

## Article

# Free and Open Source 3-D Model Customizer for Websites to Democratize Design with OpenSCAD

Yuenyong Nilsiam <sup>1</sup> and Joshua M. Pearce <sup>1,2,\*</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931, USA; ynilsiam@mtu.edu

<sup>2</sup> Department of Materials Science and Engineering, Michigan Technological University, Houghton, MI 49931, USA

\* Correspondence: pearce@mtu.edu; Tel.: +1-906-487-1466

Academic Editor: Hamid Reza Karimi

Received: 5 June 2017; Accepted: 27 June 2017; Published: 3 July 2017

**Abstract:** 3-D printing has entered the consumer market because of recent radical price declines. Consumers can save substantial money by offsetting purchases with DIY pre-designed 3-D printed products. However, even more value can be obtained with distributed manufacturing using mass customization. Unfortunately, the average consumer is not technically sophisticated enough to easily design their own products. One solution to this is the use of an overlay on OpenSCAD parametric code, although current solutions force users to relinquish all rights to their own designs. There is thus a substantial need in the open source design community for a libre 3-D model customizer, which can be used in any design repository to democratize design. This study reports on the design, function, and validation of such software: the Free Open Source 3-D Customizer. It is demonstrated with a case study of the customization of 3-D printable external breast prosthetics. The results showed that novice users can adjust the available parameters according to their needs and save these to a new file on a website. This PHP (recursive acronym for PHP: Hypertext Preprocessor) library is free and open source and has potential for increasing the usefulness of online repositories to enable distributed manufacturing using consumer customized 3-D printable products.

**Keywords:** rapid prototyping; RepRap; 3-D printing; OpenSCAD; customization; open hardware; free and open source software; custom designs; distributed manufacturing; open design

## 1. Introduction

The recent open source release of the self-replicating rapid prototyper (RepRap) 3-D printer [1–3] has democratized digital manufacturing and rapid prototyping [4]. The use of the open source hardware paradigm [5,6] led to a rapid technical evolution, which resulted in aggressive cost declines [4]. Rapid prototyping went from a technology only available to top-tier industrial firms and research and development centers able to afford 3-D printers that cost several hundred thousand dollars to a desktop 3-D printer market dominated by RepRap derivative machines [7,8]. Early adopters of these 3-D printers used them similarly to the way large industry was accustomed to using them, i.e. for prototyping new designs, but the maker community, who were the most avid users, began to use them for direct digital manufacturing and peer production [9]. One of the most enthusiastic communities to adopt digital peer-production with RepRaps was the scientific community, which produced a long list of 3-D printable scientific equipment to create entire open source labs [10–14]. Likewise, teachers adopted the technology in order to provide high-quality educational experiences in their classrooms [15,16]. Finally, the technically savvy wing of the international and sustainable development movement began making open source appropriate

technology (OSAT) for the developing world [17,18]. All of these initial groups of users were reasonably sophisticated designers (e.g. makers, scientists, teachers, and engineers working in development). They were experienced with traditional computer aided design (CAD) packages as well as a script based CAD package called OpenSCAD [19], which dominated the early designs in the RepRap community as the first coding environment and language for modeling mechanical 3-D objects [20].

The market for desktop 3-D printers, however, has moved away from the technically-sophisticated early adopters to the mass consumer market [21]. This is in part due to the unmet need for consumers to design and prototype their own products but is more likely driven in part by the economics of distributed manufacturing as two recent studies have shown that 3-D printing technology is lucrative to adopt for average consumers if they use pre-existing open source designs. In the first study, US \$500 for RepRap components was justified after only 25 h of printing readily available consumer products [22]. Building a RepRap is still non-trivial so a second study [23] investigated the economic case for a US \$1250 fully assembled RepRap 3-D printer (Lulzbot Mini, Aleph Objects, Inc., Loveland, Colorado, USA). The savings from off-setting purchases again were considerable as printing free designs could provide consumers with nearly a 1000% ROI (Return On Investment) over a printer lifetime of five years printing out only one product a week [23]. Both of these studies relied on the millions of free 3-D printable designs already available in the dozens of free online repositories. Until recently the most notable repository was Thingiverse, which was set up by Makerbot to popularize 3-D printers among consumers. Thingiverse has over two million free designs as of May 2017 in a large part due to its Customizer App, which enables designers to upload OpenSCAD code and make it readable for average consumers who can then create custom designs easily. Unfortunately, Makerbot, which was an early “open hardware hero” among companies, became closed source [24]. It was purchased in 2013 by a conventional additive manufacturing company, Stratasys, for \$403 million [25]. The abandonment of the open hardware ethos largely lost the support from the community [26] as it was viewed as a betrayal by Makerbot’s early technically savvy “maker” customers who had helped develop the RepRap machines on which Makerbot was based as well as their later products. Many of the best designers boycotted sharing new designs on Thingiverse with some even stripping down popular designs [27]. The class action lawsuit over the poor proprietary product designs [28] and economic misfortune that then befell Makerbot [29–31] can be at least in part attributed to the betrayal of the open source community. This caused a fracturing of the open source 3-D design community with dozens of other repositories under use [32]. In addition, the customizer app was locked away by a proprietary company that changed the license agreement on Thingiverse to force users to “irrevocably waive (and cause to be waived) any claims and assertions of moral rights or attribution with respect to your User Content” [33]. This forfeiture of design rights is unacceptable to many open source designers because the company is ignoring the license they choose for their work. This leaves the 3-D printable designer in a challenging position for which there are three sub-optimal solutions: (1) give up license rights to Makerbot/Stratasys while accessing the larger novice 3-D printer user base with customizable designs from the Thingiverse Customizer App; (2) maintain the open source rights to your own designs by publishing on another repository (e.g. Youmagine [34] or MyMiniFactory [35]) while only enabling the minority of technically sophisticated users to customize the design using OpenSCAD directly; or (3) develop your own single use customizer app (e.g. the handomatic, a customizer for the Enabling the Future community that crowd source 3-D printed hand prosthetics for children) [36–39].

There is thus a substantial need in the open source design community for a free and open source 3-D model customizer that can be used in any 3-D printing repository for customizing OpenSCAD code to democratize design. By making the software available to all repositories, it prevents user lock in if a repository changes their license agreement. This study reports on the design, function, and validation of such a free and open source software: the Free Open Source 3-D Customizer. An example of a breast prosthesis is used as a case study. The results are presented and

the impact of this code is discussed in the context of true distributed manufacturing using consumer customized 3-D printable products.

## 2. Materials and Methods

### 2.1. *Render-3d*

Render-3d is an open source PHP library that can be used to render 3-D file formats of STL (STereoLithography) and SCAD (OpenSCAD file) into images [40]. It is used on the Libre 3D website [41] and other open source 3-D community websites. It requires other free and open source software, including OpenSCAD [42], POV-Ray [43], and Composer [44], to be installed on the webserver for Render-3d to be able to operate.

OpenSCAD [42] is free and open source software for creating 3-D CAD models. It uses a script based input format. OpenSCAD reads a script file and renders it into a 3-D model. Users can edit the model by editing the code in the script file. OpenSCAD has the ability to use parameters in the script file (e.g. parametric design), which is essential for the open source Free 3-D Customizer described in this paper to interpret those parameters and offer user adjustable inputs (i.e. textboxes, dropdown boxes, ranges, and sliders). OpenSCAD is used to convert SCAD files (the OpenSCAD native file format) into STL files on the server. STL is a file format describing a 3-D model by using series of connected triangles to create the surface of the model and is currently the most popular file format for RepRap 3-D printers.

POV-Ray [43] is free and open source software for rendering 3-D images. It accepts a STL file as an input, generates a 3-D plot, and converts this into a PNG (Portable Network Graphics) image file.

Composer [44] is a dependency manager for PHP, and it is also free and open source software. Render-3d depends on other PHP libraries in order to operate successfully so it includes composer.json, which contains the required libraries for Render-3d. After a Composer execution, all required libraries should be installed and ready for Render-3d.

#### Render3d Class

Render-3d is a main PHP class of the Render-3d library that includes functions or methods to load 3-D model files, generate 3-D scenes, and save the scenes as images. The methods and their functions are listed below:

- `__construct` method is for initialing a root path for the Render-3d class.
- `workingDir` method is for getting and/or setting the current working directory. The working directory needs full permission to read, write, and execute (777).
- `sceneDir` method is for getting the scene directory that is used by Render-3d. This scene directory cannot be changed.
- `filename` method is for getting and setting the current working file. This method copies the original 3-D model file to the working directory and calls the file and filetype method.
- `file` method is for getting and setting the current base name of the processing files.
- `fileType` method is for getting and setting the current file type.
- `dirMask` method is for getting and setting the current directory permission for creating new directory.
- `executable` method is for getting and setting the location for the given command. This method is used for setting path and command for OpenSCAD and POV-Ray.
- `convertTo` method is for converting the current file to the given file type.
- `render` method is for rendering the 3-D model file into the image file and returning the full path of the image.
- `getConverter` method is for getting a converter to convert fromType to toType variables.
- `getRenderer` method is for getting renderer object of the given engine. The default engine is POV-Ray.
- `registerConverter` method is for registering a new converter object.

- registerRenderer method is for registering a rendering engine.
- cmd method is for running commands on the command line.
- options method is for getting and setting options used by converters and renderers.
- getBufferAndClean method is for getting contents in the buffer and then cleaning the buffer.
- startBuffer method is for starting the buffer.
- stopBuffer method is for stopping the buffer.

## 2.2. Free 3D Customizer

The Free 3-D Customizer is an extended PHP library of the Render-3d library. It provides methods to read a SCAD file and generate HTML elements for users to adjust parameters in the SCAD file. The source code of the library is available under a GNU (recursive acronym: GNU's Not Unix) Lesser General Public License v2.1 at <https://github.com/mtu-most/most-3-d-customizer>.

The Customizer class is an extended class of the Render-3d class, which means that it includes all the methods of Render-3d class listed in Section 2.1.1. The extended methods include:

- readSCAD method is for reading SCAD files and extracting all the adjustable parameters.
- writeSCAD method is for writing new SCAD files with customized parameter values.
- generateVname method is for generating formatted parameter names and returns a string of these.
- generateElement method is for generating HTML elements according to the possible values of each parameter. Elements are textbox, dropdown boxes, sliders, or ranges.

## 2.3. Variables in SCAD File

The format of variable declaration follows the customizer of MakerBot [45] so that all prior users of the MakerBot customizer can port their code to any/all 3-D printing repositories that utilize the Free 3-D Customizer. All variables must be placed at the top of a SCAD file before the first module in the file. The format of the variable description, variable name, value, and possible values must be in the following format:

```
// description
variable_name = value; // possible values
```

The description of a variable must be a one-line comment before the variable declaration. The variable name must separate each word by an underscore symbol. The default value must be given at the declaration of the variable. The possible values might be assigned in the same line of the variable declaration after the one line comment symbol (//).

### Possible Values Format

- Undefined possible values

If a variable is declared and a value assigned to it without calculation and depending on other variables, then that variable should be recognized as an adjustable parameter with a textbox. An example of a variable with undefined possible values is as follows:

```
hole_diameter = 2.5;
```

- Dropdown box possible values

If an option of possible values that is separated by commas is assigned after a declaration of a variable, then that variable should be recognized as an adjustable parameter with a dropdown box. The option values can be numbers, texts, and values with labels. Examples are as follows:

- Numbers
 

```
// How deep should the center hole be?
hole_depth = 1; // [0,1,5,50]
```
- Texts
 

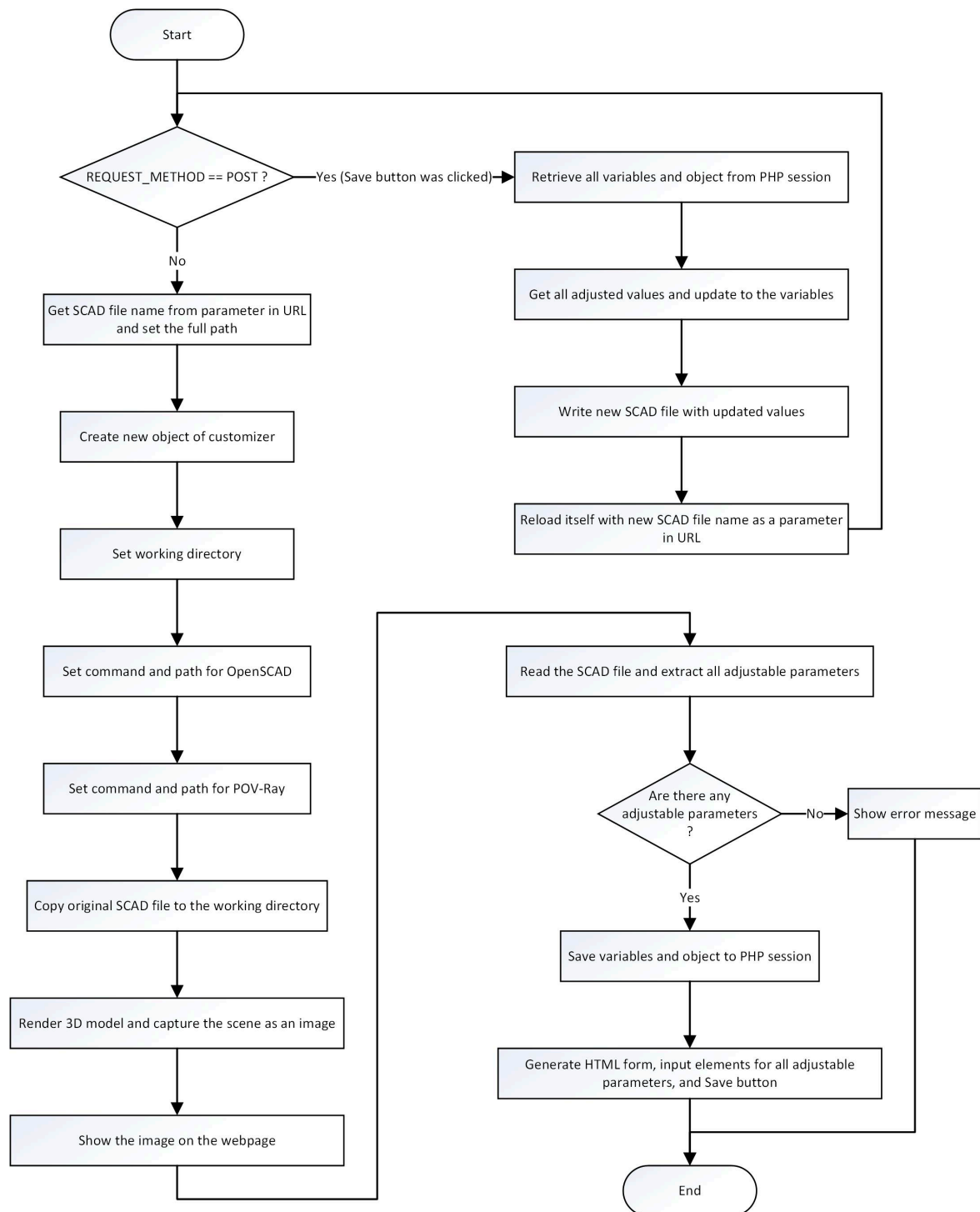
```
show_wheels = "yes"; // [yes,no]
```

- Values with labels  
cube\_size = 20; // [10:Small,20:Medium,30:Large]
- Slider possible values  
If a range of possible values that is separated by a colon for min, step, and max values is assigned after a declaration of a variable, then that variable should be recognized as an adjustable parameter with a slider. The range can be only max value, min and max values, and min, step, and max value. Examples are as follows:
  - Max value only (min value is set to zero)  
// How thick should the side wheels be?  
wheel\_thickness = 1; // [40]
  - Min and max value  
// How thick should the side wheels be?  
wheel\_thickness = 1; // [1:40]
  - Min, step, and max value  
// How thick should the side wheels be?  
wheel\_thickness = 1; // [1:0.5:40]

### 3. Results

The flowchart of the use of the Free Open Source 3-D Customizer is shown in Figure 1. First, the page checks the request method to determine if it is a post method or not. If it is not a post method, then the SCAD filename would be received from the parameter in the URL, the customizer object would be created, and the basic settings would be set. After that, the SCAD file would be copied to a working directory, rendered into an STL file, and captured as an image in a 3-D scene. The SCAD file would also be read for parameters, and a means for users to customize the design would be created on the webpage. When a user clicks on the save button, the request method would be posted so the page would retrieve all parameters and customizer objects in PHP sessions; then all adjusted parameters would be updated and written into a new SCAD file. Then the page would be reloaded with the new SCAD filename as a parameter in the URL.

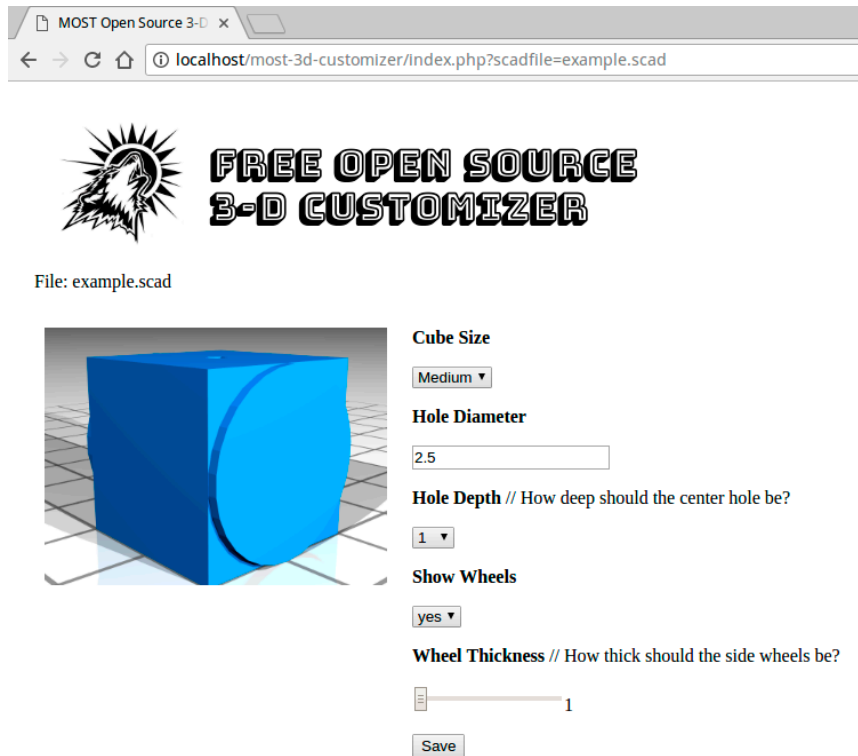
To show that the Customizer has been successfully developed, a PHP file named index.php was created to demonstrate the use of it.



**Figure 1.** The flowchart of the use of the Free Open Source 3-D Customizer.

### 3.1. PHP Files

The initial page of index.php is shown in Figure 2. The default cube size is medium, with a hole diameter of 2.5 mm and a depth of 1 mm along with 1 mm wheels. These parameters can be adjusted on the webpage by a dropdown menu (to a large cube, hole depth of five), increasing the hole diameter to 5 mm with text input, and the use of a slider for wheel thickness, as can be seen in Figure 3. The new SCAD file was saved, and the page was reloaded with the new file as in Figure 4.



**Figure 2.** The initial page of Free and Open Source 3-D Customizer loading the example SCAD file from Section 2.3



**Figure 3.** The parameters were adjusted on the Free Open Source 3-D Customizer page to create a new design.



**Figure 4.** The new SCAD file was saved, and the Free Open Source 3-D Customizer page was reloaded with this new SCAD file.

As can be seen in Figures 2–4, the code is effective and makes the customization of 3-D printable OpenSCAD designs easily approachable by a technically non-sophisticated user.

### 3.2. Practical Example: External Breast Prosthesis

In both the developed and less developed world, breast cancer is the most common type of cancer in women, with over 1.6 million new cases per year [46]. Cancer treatments have been improving, and patients with breast cancer can expect to significantly extend their lifespans with surgery and chemotherapy. Unfortunately, losing part of or the entire breast through surgery (for both breast conserving surgery and mastectomy) often leads to further problems during the recovery period, including psychosocial morbidity (anxiety, depression, body image/self-image, distress, anxiety, sexuality, and self-esteem) [47–49]. In addition, breast cancer patients often choose to avoid reconstructive surgery because of finances, physical conditions, or the recovery time [50]. For these women, external breast prosthesis (EBP) has been shown to be an effective solution [51] and is used by up to 90% of women after surgery in western countries [50]. The evidence that a customized breast prosthesis is effective is well established, although they are still not always used. The reasons for not using an EBP are comfort (either because of needing a custom size, the material selected being uncomfortable in summer or winter, or material selection causing rash), an un-natural appearance, and cost [52]. All of these problems can be addressed through the use of a customized 3-D printed EBP as FFF (fused filament fabrication) materials selection has increased substantially recently and there are several thermoplastic elastomers and thermoplastic polyurethanes on the market with many other flexible materials under development. Comfort can be addressed by custom sizing and the careful selection of 3-D printing material (along with possible post treatment with



silicone or other coating) and shape to ensure the patient's happiness with appearance. Finally, the cost of a distributed manufactured EBP can be significantly less than a custom commercial EBP. To enable female cancer patient survivors to custom design their own EBP, a customizable prosthetic breast model (breast.scad) is demonstrated here with the Free Open Source 3-D Customizer. The OpenSCAD code is as follows:

```
//Two methods to measure for breast prosthetic either width of desired breast size
(substitute in b directly) or Length over top of breast where l = b × (3/4)

// Using width of breast size or length over top of breast?
measure_choice = "width"; // [width,length]

// What is the value in mm?
value = 150;

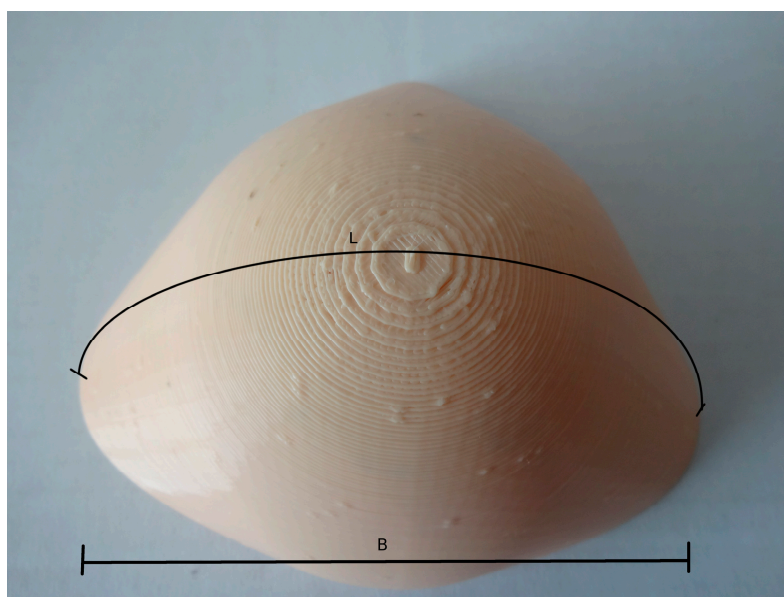
// Flat or rounded back?
flat_or_rounded = "flat"; // [flat,rounded]

//ignore these
b = (measure_choice == "width"? value:value × 4/3);
w = b × 2/3; //hidden sphere
$fn = 100/1;

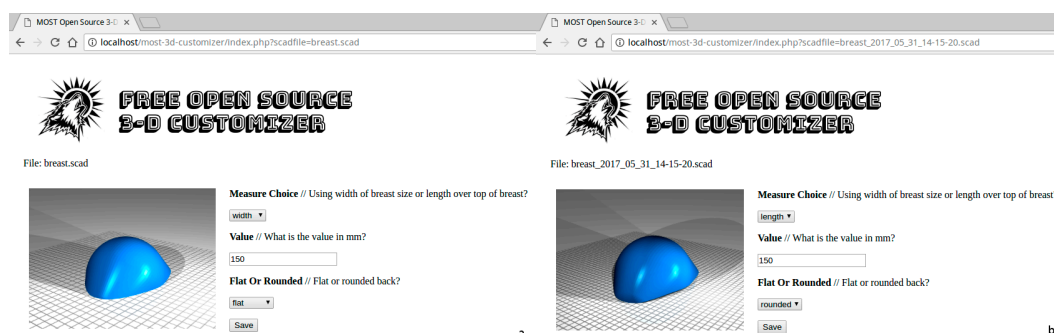
module breast(){
  difference(){
    hull(){
      sphere(w/1.75);//Sphere main
      translate([w/5,w/2,-w/8])sphere(w/3);//sphere left
      translate([w/5,-w/2,-w/8])sphere(w/3);//sphere right
      translate([0,0,w/2])sphere(w/5);//sphere top
    }
    if (flat_or_rounded == "flat") {
      translate([w,0,0])cube([w*2,w*2,w*2],center=true);//cleaving back with flat back
    }
    else {
      translate([w*5,0,0])sphere(w*5);//cleaving back with big sphere for round back
    }
  }
}

rotate([0,90,0])breast(); //print flat
```

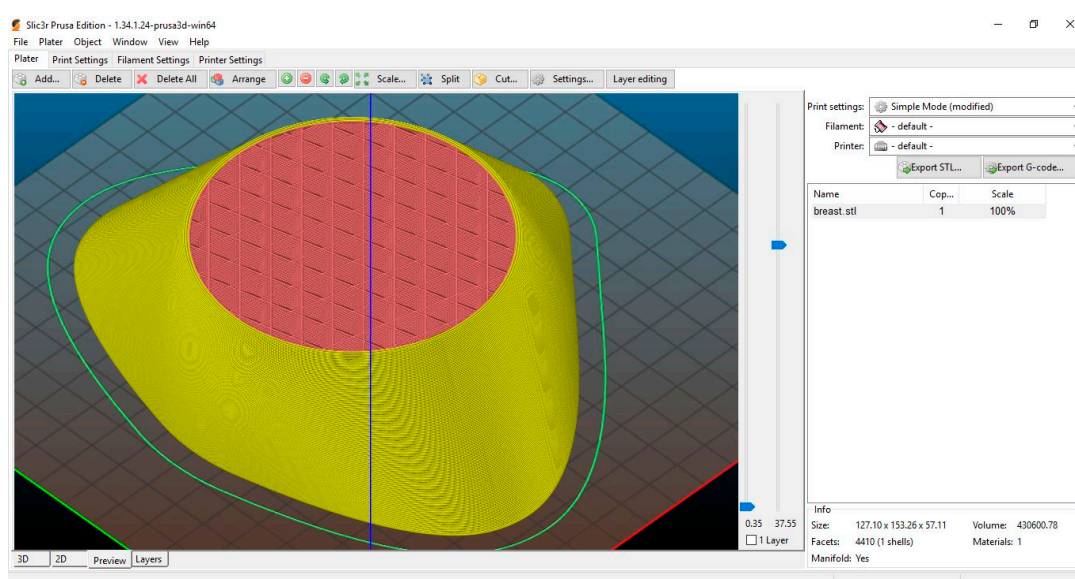
The measurement of the breast can be done in two ways; the width of the breast (B) for the desired width of the EBP or the length over the top (L) of the breast as shown in Figure 5. As shown in Figure 6, the user can choose to use either customizable method (by drop down menu) and then insert any value of in millimeters into the text box. Finally, the user can choose either a flat or rounded back using a second drop down menu. The former makes printing without support on a FFF 3-D printer easiest, and the latter would be more comfortable for the wearer of the EBP. It should be pointed out here that this is simply a first step in open source design for an EBP to demonstrate the Free Open Source 3-D Customizer, as the OpenSCAD code could be adapted in the future to help women with partially intact breasts or uneven breasts and even those simply wanting a custom bra insert to increase the appearance of breast size without surgery. In addition, the partial infill options in 3-D printing enable the potential for a degree of customization not possible with mass-produced fitted EBPs. This can be done with the use of an open source slicer such as Slic3r [53] and the new functionality offered by Slic3r Prusa Edition that includes the ability to ensure wall thickness to eliminate z step lines (as seen on the top of EBP in Figure 5) [54]. Thus users can vary the density of the EBP for optimal comfort and natural feel and look (as shown in Figure 7).



**Figure 5.** The measurement of the width of the breast (B) or the length over the top (L).



**Figure 6.** The prosthetic breast model customizable in size on the Free Open Source 3-D Customizer with different options (a) for flat width measurement and (b) for rounded length measurement.



**Figure 7.** Changing infill (red) of the prosthetic breast model in Slic3r Prusa Edition. The wall or skin (yellow) has no z step line.

### 3.3. Technical Details

As shown, this solution was viable. However, there are several methods that could have been used to provide similar solutions to those used in this study. For example, the passing of a SCAD file name parameter can be done through other methods such as using the POST of HTML request method. The example in the source code of this paper passing it as parameter in URL is just one solution to the problem.

The starting folder (start) and working folder (files) must be created with read and write permission as the Free Open Source 3-D Customizer needs to read and write to both folders. The working folder is written by Customizer when rendering the 3-D model and saving the scene as a PNG image. The starting folder is written when the parameters are adjusted and saved to a new SCAD file.

All paths on the server side should be absolute paths. However, paths on the client side should be relative paths such as paths to images.

If a new PHP class is added to the library, the Composer can be updated by running the command in the terminal where the Composer was installed as follows:

```
$ php composer.phar dump-autoloader.
```

For debugging the PHP file, using the echo command to print out variables is recommended. Otherwise, the var\_dump command can also be helpful to print out the structure and values inside a variable. If these commands do not print out anything, there must be some errors before the commands. The variable name is case-sensitive so typos are a common mistake that can cause errors.

Finally, to store an object in a PHP session, the object needs to be serialized before it is assigned to the session. Then, before retrieving the object from the session, it needs to be unserialized. The example is given in the index.php file.

## 4. Discussion

3-D printing is being used by ordinary people who have little technical knowledge to create their own custom products [55]. This is mass individualization [56], which provides the largest benefit of distributed manufacturing; the ability for users to customize products for their needs. This study demonstrated the ability to do this with the customization of EBPs. Thus, users can make themselves a custom EBP for a few dollars in raw materials and offset the purchase of commercial EBPs (which are not customized) that cost tens to hundreds of dollars. These abilities also bring business opportunities to start 3-D printing shops for customized items [57] (e.g. a business could use the code developed here to enable customers to easily customize products in store or at home on a web-browser that the 3-D print shop could then fabricate on demand). The current global value chains are affected by this distributed manufacturing model [58] and the work demonstrated would only accelerate this transition by allowing more consumers to participate.

This is important because it enables economic challenges to other markets. Examples of high-cost custom 3-D printable products include orthotics [59,60], laboratory equipment [61,62], self-adjustable glasses [63], photovoltaic racking [64], robotic creatures [65], terahertz (THz) lenses [66], prosthetic feet [67], and even more mundane products like toys [68].

The development of the Free Open Source Customizer shown here will also enable many projects to develop faster. For example, now the Enabling the Future project can iterate faster (as a new handomatic website does not need to be designed for each new hand design for incremental improvements). On a broader scale, open source 3-D abstract models or metamodels for 3-D printing in an open design community have encouraged knowledge reuse for customization and increased the number of open source 3-D models [69]. Knowledge reuse for customization is in the middle between reuse for replication and reuse for innovation. 3-D metamodels can be made up of OpenSCAD files that include parameters with possible values or ranges of possible values, which can be adjusted to generate new models. It is possible by the aggregation of variable choices that the new design may never have been considered by the original designer. 3-D metamodels are created

by experts and used by novices to generate variants of 3-D designs. Thus, designers can design not just a single object but all future objects in that class with careful parametric design and then make this available to novice users to make specific bespoke objects. In the past, on Thingiverse, the number of shared models increased significantly after the availability of the customizer app, which helped novice users to easily customize and generate new models. Here, with the Free Open Source Customizer, users of all websites that utilize the code will have that same design freedom. It is clear that the availability of 3-D metamodels with the Free and Open Source Customizer described here along with 3-D printing helps to enhance the case for distributed manufacturing. This ease of use for making custom products with 3-D printing provides an even stronger case to weaken existing patent law [70].

For future work, such a system can be expanded to automatically adapt the design with a choice of material based on the mechanical properties of the 3-D printer. In addition, the designs could be automatically adapted with an input of the STL scanned off of individual. For example, users could scan or use photogrammetry to input a design of their face, and then the customizer could automatically adapt a custom Halloween mask around the face shape of the user. The more advanced version would also automatically adapt the design with the change based on the materials, type, and/or model of the 3-D printer available to the user.

## 5. Conclusions

In this paper, the authors presented the development of an open source PHP library to support SCAD file customization by adjusting the shape parameters and then saving these to a new SCAD file. The new model can be generated from the file and is ready for 3-D printing. The Render-3d library had been extended to the Free Open Source 3-D Customizer library. Moreover, the example of a webpage using the Free Open Source 3-D Customizer library was successfully demonstrated along with the case study for the customization of the 3-D printable external breast prosthetic. The results showed that an appropriately-formatted SCAD script allows novice users to adjust available parameters according to the user's needs and to save it to a new file on the website. This PHP library is free and open source and has potential for increasing the usefulness of online repositories for open source 3-D models.

**Acknowledgments:** This study was supported by the Michigan Tech Open Sustainability Technology Lab.

**Author Contributions:** Joshua M. Pearce conceived and designed the experiments; Yuenyong Nilsiam wrote the code and performed the experiments; and Yuenyong Nilsiam and Joshua M. Pearce wrote the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sells, E.; Bailard, S.; Smith, Z.; Bowyer, A.; Olliver, V. RepRap: The replicating rapid prototyper-maximizing customizability by breeding the means of production. In Proceedings of the World Conference on Mass Customization and Personalization, Cambridge, MA, USA, 10–11 October 2010.
2. Jones, R.; Haufe, P.; Sells, E.; Iravani, P.; Olliver, V.; Palmer, C.; Bowyer, A. RepRap—The replicating rapid prototyper. *Robotica* **2011**, *29*, 177–191.
3. Bowyer, A. 3D printing and humanity's first imperfect replicator. *3D Print. Addit. Manuf.* **2014**, *1*, 4–5.
4. Rundle, G. *A Revolution in the Making*; Simon and Schuster: New York, NY, USA, 2014.
5. Gibb, A.; Abadie, S. *Building Open Source Hardware: DIY Manufacturing for Hackers and Makers*, 1st ed.; Addison-Wesley Professional: Boston, MA, USA, 2014.
6. Raymond, E. The cathedral and the bazaar. *Knowl. Technol. Policy* **1999**, *12*, 23–49. doi:10.1007/s12130-999-1026-0.
7. Wohlers, T. *Wohlers Report 2016: 3D Printing and Additive Manufacturing State of the Industry Annual Worldwide Progress Report*; Wohlers Associates Inc.: Fort Collins, CO, USA, 2016.
8. Frauenfelder, M. *Make: Ultimate Guide to 3D Printing 2014: Maker Media*; O'Reilly Inc.: Sepaspol, CA, USA, 2013.

9. Moilanen, J.; Vaden, T. 3D printing community and emerging practices of peer production. *First Monday* **2013**. doi:10.5210/fm.v18i8.4271.
10. Pearce, J.M. Building research equipment with free, open-source hardware. *Science* **2012**, *337*, 1303–1304. doi:10.1126/science.1228183.
11. Pearce, J. *Open-Source Lab: How to Build Your Own Hardware and Reduce Research Costs*; 1st ed.; Elsevier: Waltham, MA, USA, 2014.
12. Baden, T.; Chagas, A.; Marzullo, T.; Prieto-Godino, L.; Euler, T. Open laware: 3-D printing your own lab equipment. *PLoS Biol.* **2015**, *13*. doi:10.1371/journal.pbio.1002175.
13. Coakley, M.F.; Hurt, D.E.; Weber, N.; Mtingwa, M.; Fincher, E.C.; Alekseyev, V.; Chen, D.T.; Yun, A.; Gizaw, M.; Swan, J.; et al. The NIH 3D print exchange: A public resource for bioscientific and biomedical 3D prints. *3D Print. Addit. Manuf.* **2014**, *1*, 137–140.
14. Coakley, M.; Hurt, D.E. 3D Printing in the laboratory maximize time and funds with customized and open-source labware. *J. Lab. Autom.* **2016**, *21*. doi:10.1177/2211068216649578.
15. Kentzer, J.; Koch, B.; Thiim, M.; Jones, R.W.; Villumsen, E. An open source hardware-based mechatronics project: The replicating rapid 3-D printer. In Proceedings of the 4th International Conference on Mechatronics, Kuala Lumpur, Malaysia, 17–19 May 2011; pp. 1–8.
16. Schelly, C.; Anzalone, G.; Wijnen, B.; Pearce, J.M. Open-source 3-D printing technologies for education: Bringing additive manufacturing to the classroom. *J. Vis. Lang. Comput.* **2015**, *28*, 226–237.
17. Pearce, J.M.; Blair, C.; Laciak, K.J.; Andrews, R.; Nosrat, A.; Zelenika-Zovko, I. 3-D printing of open source appropriate technologies for self-directed sustainable development. *J. Sustain. Dev.* **2010**, *3*, 17–29.
18. Birtchnell, T.; Hoyle, W. *3D Printing for Development in the Global South: The 3D4D Challenge*; Springer: Berlin, Germany, 2014.
19. Kintel, M.; Wolf, C. *OpenSCAD, The Programmers Solid 3D CAD Modeller*; <http://www.openscad.org/>.
20. Valero-Gomez, A.; Gonzalez-Gomez, J.; Almagro, M.; Salichs, M.A. Boosting mechanical design with the C++ OOML and open source 3D printers. In Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON), Marrakech, Morocco, 17–20 April 2012; pp. 1–7.
21. Wohlers, T.T.; Caffrey, T. *Wohlers Report 2015: 3D Printing and Additive Manufacturing State of the Industry Annual Worldwide Progress Report*; Wohlers Associates: Fort Collins, CO, USA, 2015.
22. Wittbrodt, B.; Glover, A.; Laureto, J.; Anzalone, G.; Oppliger, D.; Irwin, J.; Pearce, J. Life-cycle economic analysis of distributed manufacturing with open-source 3-D printers. *Mechatronics* **2013**, *23*, 713–726.
23. Petersen, E.; Pearce, J. Emergence of home manufacturing in the developed world: Return on investment for open-source 3-D printers. *MDPI Tech.* **2017**, *5*, 7, doi:10.3390/technologies5010007.
24. Giseburt, R. Is one of our open source heroes going closed source? *Make Mag.* Available online: <http://makezine.com/2012/09/19/is-one-of-our-open-source-heroes-going-closed-source/> (accessed on 20 March 2017).
25. Terdiman, D. Stratasys acquires MakerBot in \$403 million deal. CNET. Available online: <https://www.cnet.com/news/stratasys-acquires-makerbot-in-403-million-deal/> (accessed on 20 March 2017).
26. Pearce, J.M. Emerging business models for open source hardware. *J. Open Hardw.* **2017**, *1*, 2. doi:10.5334/joh.4.
27. Prusa, J. Occupy thingiverse test cube. Available online: <http://www.thingiverse.com/thing:30808> (accessed on 20 March 2017).
28. Lawsuit. Consolidated amended complaint for violations of the federal securities laws. Available online: <https://cdn-shop.adafruit.com/pdfs/makerbot/classaction.pdf> (accessed on 20 March 2017).
29. Benchoff, B. Makerbot has now cut 36% of staff in last 6 months. Available online: <http://hackaday.com/2015/10/09/makerbot-has-now-cut-36-of-staff-in-last-6-months/> (accessed on 20 March 2017).
30. Benchoff, B. The MakerBot obituary. Available online: <http://hackaday.com/2016/04/28/the-makerbot-obituary/> (accessed on 20 March 2017).
31. Petch, M. 3D printing business MakerBot lays off 30% of staff, stratasys remains “committee” 3D printing industry. Available online: <https://3dprintingindustry.com/news/3d-printing-business-makerbot-lays-off-30-staff-stratasys-remain-committed-105846/> (accessed on 20 March 2017).
32. Printable part sources. Available online: [http://reprap.org/wiki/Printable\\_part\\_sources](http://reprap.org/wiki/Printable_part_sources) (accessed on 20 March 2017).

33. Makerbot terms of use. Available online: <https://www.thingiverse.com/legal/terms> (accessed on 20 March 2017).
34. Youmagine. Available online: <https://www.youmagine.com/> (accessed on 20 March 2017).
35. MyMiniFactory. Available online: <https://www.myminifactory.com/> (accessed on 20 March 2017).
36. Schull, J. Enabling the future: Crowdsourced 3D-printed prosthetics as a model for open source assistive Technology Innovation and mutual aid. In Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility Lisbon, Portugal, 26–28 October 2015.
37. Burn, M.B.; Ta, A.; Gogola, G.R. Three-dimensional printing of prosthetic hands for children. *J. Hand Surg.* **2016**, *41*, e103–e109.
38. King, M.; Phillips, B.; Shively, M.; Raman, V.; Fleishman, A.; Ritter, S.; Mehta, K. Optimization of prosthetic hand manufacturing. In Proceedings of the 2015 IEEE Global Humanitarian Technology Conference (GHTC), Seattle, WA, USA, 8–11 October 2015; pp. 59–65.
39. Handomatic. Available online: <http://enablingthefuture.org/handomatic/> (accessed on 20 March 2017).
40. Render-3d. Available online: <https://github.com/libre3d/render-3d> (accessed on 20 March 2017).
41. Libre 3D. Available online: <http://libre3d.com/> (accessed on 20 March 2017).
42. OpenSCAD. Available online: <http://www.openscad.org/> (accessed on 20 March 2017).
43. POV-Ray. Available online: <http://www.povray.org/> (accessed on 20 March 2017).
44. Composer. Available online: <https://getcomposer.org/> (accessed on 20 March 2017).
45. Developer documentation. Available online: <http://customizer.makerbot.com/docs> (accessed on 20 March 2017).
46. Borghesan, D.H.P.; Gravena, A.A.F.; Lopes, T.C.R.; Brischiliari, S.C.R.; de Oliveira, M.; Demitto, C.M.D.A.; de Barros Carvalho, M.D.; Pelloso, S.M. Variables that affect the satisfaction of Brazilian women with external breast prostheses after mastectomy. *Asian Pac. J. Cancer Prev.* **2014**, *15*, 9631–9634.
47. Montazeri, A. Health-related quality of life in breast cancer patients: A bibliographic review of the literature from 1974 to 2007. *J. Exp. Clin. Cancer Res.* **2008**, *27*, 32.
48. McArdle, J.M.; George, W.D.; McArdle, C.S.; Smith, D.C.; Moodie, A.R.; Hughson, A.M.; Murray, G.D. Psychological support for patients undergoing breast cancer surgery: A randomised study. *BMJ* **1996**, *312*, 813–816.
49. Al-Ghazal, S.K.; Fallowfield, L.; Blamey, R.W. Comparison of psychological aspects and patient satisfaction following breast conserving surgery, simple mastectomy and breast reconstruction. *Eur. J. Cancer* **2000**, *36*, 1938–1943.
50. Kubon, T.M.; McClennen, J.; Fitch, M.I.; McAndrew, A.; Anderson, J. A mixed-methods cohort study to determine perceived patient benefit in providing custom breast prostheses. *Curr. Oncol.* **2012**, *19*, e43.
51. Roberts, S.; Livingston, P.; White, V.; Gibbs, A. External breast prosthesis use: Experiences and views of women with breast cancer, breast care nurses, and prosthesis fitters. *Cancer Nurs.* **2003**, *26*, 179–186.
52. Liang, Y.N.; Xu, B. Factors influencing utilization and satisfaction with external breast prosthesis in patients with mastectomy: A systematic review. *Int. J. Nurs. Sci.* **2015**, *2*, 218–224.
53. Slic3r. Available online: <http://slic3r.org/> (accessed on 26 May 2017).
54. Prusa, J. Introducing slic3r prusa edition. Available online: <http://www.prusaprinters.org/introducing-slic3r-prusa-edition/> (accessed on 26 May 2017).
55. Brooks, G.; Kinsley, K.; Owens, T. 3D printing as a consumer technology business model. *Int. J. Manag. Inf. Syst.* **2014**, *18*, 271.
56. Jiang, P.; Leng, J.; Ding, K.; Gu, P.; Koren, Y. Social manufacturing as a sustainable paradigm for mass individualization. Proceedings of the Institution of Mechanical Engineers, Part B: *J. Eng. Manuf.* **2016**, *230*, 1961–1968.
57. Laplume, A.; Anzalone, G.C.; Pearce, J.M. Open-source, self-replicating 3-D printer factory for small-business manufacturing. *Int. J. Adv. Manuf. Technol.* **2016**, *85*, 633–642.
58. Laplume, A.O.; Petersen, B.; Pearce, J.M. Global value chains from a 3D printing perspective. *J. Int. Bus. Stud.* **2016**, *47*, 595–609.
59. Dombroski, C.E.; Balsdon, M.E.; Froats, A. The use of a low cost 3D scanning and printing tool in the manufacture of custom-made foot orthoses: A preliminary study. *BMC Res. Notes* **2014**, *7*, 443.
60. Wittbrodt, B.; Laureto, J.; Tymrak, B.; Pearce, J.M. Distributed manufacturing with 3-D printing: A case study of recreational vehicle solar photovoltaic mounting systems. *J. Frugal Innov.* **2015**, *1*, 1.

61. Zhang, C.; Wijnen, B.; Pearce, J.M. Open-source 3-D platform for low-cost scientific instrument ecosystem. *J. Labo Autom.* **2016**, *21*, 517–525.
62. Zhang, C.; Anzalone, N.C.; Faria, R.P.; Pearce, J.M. Open-source 3D-printable optics equipment. *PLoS ONE* **2013**, *8*, e59840.
63. Gwamuri, J.; Wittbrodt, B.T.; Anzalone, N.C.; Pearce, J.M. Reversing the trend of large scale and centralization in manufacturing: The case of distributed manufacturing of customizable 3-D-printable self-adjustable glasses. *Chall. Sustain.* **2014**, *2*, 30–40.
64. Wittbrodt, B.; Pearce, J.M. 3-D printing solar photovoltaic racking in developing world. *Energy Sustain. Dev.* **2017**, *36*, 1–5.
65. Megaro, V.; Thomaszewski, B.; Nitti, M.; Hilliges, O.; Gross, M.; Coros, S. Interactive design of 3D-printable robotic creatures. *ACM Trans. Gr.* **2015**, *34*, 216.
66. Busch, S.F.; Weidenbach, M.; Fey, M.; Schäfer, F.; Probst, T.; Koch, M. Optical properties of 3D printable plastics in the THz regime and their application for 3D printed THz optics. *J. Infrared Millim. Terahertz Waves* **2014**, *35*, 993–997.
67. Yap, J.; Renda, G. Low-cost 3D-printable prosthetic foot. In Proceedings of the 3rd European Conference on Design4Health, Sheffield, UK, 13–16 July 2015.
68. Petersen, E.E.; Kidd, R.W.; Pearce, J.M. Impact of DIY Home Manufacturing with 3-D printing on the toy and game market, to be published.
69. Kyrialou, H.; Nickerson, J.V.; Sabinis, G. Knowledge reuse for customization: Metamodels in an open design community for 3D printing. *MIS Q.* **2017**, *41*, 315–332.
70. Osborn, L.S.; Pearce, J.; Haselhuhn, A. A Case for weakening patent rights. *St. John's Law Rev.* **2015**, *89*, 1185–1253.



© 2017 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).