

# Polygen3D user manual

## Table of Contents

1. Introduction.....	1
2. Definitions.....	1
3. The graphical user interface (GUI).....	2
3.1 Controls.....	2
3.1.1 Preview box.....	2
3.1.2 Surface list.....	2
3.1.3 Surface properties.....	2
3.1.3.1 Surface name.....	2
3.1.3.2 Surface texture and border color.....	2
3.1.3.3 General appearance properties.....	3
3.1.4 Project tools and properties.....	3
3.1.4.1 Project properties.....	3
3.1.4.2 Project tools.....	3
4. The command line interface (CLI).....	3
4.1 Arguments.....	4
4.1.1 The help argument.....	4
4.1.2 Height.....	4
4.1.3 Width.....	4
4.1.4 Surface.....	4
4.1.4.1 Texture.....	4
4.1.4.2 Border.....	4
4.1.4.3 Count.....	5
4.1.4.4 Edge shifting.....	5
4.1.4.5 Polygon size.....	5
4.1.4.6 Position.....	5

## 1. Introduction

Thank you for downloading Polygen3D! This guide will help you use Polygen3D and gives you detailed instructions about each of Polygen3D's features.

Polygen3D has both a graphical user interface and a command line interface. To see how each interface is launched, look at the corresponding sections of this document.

## 2. Definitions

Polygen3D uses various words and phrases to describe various aspects of its operation:

A surface refers to one set of polygons generated by Polygen3D. The polygons will form either a square, rectangle or parallelogram, which will appear as a three-dimensional surface, hence its name. Multiple surfaces may be created in a single graphic.

## **3. The graphical user interface (GUI)**

The graphical user interface is the most easy to use interface of Polygen3D. To launch the graphical user interface, launch Polygen3D without any arguments. Java will by default open this interface when Polygen3D is launched. Alternately, it can be launched via command line using “java -jar polygen3d.jar”.

### **3.1 Controls**

The graphical user interface provides a range of controls to use in surface manipulation.

#### **3.1.1 Preview box**

The preview box is the left-most control group. This box displays a preview of the graphic as it will appear if exported to an SVG file. Please note that this preview may not always be 100% accurate. Previews will by default render every time a surface setting is changed, however, it may be manually rendered with the Render preview button.

#### **3.1.2 Surface list**

The surface list is the center control group. This list displays all the surfaces currently present in the project. Surfaces may be added by pressing the Add button, and removed by pressing the Remove button. There is no way to re-create a removed surface automatically.

Surfaces may overlap, and the order of the surfaces in the surface list determines which surfaces will be rendered on top of each other. Surfaces near the beginning of the list will render over surfaces near the end of the list. Surfaces may be moved up and down the surface list using the Up and Down buttons.

#### **3.1.3 Surface properties**

The surface properties group is the right-most control group. Using the controls here will let you alter the appearance of each surface in the surface list.

##### **3.1.3.1 Surface name**

The surface name indicates the name of the surface in the surface list. This is only used for readability purposes. It is recommended to use a name that describes the surface, for instance; a surface with a smiley face on it may be named “Smiley surface”.

##### **3.1.3.2 Surface texture and border color**

The surface texture field indicates the path to a raster graphics file used as the surface's texture. Pressing the Browse button allows choosing a file using a file browsing dialog.

The border color field indicates which color should be the border color of all individual polygons in a surface. This must be a color in the #rrggbb format. Color names like “blue” are not allowed. Use the Select button to choose a color using a color dialog.

### **3.1.3.3 General appearance properties**

The general appearance properties represent the four fields “Surface position”, “Polygon count”, “Polygon size” and “Edge shifting”.

The surface position property indicates the position of the surface on the graphic.

The polygon count property indicates the amount of polygons on the X- and Y-axes of the surface.

The polygon size property indicates the size of each polygon in a surface (not the surface itself!).

The edge shifting property indicates how far to the right and bottom each polygon should be shifted on the bottom/right side. X indicates how much the bottom side should be shifted right relative to the top side, and Y indicates how much the right side should be shifted down relative to the left side. Edge shifting is the property that gives surfaces the shape of a parallelogram. Surfaces may be shifted left/up or right/down using negative and positive values, respectively.

### **3.1.4 Project tools and properties**

The project tools and properties panel is the bottom-most control group. The properties here define project-wide settings.

#### **3.1.4.1 Project properties**

The vector size property indicates the overall width and height of the graphic. Everything outside the bounds specified by the vector size property will be cut off from the exported graphic.

#### **3.1.4.2 Project tools**

The project tools panel consists of the controls below the surface list and surface property panels.

The Save project button will open a dialog for saving the Polygen3D project currently being worked on. Saving the project will save all settings and all surfaces to a file for future continuation.

The Open project button will let you open a previously saved project to make changes and continue your work.

The Auto-render check-box indicates whether or not the preview box should update on every surface setting change. If you're using texture files or are rendering many/complex surfaces, it is highly recommended to turn this off to save computer resources.

The Export as SVG button is used to export finished works to a scalable vector format. Scalable vector graphic files can be opened by all modern web browsers, and by many graphic viewers.

The Get CLI arguments button is used to get the command line arguments one could use to generate the exact same graphic as is currently open in the graphical user interface. This serves as an alternative to project files.

The Render preview button will force a re-render of the entire graphic in the preview box.

## **4. The command line interface (CLI)**

The command line interface is useful to generate graphics when a GUI would not be suitable or possible to use. To launch the command line interface, launch Polygen3D with arguments. For

instance, it can be launched via command line using “java -jar polygen3d.jar –help”.

## **4.1 Arguments**

The command line interface is used with a variety of arguments to produce the wanted output.

### **4.1.1 The help argument**

Argument: --help. Aliases: None. Parameters: None. Must be used without other parameters.

The help argument is used to get help about Polygen3D's other arguments, including which arguments are available, their syntax and their meaning.

### **4.1.2 Height**

Argument: --height. Aliases: -h. Parameters: pixels. Required.

The height argument specifies the height of the generated graphic in pixels. This must be a positive integer.

### **4.1.3 Width**

Argument: --width. Aliases: -w. Parameters: pixels. Required.

The width argument specifies the width of the generated graphic in pixels. This must be a positive integer.

### **4.1.4 Surface**

Argument: --surface. Aliases: -s. Parameters: surface arguments. Optional, repeatable.

The surface argument defines a surface in the final graphic. Used with surface parameters. The surface argument may be repeated: Additional surfaces will be rendered after/on top of previous surfaces.

#### **4.1.4.1 Texture**

Argument: --texture. Aliases: -t. Parameters: path. Optional.

The texture argument is used together with the surface argument to define a surface's texture. A relative path will be rooted in the directory in which Polygen3D is located.

#### **4.1.4.2 Border**

Argument: --border. Aliases: -b. Parameters: color. Optional.

The border argument is used together with the surface argument to define a surface's inner and outer border color. The color must be in #rrggbb format, #rgb format or the color's common name. Please note that number signs must be escaped using a backslash: \#rrggbb.

#### **4.1.4.3 Count**

Argument: --xcount. Aliases: -xc. Parameters: count. Required.

Argument: --ycount. Aliases: -yc. Parameters: count. Required.

The count arguments are used to define the amount of polygons on the X- and Y-axes of the surface. Both must be positive integers.

#### **4.1.4.4 Edge shifting**

Argument: --xshift. Aliases: -xi. Parameters: pixels. Required.

Argument: --yshift. Aliases: -yi. Parameters: pixels. Required.

The edge shifting arguments are used to define the amount of pixels the bottom corners should be shifted right relative to the top corners (xshift) and the amount of pixels the right corners should be shifted down relative to the left corners (yshift) in each polygon. Setting these parameters to non-zero values will rotate and/or shape the surface like a parallelogram. Both must be positive or negative numbers. Numbers with decimals are allowed.

#### **4.1.4.5 Polygon size**

Argument: --xnext. Aliases: -xn. Parameters: pixels. Required.

Argument: --ynext. Aliases: -yn. Parameters: pixels. Required.

The polygon size arguments are used to set the width (xnext) and height (ynext) of each polygon on the surface. Sizes are calculated as the vertical distance from the top-left corner to the bottom-left corner (xnext) and the horizontal distance from the top-left corner to the top-right corner (ynext). Both parameters must be numbers (decimals allowed).

#### **4.1.4.6 Position**

Argument: --xstart. Aliases: -xs. Parameters: xpos. Required.

Argument: --ystart. Aliases: -ys. Parameters: ypos. Required.

The position arguments are used to set the location of the surface on the graphic. Both parameters must be positive numbers (decimals are allowed).