
Flexible Color Picker, free asset for Unity

Ward Dehairs
<http://wardddev.com>

1 Basic use guide

To integrate the color picker into your Unity project, simply add the provided prefab 'FlexibleColorPicker' to a canvas. Once added, you can access the output color of the FlexibleColorPicker script via the parameter 'color'. An extremely simple use case would for example be the following:

```
public FlexibleColorPicker fcp; public Material material;  
  
private void Update() material.color = fcp.color;
```

1.1 Customization

All the 'picking images' of the color picker can be turned on or off and resized at will. 1 dimensional pickers, e.g. sliders will automatically orient themselves along the longest axis. All picker images are connected via a public array variable to the main script, this array should never be modified. It is not recommended to delete elements of the color picker, but rather turn the gameobjects off.

1.1.1 Naming

Picker images refer to 1D or 2D sliders that let the user change the selected colors. Color values are denoted by their starting letter: Red, Green, Blue, Hue, Saturation, Value and Alpha.

1.1.2 Markers

Markers pointing to the current value of a picker image are recognized by their name. They should contain the words 'marker' and 'hor' or 'ver' to indicate their function. 1 dimensional pickers have a horizontal and vertical marker included by default. These are automatically turned on or off based on the detected direction. The 2 dimensional picker has only one marker. The markers can be freely customized, as long as the point they aim towards remains in the middle of the images and their names still contain the right keywords.

1.1.3 Main Picking Mode

Main picking mode can be set by the user or set by the developer ahead of time. This changes which color values are shown in the 2D picking image. The letters denote Hue, Saturation and Value.

1.1.4 Lightweight Mode

Lightweight mode will prevent the color picker from changing the picking textures when the user changes their selection. This makes the picker faster at the cost of less interactivity. When using lightweight mode it is recommended to turn off the main 2D picker image and the options to change the main picking mode. This is because the 2D picker tends to be confusing when it does not change dynamically.

2 Scripting details

2.1 BufferedColor

The Color Picker works internally with a `BufferedColor` object, this color can retain hue and saturation values even when these are singular for its color. This makes sure that these values do not automatically reset if the user select a singular color: e.g. pure gray has no valid hue value, which defaults to 0 indicating red. If the user desaturates blue it would suddenly jump to being red once completely desaturated.

2.2 Public access

- **PickerImages**; Should always contain the 10 types of picking images in order!
- **HexInput**; Connects to the hexadecimal color input field.
- **Dropdown**; Connects to the main picking mode selection dropdown.
- **mode**; Current main picking mode for the 2D picking image.
- **startingColor**; Color value set upon `Start()`
- **lightweightMode**; Should textures not change dynamically for performance reasons.
- **SetPointerFocus**; Set a specific picker image as the focused image for subsequent updates.
- **PointerUpdate**; Set new color value via a picking image.
- **TypeHex**; Set new color value via hex input.
- **FinishTypeHex**; idem, for finishing input.
- **ChangeMode**; Change main 2D picking mode (via button)

2.3 Public static functions

The Color Picker provides a few free functions that may be useful in further scripting: functions for sanitizing hex strings and for converting between rgb and hsv color formats.

2.4 Internal workings

These have been documented in the script itself where necessary, look for full-line comments for the structure.

2.5 Performance / optimization

The Flexible Color Picker was developed with clean and readable code in mind, in preference over performance. Optimizations may still be worthwhile when generating picking textures.