

# Thirty Thousand 3D Models from Thingiverse

Felix W. Baumann 

TWT GmbH Science & Innovation, 70565 Stuttgart, Germany; felix.baumann@tw-t-gmbh.de;  
Tel.: +49-711-21-57-77 (ext. 1207)

Received: 18 December 2017; Accepted: 16 January 2018; Published: 18 January 2018

**Abstract:** This dataset contains files and geometrical analysis of 3D model data, acquired from the thingiverse online repository. More than thirty thousand stereolithography files (STL) were retrieved and analysed. The geometrical analysis of the respective models is presented along with model renderings in both GIF and PNG format, and pre-sliced machine instructions as GCode. This dataset is intended to be used as a basis for further research in Additive Manufacturing (AM), such as 3D printing time estimation, printability assessment or slicing algorithm development. All files retrieved are user-generated, with the respective user and associated licence presented in the overview. The dataset was acquired between 2016 and 2017.

**Dataset:** 10.5281/zenodo.1098527

**Dataset License:** CC-BY

**Keywords:** additive manufacturing; 3D printing; online repository; user-generated content; 3D model; geometrical analysis

## 1. Introduction

There is a large number of online repositories for digital models for consumers, e.g., <https://grabcad.com>, <https://3dwarehouse.sketchup.com>, or <https://www.tinkercad.com/things>. Some of these repositories are directly aimed for digital models for additive manufacturing or 3D printing. Additive manufacturing (AM) and 3D printing are used synonymously in this work. The most prominent is *thingiverse* [1]. Such repositories enable users to share their work, acquire and mix other user's content. Furthermore, *thingiverse* caters for its users by offering community functionality that enables discussions on specific 3D printing related topics, such as models, parameter selection, handling and management issues, and 3D printer selection.

The focus of *thingiverse* is consumer-centric with a number of abilities to upload, host, share, re-mix, discuss, and acquire objects via 3rd party providers. Monetisation is also enabled in some online repositories, i.e., sell, their designs. An indication for this community centricity is the availability of public statistics on the object, e.g., download numbers and view counter, the integration of third party social media capabilities, e.g., share on Facebook [2] or Twitter [3], and the availability of "like" and other indicators of appreciation.

*Thingiverse* can be regarded as the most prominent example of 3D printing model data by the metric of results on the *Google Scholar* index for scientific publications. On this index searching for *thingiverse* yields 2640 results compared to 54 results for *Repables*, 174 results for *YouMagine*, 1010 results for *GrabCAD*, 740 results for *tinkerCAD*, 1970 results for *3D Warehouse*, and 134 results for *MakePrintable*. All results exclude citations and patents. These are other available repositories for digital models for 3D printing. Furthermore, *thingiverse* has been identified as the largest model repository for this domain in Alcock et al. [4], as one of two most common sources for digital models in Evans [5] and as a highly recommended source for digital models by Mallon [6].

Acquiring digital models from online repositories is one method to provide the required digital models for 3D printing. Other methods are the construction or design of models, using modeling or CAD (Computer Aided Design) software, or the reverse engineering of existing objects using laser scanning equipment.

Online repositories are a suitable source of digital models for end-users or consumers with limited abilities for reverse-engineering or 3D-modeling skills.

Furthermore, such online repositories are suitable to extract trends and concerns of consumers, thus, allowing for market or scientific research into crowds [7] or for research into aesthetic design [8].

For this work, the terms (online) repository, (online) community and platform are used synonymously in the context of the *thingiverse* platform. This repository is for user generated digital models and files, and functions as a community for collaboration and communication on these models and related issues. The rationale for this work is to provide a dataset of user-generated 3D models which are used for geometrical analysis. This analysis is intended to be used to generate models for 3D printing time estimations, based on available information. The separately published research focuses on Fused Deposition Modeling (FDM) type 3D printing. In FDM, a semi-molten plastic material is extruded and forms the physical object, layer by layer. FDM 3D printers are commonly encountered in private settings [9] but also in professional enterprises.

Research into user-generated content such as with this study must be analyzed for applicability into professional, e.g., industrial, domains, where digital models are commonly of greater complexity, must be inquired in further research. See Green et al. [10] for an example of commercialized user designed objects. This dataset can be beneficial for compiling model sets of varying degrees of complexity for machine learning, printability and optimization research.

The *thingiverse* platform was chosen as it is a very well-known platform, with a large number of files and provides essential meta-data for the models in an easy to use, i.e., machine-readable, form. User-generated content for 3D printing is suitable for the generation of research datasets usable for machine learning algorithms and benchmarks as they reflect actual content which is of interest to end users. Furthermore, information from such public repositories is usable for trend-analysis which can be of interest to the idea generation phase of product development. The models are generally designed for 3D printing on a Fused Deposition Modeling (FDM) type 3D printer [11].

This work presents a dataset of digital models, pre-sliced machine instructions, model renderings and geometrical analysis, acquired from *thingiverse*. Over 30,000 models were acquired, analyzed and are presented hereinafter. This dataset is prepared to enable other researchers to test slicing algorithms, analyze model geometry, use in machine learning scenarios and provide a historical insight in user-generated content for 3D printing. This work is an extension to Baumann and Roller [12], in which about 10,000 models were acquired and analyzed for their dominant license choice. For the introduction, textual elements were re-used from this text as the underlying problem is nearly identical.

The remainder of this work is structured as follows. In Section 2, the data acquisition process is described. Following, in Section 3, the acquired data is presented in an overview. Section 4 explains the individual fields in the dataset. In Section 5, the utilized software and the metric hash generation is detailed. This work is concluded with a summary in Section 6.

## 2. Acquisition

The model files were downloaded from *thingiverse* using a custom *bash* script. Models on *thingiverse* are part of so called *Things*, which are identified by a unique integer number. Multiple models and other files, such as renderings or pre-sliced machine instructions, can be part of a *Thing*. Each *Thing* has one creator. Furthermore, each *Thing* can have:

- Download counter, indicating how often parts of the *Thing* have been downloaded;
- View counter, indicating how often the *Thing* has been watched in a browser by a third party;
- Like counter, similar to other online communities,

- Watch counter, indicating how often the *Thing* has been added to watch lists by other users. Watch lists are for users to keep track on changes of other users *Things*;
- Remix counter, indicating how often the respective *Thing* was remixed, i.e., re-used in other user's designs,
- Collection counter, indicating how often the respective *Thing* is added to collections or groups of *Things*, most often on a specific topic or theme;
- Zero or more comments by members of the online community;
- Zero or more tags, that categorize the *Thing* according to the creator;
- A textual description;
- An associated license, detailing the conditions under which the *Thing* can be used

Each *Thing* can be directly access via a Web browser by its *ThingID*. For the acquisition of the models, an upper limit of 1,609,996 of the *ThingID* was set following a previous experiment. In the data collection script, 100,000 *ThingIDs* were randomly tried for availability. If a *Thing* with the randomly tried *ThingID* was available, the first file from it was downloaded and the above information stored.

The downloaded files were consecutively analysed geometrically by custom scripts written in *BASH* and Python. The files were furthermore converted to GCode, using the *Slic3r* software, and AMF. See Section 5 for details on the utilized software. Renderings of the models were created in PNG and GIF format.

The usage of the acquired data is covered under the German *Urheberrechts-Wissensgesellschafts-Gesetz*. (<https://www.bmjv.de/SharedDocs/Gesetzgebungsverfahren/Dokumente/BGBl-UrhWissG.html>) which allows for the re-distribution of 15% of the original content for the purpose of research and, furthermore, allows for the text and data mining of existing corpora. *Thingiverse* claims to have 981,600 models (<https://www.thingiverse.com/about>) indexed, the extracted 31,121 models are far less than the allowed 15%. Under US law, fair-use (<https://www.copyright.gov/fair-use>) of this data is claimed as no commercial interest is pursued by the author.

### 3. Results

A total of 31,121 STL files were downloaded using the method described in Section 2. The data is uploaded to *Zenodo* under DOI: 10.5281/zenodo.1098527. For this manuscript, an overview of the retrieved data is presented as an HTML (Hypertext Markup Language) document, with links to the respective files. This overview document is available as supplementary material to this manuscript, see below. *Zenodo* does not allow large numbers of individual files in depositions, therefore, the deposition contains aggregations as archives. The archives are of ZIP format and are separated by file type. The following archives are available.

- all-stl.zip, contains 31,121 files of a total (uncompressed) size of approximately 66.2 GiB.
- all-gif.zip, contains 31,120 files of a total (uncompressed) size of approximately 13.0 GiB.
- all-png.zip, contains 31,121 files of a total (uncompressed) size of approximately 2.4 GiB.
- all-amf.zip, contains 31,032 files of a total (uncompressed) size of approximately 30.0 GiB.
- [0-9]-gcode.zip, contains 30,515 files of a total (uncompressed) size of approximately 230.0 GiB. The archives are sorted by the first digit of the filename.

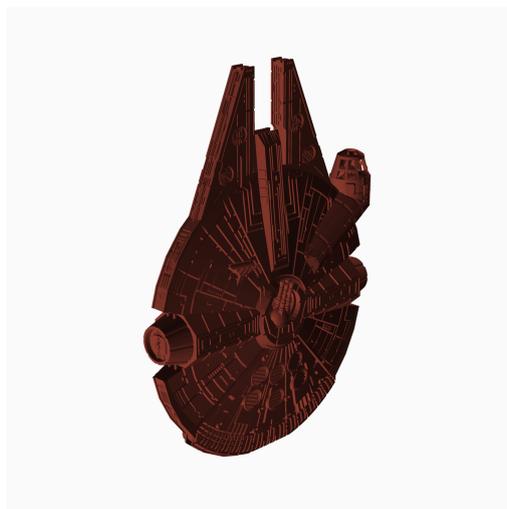
Archives with fewer than 31,121 files are due to conversion or creation errors.

To illustrate the nature of the acquired files, the following figures are presented that have the highest numbers of downloads.

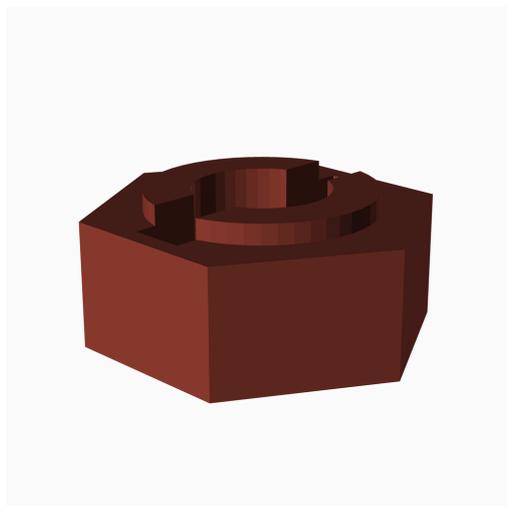
- Thing ID: 86187, "Royal manticore navy", by "jamesH" with 10,4701 downloads; See Figure 1;
- Thing ID: 919475, "fillenium malcon", by "aaskedall" with 101,438 downloads; See Figure 2;
- Thing ID: 42198, "OpenRC 1:10 4WD Truggy Concept RC Car", by "barspin" with 99,346 downloads; See Figure 3;



**Figure 1.** Royal manticore navy, by jamesH.



**Figure 2.** fillenium malcon, by aaskedall.



**Figure 3.** OpenRC 1:10 4WD Truggy Concept RC Car, by barspin.

In Section 4, the content of the individual columns is explained. Furthermore, the overview is provided as a Comma-separated Values (CSV) file for ease of use in software.

#### 4. Dataset Description

The dataset is provided as an overview in form of an HTML document. The individual columns of the tabular representation are presented in the following with short explanations. Missing entries are indicated by *N/A*. From the CSV overview document 57 entries were removed due to malfunctioned encoding. The CSV file contains 31,064 entries.

- **File #**, internal file number
- **Filename**, the original name of the downloaded STL file.
- **Filesize [Byte]**, the file size of the original STL file in bytes.
- **Thing ID**, the ID of the *Thing* for which the model was downloaded. This entry is enriched with a hypertext URL that points to the *Thing* on <https://thingiverse.com>.
- **Download Date**, datestamp when the file was downloaded from <https://thingiverse.com>.
- **Download Time**, timestamp when the file was downloaded from <https://thingiverse.com>.
- **Processing Time [s]**, time that was required to analyse the model file geometrically. Processing time is measured in seconds.
- **Thingname**, the original title provided for the *Thing* of which the download is part of.
- **License**, the license information under which the original *Thing* is made available.
- **User**, the original creator of the model and the *Thing*. This entry is enriched with a hypertext URL that points to the user's page on <https://thingiverse.com>.
- **Creation Date**, the data on which the original *Thing* was uploaded to <https://thingiverse.com>.
- **Likes**, the number of likes registered on <https://thingiverse.com> for the specific *Thing* as of the date on which the file was downloaded.
- **Comments**, the number of comments registered on <https://thingiverse.com> for the specific *Thing* as of the date on which the file was downloaded.
- **Views**, the number of views registered on <https://thingiverse.com> for the specific *Thing* as of the date on which the file was downloaded.
- **Downloads**, the number of downloads registered on <https://thingiverse.com> for the specific *Thing* as of the date on which the file was downloaded.
- **# of Files**, the number of files that are registered for the specific *Thing* on <https://thingiverse.com>.
- **Tags**, the tags that are registered for this specific *Thing*. These entries are enriched with hypertext URLs that link to the tag collection site on <https://thingiverse.com>. In the CSV file, the tags are separated by vertical bars, |.
- **Genus**, describes the number of holes that are in the model. Calculated as  $Genus = (Vertices - Edges + Facets - 2) / -2$ .
- **Vertices**, the number of vertices in the model file as reported by the analysis script utilizing the *freecad* library.
- **Edges**, the number of edges in the model file as reported by the analysis script utilizing the *freecad* library.
- **Facets**, the number of facets in the model file as reported by the analysis script utilizing the *freecad* library.
- **Max Distance**, the maximum distance between any two vertices in from the random selection of 20% of all vertices in the model.
- **Min Distance**, the minimum distance between any two vertices in from the random selection of 20% of all vertices in the model.
- **Metric Hash**, the hash value of the derived metric describing the object. Described in detail in Section 5.4.

- **Rendering (PNG)**, the name of the PNG rendering of the model.
- **Rendering (GIF)**, the name of the GIF rendering of the model.
- **# of Moves**, the total number of moves that are present in the GCode file. This is derived from the *G1* instructions in the file.
- **# of Non-Extrusion Moves**, the number of moves that are present in the GCode file which do not instruct to extrude material. This is derived from the *G1* instructions in the file where the extrusion parameter, i.e., *A*, *B*, or *E*, is not larger than the same parameter in the previous extruding instruction.
- **# of X-Only Direction Moves**, the number of moves that are only in the X direction. Derived from the *G1* command in the file where both Y and Z parameters are not set or are equal to the parameters in a previous movement instruction.
- **# of Y-Only Direction Moves**, the number of moves that are only in the Y direction. Derived from the *G1* command in the file where both X and Z parameters are not set or are equal to the parameters in a previous movement instruction.
- **# of Z-Only Direction Moves**, the number of moves that are only in the Z direction. Derived from the *G1* command in the file where both X and Y parameters are not set or are equal to the parameters in a previous movement instruction.
- **# of X Dominant Moves**, the number of moves where the amount of movement in the X direction is dominant over movement in any other direction. Derived from the *G1* command in the file.
- **# of Y Dominant Moves**, the number of moves where the amount of movement in the Y direction is dominant over movement in any other direction. Derived from the *G1* command in the file.
- **# of Z Dominant Moves**, the number of moves where the amount of movement in the Z direction is dominant over movement in any other direction. Derived from the *G1* command in the file.
- **# of X-Y Equal Moves**, the number of moves where the distance traveled in the X and Y direction are equal, i.e., diagonal moves. Derived from the *G1* command in the file.
- **Distance Traveled X [mm]**, the total distance traveled in the X direction over the complete file. The individual distances for each movement are summarised and yield this result.
- **Distance Traveled Y [mm]**, the total distance traveled in the Y direction over the complete file. The individual distances for each movement are summarised and yield this result.
- **Distance Traveled Z [mm]**, the total distance traveled in the Z direction over the complete file. The individual distances for each movement are summarised and yield this result.
- **Avg. Length Move X Direction [mm]**, the average length of a movement in the X direction over the complete file. This value is calculated by dividing the total length traveled in the X direction by the number of movements involving the X axis.
- **Avg. Length Move Y Direction [mm]**, the average length of a movement in the Y direction over the complete file. This value is calculated by dividing the total length traveled in the X direction by the number of movements involving the Y axis.
- **Avg. Length Move Z Direction [mm]**, the average length of a movement in the Z direction over the complete file. This value is calculated by dividing the total length traveled in the X direction by the number of movements involving the Z axis.
- **# Instructions involving X-axis**, the total number of instructions that involve the X axis. This value includes extrusion and non-extrusion instructions.
- **# Instructions involving Y-axis**, the total number of instructions that involve the Y axis. This value includes extrusion and non-extrusion instructions.
- **# Instructions involving Z-axis**, the total number of instructions that involve the Z axis. This value includes extrusion and non-extrusion instructions.
- **# of Lines**, the total number of lines present in the GCode file. This value includes blank lines and comments.

- **# of Instructions**, the number of instructions in the GCode file that are not comments or other non-instructing lines.
- **Extrusion (A) Length [mm]**, the amount of material extruded in the 3D printing process by the first extruder. Most GCode dialects support two extruders (A and B) per 3D printer. When the currently active extruder is selected in the GCode, extrusion instructions via the *E* parameter are also possible.
- **Extrusion (B) Length [mm]**, the amount of material extruded in the 3D printing process by the second extruder.
- **Extrusion (E) Length [mm]**, the amount of material extruded in the 3D printing process by the all extruders.
- **Filament Estimation (GPX) [m]**, the filament required for 3D printing the complete object as estimated by the *GPX* software.
- **Print Time Estimation (GPX) [s]**, the processing time for 3D printing the complete object as estimated by the *GPX* software.
- **Size X Dim [mm]**, the size of the bounding box enclosing the model in the X direction.
- **Size Y Dim [mm]**, the size of the bounding box enclosing the model in the Y direction.
- **Size Z Dim [mm]**, the size of the bounding box enclosing the model in the Z direction.
- **# Facets**, the number of facets of the model as reported by the *admesh* software.
- **Shell**, the number of shells of the model as reported by the *admesh* software.
- **Model Volume [mm<sup>3</sup>]**, the volume of the model in mm<sup>3</sup> as reported by the *admesh* software.
- **GCode File**, the pre-sliced GCode file generated by the *Slic3r* software for the given layer-height and infill. Further parameters are available in the Appendix A.
- **Layer Height [mm]**, the layer height in mm with which the GCode file is created by the *Slic3r* software. A layer height of 0.3 mm is set as a default.
- **Infill [%]**, the percentage of infill with which the GCode file is created by the *Slic3r* software. 20% infill is set as a default value that is reflected by the value 0.2 in the results.
- **Filesize Gcode [Byte]**, the file size of the pre-sliced GCode file created by the *Slic3r* software in bytes.
- **Filament Used (*Slic3r*) [mm]**, the amount of filament to be extruded in the 3D printing process as calculated by the *Slic3r* software for the given infill and layer height.
- **AMF File**, the AMF file that is converted from the STL file and represents the identical model. The conversion script from *Slic3r* is utilized for the conversion.
- **File size AMF [Byte]**, the file size of the AMF file in bytes.

## 5. Software

In this section, the utilized software is described. The complete analysis is performed on a desktop computer (Intel(R) Core(TM) i5-4570, 16 GiB RAM) running the Linux operating system (ArchLinux, <https://www.archlinux.org>).

### 5.1. GPX

*GPX* [13] is a conversion software that prepares files for 3D printing on *Makerbot* [14] 3D printers. The *Makerbot* 3D printers use a proprietary GCode format called X3G. *Makerbot* is the owner of *thingiverse*. This software is used to calculate a printing time estimation. The version used is 2.5.3.

### 5.2. Slic3r

*Slic3r* [15] is an open source slicing software that prepares STL model files for 3D printing by producing a machine instruction file, i.e., a GCode file. The parameters and configuration used to create the GCode files are presented in the Appendix A. The version used is 0.9.11-dev.

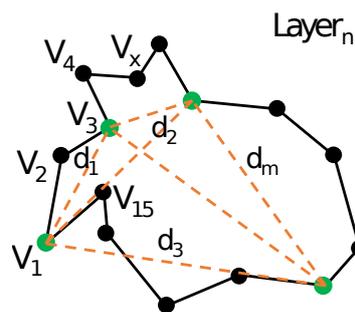
### 5.3. Admesh

*Admesh* [16] is a software to read, analyse and repair STL, both binary and ASCII, files. It is used to generate geometrical information on the model. The version used is 0.98.2.

### 5.4. Metric Hash

In order to make model files comparable, a metric is devised that is based on the geometrical representation of the model. For this metric, random samples of 20% of all vertices are selected pairwise. The distance between those two vertices is calculated and normalized by the maximum distance between all vertices in this sample. Ten buckets between 0.0 and 1.0 are created with equal distance. The metric is derived by counting how often a pairwise distance occurs in each of these buckets. The 10 values are then converted to a hexadecimal representation for readability. This metric is designed to yield similar results for geometrically similar objects. Evaluation of this metric is not yet performed. See Appendix B for the implementation of the metric calculation as a *BASH* [17] script that converts the individual elements to a hexadecimal representation.

The following Figure 4 illustrates the principle to derive the metric for similarity. In this figure, the vertices  $V_x$  with  $x = 1, \dots, 15$ , are all vertices for one single layer or height of the model. From the totality of the vertices in each layer  $n$ , a subset is randomly selected and the distances between each vertex pair is calculated, see  $d_m$  with  $m = 1, \dots, 6$ . The distances are then normalized and processed as described in Appendix B.



**Figure 4.** Distance Calculation for Similarity Metric

## 6. Summary

This dataset is provided to facilitate research in AM. User-generated content from the *thingiverse* repository was selected, retrieved and analyzed to further research in areas such as historical development of user-generated content and associated meta-data, slicing algorithm development, 3D printing time estimation, or complexity analysis. The data was retrieved in an automated manner via the repository's website. The data was analyzed and processed using open-source software. This manuscript presents the overview over the complete dataset as an HTML document.

**Supplementary Materials:** The following are available online at [www.mdpi.com/2306-5729/3/1/5](http://www.mdpi.com/2306-5729/3/1/5), Table S1: Dataset Overview Document.

**Acknowledgments:** No funding was received for this work.

**Author Contributions:** This work was conducted by Felix W. Baumann.

**Conflicts of Interest:** The author declares no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

AM	Additive Manufacturing
AMF	Additive Manufacturing File Format
CSV	Comma-separated Values
GCode	Machine Code
GiB	gibibyte
GIF	Graphics Interchange Format
HTML	Hypertext Markup Language
STL	Stereolithography File Format
PNG	Portable Network Graphics
ZIP	Zip File Format

## Appendix A. *Slic3r* Parameters

```
avoid_crossing_perimeters = 0
bed_shape = 0x0,200x0,200x200,0x200
bed_temperature = 125
before_layer_gcode =
bridge_acceleration = 0
bridge_fan_speed = 100
brim_width = 0
complete_objects = 0
cooling = 1
default_acceleration = 0
disable_fan_first_layers = 3
duplicate_distance = 6
end_gcode = M104 S0; turn off temperature\nG28 X0; home X axis\nM84;
disable motors\n
extruder_clearance_height = 20
extruder_clearance_radius = 20
extruder_offset = 0x0
extrusion_axis = E
extrusion_multiplier = 1
fan_always_on = 0
fan_below_layer_time = 60
filament_colour = #FFFFFF
filament_diameter = 1.75
first_layer_acceleration = 0
first_layer_bed_temperature = 0
first_layer_extrusion_width = 200%
first_layer_speed = 30
first_layer_temperature = 250
gcode_arcs = 0
gcode_comments = 0
gcode_flavor = makerware
infill_acceleration = 0
infill_first = 0
layer_gcode =
max_fan_speed = 100
max_print_speed = 80
```

```
max_volumetric_speed = 0
min_fan_speed = 35
min_print_speed = 10
min_skirt_length = 0
notes =
nozzle_diameter = 0.4
only_retract_when_crossing_perimeters = 1
ooze_prevention = 0
output_filename_format = [input_filename_base].gcode
perimeter_acceleration = 0
post_process =
pressure_advance = 0
resolution = 0
retract_before_travel = 2
retract_layer_change = 0
retract_length = 2
retract_length_toolchange = 10
retract_lift = 0
retract_restart_extra = 0
retract_restart_extra_toolchange = 0
retract_speed = 40
skirt_distance = 6
skirt_height = 1
skirts = 1
slowdown_below_layer_time = 5
spiral_vase = 0
standby_temperature_delta = -5
start_gcode = G28; home all axes\nG1 Z5 F5000; lift nozzle\n
temperature = 240
threads = 2
toolchange_gcode =
travel_speed = 130
use_firmware_retraction = 0
use_relative_e_distances = 0
use_volumetric_e = 0
vibration_limit = 0
wipe = 0
z_offset = 0
dont_support_bridges = 1
extrusion_width = 0
first_layer_height = 0.35
infill_only_where_needed = 0
interface_shells = 0
layer_height = 0.2
raft_layers = 0
seam_position = aligned
support_material = 0
support_material_angle = 0
support_material_contact_distance = 0.2
support_material_enforce_layers = 0
```

```
support_material_extruder = 1
support_material_extrusion_width = 0
support_material_interface_extruder = 1
support_material_interface_layers = 3
support_material_interface_spacing = 0
support_material_interface_speed = 100%
support_material_pattern = pillars
support_material_spacing = 2.5
support_material_speed = 60
support_material_threshold = 0
xy_size_compensation = 0
bottom_solid_layers = 3
bridge_flow_ratio = 1
bridge_speed = 60
external_fill_pattern = rectilinear
external_perimeter_extrusion_width = 0
external_perimeter_speed = 50%
external_perimeters_first = 0
extra_perimeters = 1
fill_angle = 45
fill_density = 30%
fill_pattern = honeycomb
gap_fill_speed = 20
infill_every_layers = 1
infill_extruder = 1
infill_extrusion_width = 0
infill_overlap = 15%
infill_speed = 80
overhangs = 1
perimeter_extruder = 1
perimeter_extrusion_width = 0
perimeter_speed = 60
perimeters = 3
small_perimeter_speed = 15
solid_infill_below_area = 70
solid_infill_every_layers = 0
solid_infill_extruder = 1
solid_infill_extrusion_width = 0
solid_infill_speed = 20
thin_walls = 1
top_infill_extrusion_width = 0
top_solid_infill_speed = 15
top_solid_layers = 3
```

## Appendix B. Metric Hash Calculation

Listing 1: Metric Hash Conversion Script

```
#!/bin/bash

STR=${1}
CLEAN_STR=$( echo "${STR}" | tr -d "\[\],")
```

```

for elem in ${CLEAN_STR}; do
  # multiplying by 4096 (16x16x16) because
  # we want to have 3 hex-digits for each value

  HEX=$( bc <<< "obase=16;_${elem}_*_4096" )
  if [ "${HEX}" == "0" ]; then
    echo -n "000"
  else
    if [ "${HEX:2:1}" == "." ]; then
      echo -n "0${HEX:0:1}${HEX:1:1}"
    elif [ "${HEX:1:1}" == "." ]; then
      echo -n "000"
    elif [ "${HEX:0:1}" == "." ]; then
      echo -n "000"
    else
      echo -n "${HEX:0:1}${HEX:1:1}${HEX:2:1}"
    fi
  fi
done

```

## Appendix C. Geometrical Analysis

Listing 2: Geometrical Analysis Script

```

#!/usr/bin/python

import datetime as dt
import numpy as np
import sys
import random

t1 =dt.datetime.now()

FREECADPATH = '/usr/lib/freecad/'
sys.path.append( FREECADPATH )

import FreeCAD
import Mesh
import Part
import os
import traceback

filename = sys.argv[ 1 ]
try:
  shapes = Mesh.Mesh( filename )
  print( "Part:_%s" ) % filename
  BNAME = os.path.basename( filename )
  print( "File:_%s" ) % BNAME

  shape = Part.Shape()

```

```

shapes.removeDuplicatedPoints()
shapes.removeDuplicatedFacets()
shapes.fixDegenerations()
shapes.fixSelfIntersections()
shape.makeShapeFromMesh( shapes.Topology, 0.001 )

solid = Part.makeSolid( shape )
print "shells:_%d" % len( solid.Shells )

E = len( solid.Edges )
F = len( solid.Faces )
V = len( solid.Vertexes )
max_dist=0
min_dist=sys.maxint

for k in solid.Vertexes:
    for l in solid.Vertexes:
        # avoid to calculate distance of vertex with itself
        if k.X != l.X and k.Y != l.Y and k.Z != l.Z:
            cur_dist = np.sqrt( ( k.X - l.X )**2
                                + ( k.Y - l.Y )**2 + ( k.Z - l.Z )**2 )
            if cur_dist > max_dist:
                max_dist = cur_dist
            if cur_dist < min_dist:
                min_dist = cur_dist

        # select 1000 random vertices
        # compare difference between them
        # scale to max_dist
        # and bin into 10 categories: max_dist/10 * x
        # count how many fit

spacing = range( 0, 10 )
spacing[ 0 ] = spacing[ 1 ]
    = spacing[ 2 ] = spacing[ 3 ] = spacing[ 4 ]
    = spacing[ 5 ] = spacing[ 6 ] = spacing[ 7 ]
    = spacing[ 8 ] = spacing[ 9 ] = 0
for i in range( 1, int (V / 20) ):
    r = random.randint( 0, V - 1 )
    k = random.randint( 0, V - 1 )

dist = np.sqrt( ( solid.Vertexes[ r ].X
                - solid.Vertexes[ k ].X )**2
                + ( solid.Vertexes[ r ].Y
                - solid.Vertexes[ k ].Y )**2
                + ( solid.Vertexes[ r ].Z
                - solid.Vertexes[ k ].Z )**2 )
norm_dist = dist / max_dist

if norm_dist >= 0 and norm_dist < 0.1:

```

```

    spacing[ 0 ] = spacing[ 0 ] + 1
elif norm_dist >= 0.1 and norm_dist < 0.2:
    spacing[ 1 ] = spacing[ 1 ] + 1
elif norm_dist >= 0.2 and norm_dist < 0.3:
    spacing[ 2 ] = spacing[ 2 ] + 1
elif norm_dist >= 0.3 and norm_dist < 0.4:
    spacing[ 3 ] = spacing[ 3 ] + 1
elif norm_dist >= 0.4 and norm_dist < 0.5:
    spacing[ 4 ] = spacing[ 4 ] + 1
elif norm_dist >= 0.5 and norm_dist < 0.6:
    spacing[ 5 ] = spacing[ 5 ] + 1
elif norm_dist >= 0.6 and norm_dist < 0.7:
    spacing[ 6 ] = spacing[ 6 ] + 1
elif norm_dist >= 0.7 and norm_dist < 0.8:
    spacing[ 7 ] = spacing[ 7 ] + 1
elif norm_dist >= 0.8 and norm_dist < 0.9:
    spacing[ 8 ] = spacing[ 8 ] + 1
elif norm_dist >= 0.9 and norm_dist <= 1:
    spacing[ 9 ] = spacing[ 9 ] + 1

print spacing
newSpacing = [ float( float(x) / int( V / 20 ) ) for x in spacing ]

print "max_dist:_%f" % max_dist
print "min_dist:_%f" % min_dist

print "E:_%d\nF:_%d\nV:_%d" % ( E, F, V )
Chi = V - E + F
print "Chi:_%d" % Chi
G = ( Chi - 2 ) / -2
print "Genus:_%d" % G
print "CompTime: ",
dur = dt.datetime.now() - t1
print dur
print "RES|%s|%s|%d|%f|%d|%d|%d|%f|%f|%s" % (
    filename, BNAME, G, dur.seconds + dur.microseconds / 1000000.0,
    V, E, F, max_dist, min_dist, newSpacing )
except:
    print "Exception"
    print >> sys.stderr, "(E)_Unexpected_error -3:", sys.exc_info()[0]
    traceback.print_exc( file=sys.stdout )

```

## References

1. MakerBot Industries, LLC. Thingiverse—Digital Designs for Physical Objects. 2008. Available online: <https://www.thingiverse.com> (accessed on 3 December 2017).
2. Facebook, Inc. Welcome to Facebook. 2004. Available online: <https://www.facebook.com> (accessed on 3 December 2017).
3. Twitter, Inc. Twitter. It's What's Happening. 2006. Available online: <http://twitter.com> (accessed on 3 December 2017).

4. Alcock, C.; Chilana, N.H.P.K. Barriers to Using, Customizing, and Printing 3D Designs on Thingiverse. In Proceedings of the 19th International Conference on Supporting Group Work, Sanibel Island, FL, USA, 13–16 November 2016; pp. 195–199.
5. Evans, B. 3D Models from the Cloud. In *Practical 3D Printers*; Apress: Berkeley, CA, USA, 2012; pp. 75–97.
6. Mallon, M. Maker Mania. *Public Serv. Q.* **2014**, *10*, 115–124.
7. O’Hern, M.S.; Kahle, L.R. The Empowered Customer: User-Generated Content and the Future of Marketing. *Glob. Econ. Manag. Rev.* **2013**, *18*, 22–30.
8. Petrov, A.; Pernot, J.P.; Giannini, F.; Falcidieno, B. *Mapping Aesthetic Properties to 3D Free form Shapes through the Use of a Machine Learning Based Framework*; IMATI REPORT Series 16-16; IMATI-CNR: Genova, Italy, 2016.
9. Killi, S.; Kempton, W.L.; Morrison, A. Design issues and orientations in additive manufacturing. *Int. J. Rapid Manuf.* **2015**, *5*, 289–307.
10. Green, C.; Porter, S.; Bingham, G. Product personalisation using personally meaningful data and the creation of new product attributes. *Int. J. Rapid Manuf.* **2017**, *6*, 170–184.
11. Fernandez-Vicente, M.; Canyada, M.; Conejero, A. Identifying limitations for design for manufacturing with desktop FFF 3D printers. *Int. J. Rapid Manuf.* **2015**, *5*, 116–128.
12. Baumann, F.W.; Roller, D. Thingiverse: Review and Analysis of Available Files. *Int. J. Rapid Manuf.* **2017**, in press.
13. Thomas, H. Gcode to x3g Conversion Post Processor. 2013. Available online: <https://github.com/whpthomas/GPX> (accessed on 3 December 2017).
14. MakerBot Industries, LLC. Connected 3D Printing Solutions. 2009. Available online: <https://www.makerbot.com> (accessed on 3 December 2017).
15. Ranellucci, A.; Lenox, J.; Andersen, H.B.; Bubnik, V.; Dandrimont, N.; Hindness, M.; Ledvina, P.; Sapir, Y.; Sheldrake, M.; Yanev, K.; Hodgson, G. Slic3r—G-Code Generator for 3D Printers. 2011. Available online: <http://slic3r.org> (accessed on 3 December 2017).
16. Hrončok, M.; Tomas Sykora, J. CLI and C Library for Processing Triangulated Solid Meshes. 1995. Available online: <https://github.com/admesh/admesh> (accessed on 3 December 2017).
17. Free Software Foundation, Inc. Bash—GNU Project—Free Software Foundation. 2007. Available online: <https://www.gnu.org/software/bash> (accessed on 3 December 2017).

**Sample Availability:** The STL, AMF, GCode, PNG, and GIF files are deposited to <https://zenodo.org>. See <https://dx.doi.org/10.5281/zenodo.1098527> for the complete dataset.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).