

# Mend User Manual

Stanley Hailey

# Table of Contents

1. Description .....	1
2. Feature List .....	1
3. Application Overview .....	1
3.1. Usage Details .....	1
3.2. Editor Controls .....	2
3.3. Shader Settings .....	4
4. Hotkeys .....	4
5. Contact .....	5

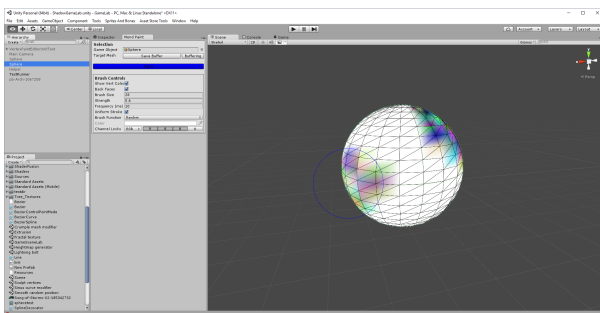
# 1. Description

Mend Vertex Painter is a simple but flexible Unity extension made for users needing to encode vertex color data or create stunningly artistic polygons within Unity.

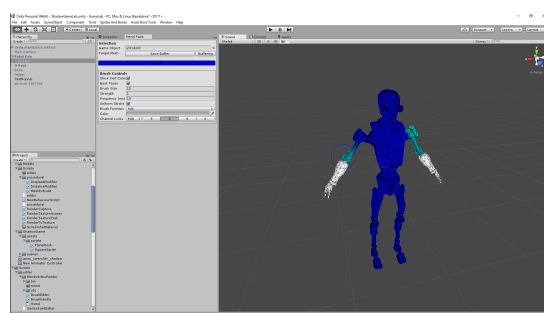
## 2. Feature List

Features supported in **Mend** v1.0.

- Display and edit vertex colors
- Paint in perspective or orthographic view
- Control brush size, strength and frequency
- Lock individual RGBA or HSBA channels
- Save edits to a file or make persistent edits directly to a mesh
- Brush functions include:
  - Blend Color Value
  - Add Color
  - Subtract Color
  - Random Color
  - Average Vertex Colors
- Paint options include:
  - Brushing front faces only
  - Uniform brushing for single strokes



Screenshot 1



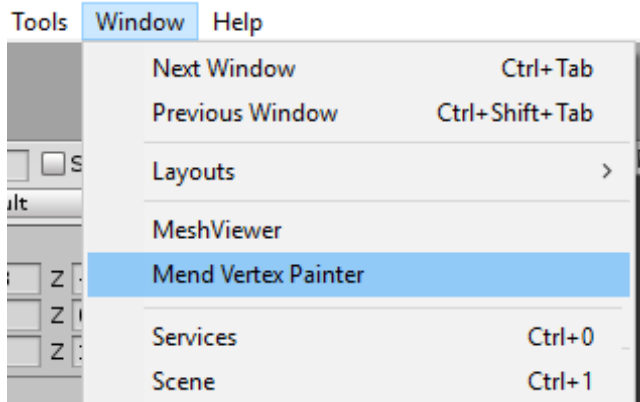
Screenshot 2

## 3. Application Overview

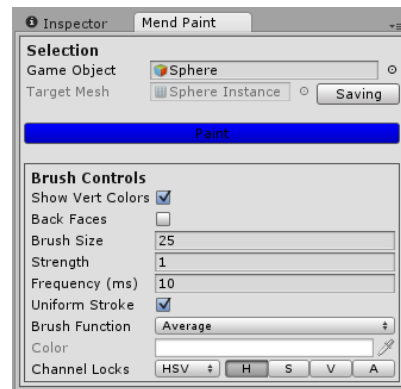
### 3.1. Usage Details

**Mend** can be accessed from the Unity menu by clicking *Window > Mend Vertex Painter*. Once open **Mend** allows you to edit vertex colors of the selected Game Object shown in the *Game Object* field. The current selection must contain a Mesh for vertex color painting. Activate the Paint button to

start painting. If *Show Vertex Colors* is enabled **Mend** switches the shader to '*Unlit/Vertex Colors*' to display vertex color edits while painting .



Menu location



Mend editor window

While painting **Mend** prevents unintentional manipulation or object reselection. Deactivate the Paint button to allow transform manipulation on the object and restore the mesh shader.

Vertex colors edits can be saved either directly to the Mesh or to a buffer which must be saved to file. Use the *Mesh Target* button to toggle between saving or buffering. Use *Save Buffer* to save a copy of the mesh to a chosen file.

## 3.2. Editor Controls

### Game Object Selection

Display or set the current selected game object.

Type: `UnityEngine.GameObject`

### Mesh Target Selection

Displays the mesh to which edits will be saved. Is set to the game object's mesh if it contains has a MeshFilter or SkinnedMeshRenderer.

Type: `UnityEngine.Mesh`

### Mesh Target State

Set the edit saving mode to saving or buffering.

Type: `Enum { Saving, Buffering }`  
Default: `Saving`

## Show Vertex Colors

Set the option to enable or show or hide the vertex color shader.

Type: bool  
Default: true

## Edit Back Faces

Set the option to enable or disable editing back-faces.

Type: bool  
Default: true

## Brush Size

Set the size of the brush. The value represents the size measured as a percentage of the screen.

Type: float  
Default: 10.0  
Range: [1.0, 50.0]

## Brush Strength

Set the blend strength of the brush. The amount to lerp between the selected color and the current vertex color. A value of 1.0 maximizes the influence of the brushed color.

Type: float  
Default: 1.0f  
Range: [0.01f, 1.0f]

## Stroke Frequency

Set the brush application frequency in milliseconds while the left mouse button is down.

Type: int  
Default: 10  
Range: [1, 4000]

## Uniform Stroke

Vertices will only be affected once per stroke while enabled.

Type: bool  
Default: true

### Brush Functions

Set the function to use when applying color to vertices.

Type: Enum { Value, Add, Subtract, Average, Random }  
Default: Value

### Brush Color

Set the color apply to vertices.

Type: UnityEngine.Color32  
Default: Color32 (255, 255, 255, 255)

### Channel Lock Mode

Set the mode for channel locks between *red*, *blue*, *green*, *alpha* and *hue*, *saturation*, *brightness/value*, *alpha*.

Type: Enum { RGBA, HSBA }  
Default: RGBA

### Channel Locks

Set the channels lock state while editing. Channels with a pressed button indicates that channel can be edited and has a channel lock value of *false* .

Type: bool[4]  
Default: [false, false, false, false]

## 3.3. Shader Settings

Mend will apply a shader on the mesh while in paint mode to display vertex colors. The wireframe color is exposed as an inspector property for 'Unlit/Vertex Colors' shader. Locate it in the inspector under the Material component for the Mesh being painted.

## 4. Hotkeys

Available hotkeys are visible in editor tooltips.

Shortcut	Action
Alt+1	Toggle Paint on or off
Alt+2	Toggle display of vert color shader
Alt+3	Toggle backFace editing on or off
Alt+4	Toggle mode between buffering and saving
Alt+5	Save mesh when buffering
Shift+ScrollWheel	Resize paint brush
Alt+ScrollWheel	Rescale paint strength

## 5. Contact

For questions or comments please contact Stanley Hailey (Lee) by email.

Email: [slhaile01@caffeine-machine.com](mailto:slhaile01@caffeine-machine.com)

Support Website: <http://caffeine-machine.com/contact>