HIGHLIGHT PLUS 2D

STANDARD &
LWRP/HDRP Compatible

Kronnect

Contents

Introduction

Thank you for purchasing!
Highlight Plus 2D is a simple yet powerful package for adding outline, glow and other effects to your sprites.

We hope you find the asset easy and fun to use. Feel free to contact us for any enquiry.
Visit our Support Forum on https://kronnect.me for help and access to the latest beta releases.

Kronnect Games
Email: contact@kronnect.me
Kronnect Support Forum: http://www.kronnect.me

Quick Start and Demo Scene

1. Import the asset into your project or create an empty project.
2. Go to Demo folder and run the demo scene to quickly test the asset effects.
3. Examine the code behind the script attached to the Demo game object.

The Demo scene contains two examples of how to render the effects (one or two ways can be used):

- *A HighlightPlus2DManager instance (the prefab) which detects when the pointer is over a sprite and triggers the effects. The manager is a single component that you can add from the top menu GameObject -> Effects -> Highlight Plus 2D -> Create Manager.*

- *A Potion sprite which has a Highlight Plus Effect and Highlight Plus Trigger component. The Highlight Plus Effect contains the effects settings while the Highlight Plus Trigger component is responsible for detecting when the pointer enters the sprite and enable those effects using the settings from the Highlight Plus Effect component.*

How to use the asset in your project

Option 1: Highlighting/customizing a SINGLE sprite

- Add HighlightEffect.cs script to any sprite. Customize the appearance options.
- Optionally add HighlightTrigger.cs script to the sprite GameObject. It will activate highlight on the sprite when mouse pass over it. Note that a collider must be present on the sprite. Note: adding a HighlightTrigger.cs script to a sprite will automatically add a HighlightEffect component.

Option 2: Highlighting/customizing ANY sprite automatically

- Select top menu GameObject -> Effects -> Highlight Plus 2D -> Create Manager.
- Customize behaviour of Highlight Manager 2D. If a sprite already has a HighlightEffect component, it will use those settings instead.

Ignoring specific sprites from highlighting

Add a HighlightEffect component to the sprite and activate the "Ignore" checkbox under Highlight section.

Exclusive and grouped outline effects

By default, Highlight Plus combines outline of overlapping sprites. If you want certain sprites to preserve their own outline you have two options:
  a) Enable the "Exclusive" toggle under the Outline section. That will make the outline to show independently for that sprite.
  b) Add a HighlightGroup2D component to the sprites in a group and assign a group number (ie. "1"). Group 0 does not do anything. The outlines of that group will not be combined with other sprites, only among sprites of the same group.

Advanced Topics and Notes

Sorting layers

Highlight Plus 2D Effects are always rendered into Sorting Layer 0. If you have sprites that use a greater sorting layer, you may need to change them, so they use a lower sorting layer.

Using scripting to add effects

Use GetComponent<HighlightEffect2D>() to get a reference to the component of your sprite.
Most properties shown in the inspector can be accessed through code, for example:

```
using HighlightPlus2D;
…

HighlightEffect2D effect = mySprite.GetComponetn<HighlightEffect2D>();
effect.outline = true;
effect.outlineColor = Color.blue;
effect.Refresh();
```

Hit FX!

Call HitFX() method to execute a hit effect:

```
using HighlightPlus2D;
…

HighlightEffect2D effect = mySprite.GetComponetn<HighlightEffect2D>();
effect.HitFX(color, fadeOutDuration, initialIntensity);
```

Events / reacting to selection

When an object is highlighted the HighlightStart and HighlightEnd messages are invoked on all scripts attached to the sprite. You can also use the OnObjectHighlightStart/OnObjectHiglhightEvent to react to highlight events and cancel the event depending on the situation.
Check PotionHighlightEventSample.cs script in the demo scene.

License of the icons

All icons in the demo scene are Public Domain.