object. Texture maps have to be defined to project 2D textures onto the 3D object. Changing the topology of the object will however distort the images or textures applied. To mitigate this, the wizard defines it so that images have to be applied first, so users are able to directly experience the effects of topology changes on the images.

The final order of operations within the wizard was:

1. Object selection (one or more objects)
2. Surface definition (individually for each object, allowing image uploads)
3. Scaling and translation (also combining multiple objects)

4. Topology manipulation
5. Socket definition (size, label)

The step of topology manipulation is clearly the most challenging one. To support the user in creating satisfactory results, some assistance is provided, such as automatically mirroring operations to other axis for symmetric results or auto-rotating the object when applying operations to simulate a turning lathe. Naturally the results depend heavily on the initial resolution of the 3D object.

One of the test scenarios was an individualized Christmas tree: given a tree object, users were able to attach Christmas tree balls which they could customize in size and texture with uploaded photos. Additionally they could put a label on the base socket and add a recess for a tea candle.

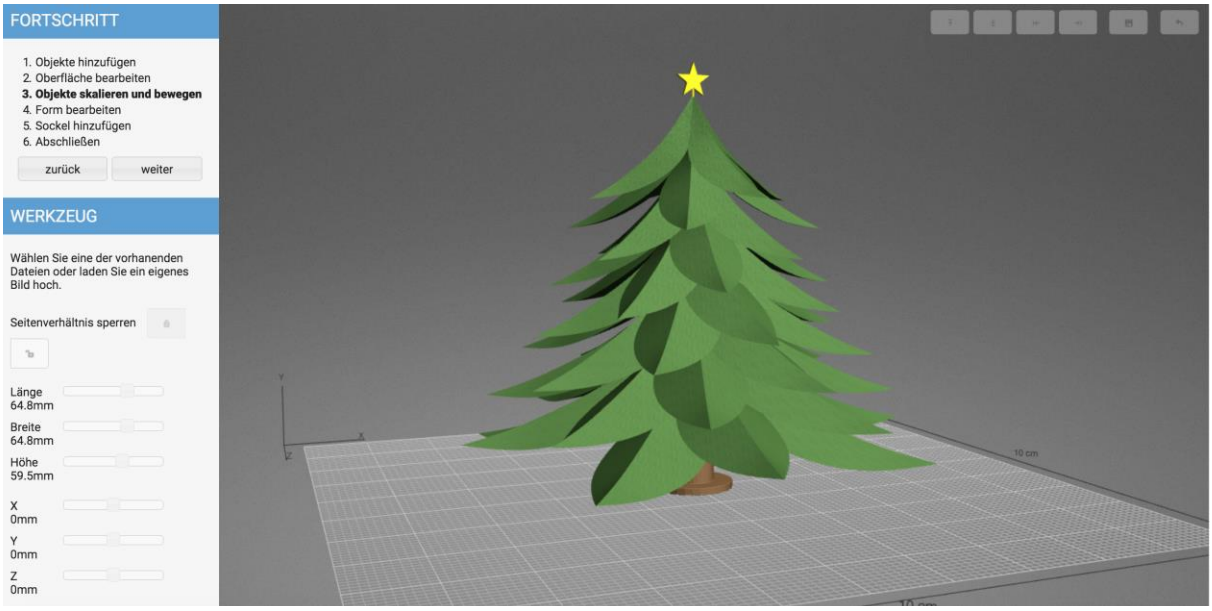

Figure 1: Jedermann3d User Interface

The project implementation allows for defining the set of operations applicable for each of the predefined objects. Tests showed that defining boundaries on the expected outcome of the customization process is experienced as helpful to the new users. Users were only allowed, for example, to change the size and texture of the Christmas tree balls but not their topology. Restricting these operations was not considered constrictive in this context.

As a next extension feature, the option of uploading individual objects to start the customization process from is planned. These objects could result from 3D scanners, which are nowadays available as smartphone appliances.

4. Conclusions

As stated before, the project Jedermann3D was started initially to research if it might be possible to create a UI for 3D manipulation in the web browser that allows inexperienced users to customize 3D models ready for 3D printing. The challenges were to break down the typical complexity of 3D software and to cope with possible inefficiencies of the browser based WebGL interface. The wizard guiding users step-by-step through the manipulation operations in a well-defined order was found to be an appropriate approach. Furthermore, the additional means implemented proved to be helpful, as with the defined set of operations per individual object or the manipulation assistance, as described above.

The WebGL interface was found to be surprisingly high-performing. Based on the library three.js[7], the client implementation is capable of manipulating objects in real-world scenarios.

For first-time users, the definition of a proposed end result with only a small number of customization possibilities has found to be satisfactory. With growing experience, the number of operations could later be extended or different scenarios for different levels of expertise could be defined.

References:

Khronos (2015): WebGL Specifications. https://www.khronos.org/registry/webgl/specs/latest/, (14.12.2015).

Tidwell, Jenifer (2010): Designing Interfaces, 2nd Edition. O'Reilly Media.

[7] http://threejs.org