Negative Shader

Asset Store Link

PLEASE LEAVE A REVIEW OR RATE THE PACKAGE IF YOU FIND IT USEFUL! Enjoy! :**)

Contact

Questions, suggestions, help needed?
Contact me at: Email: jgarza9788@gmail.com
Cell: 1-818-251-0647
Contact Info: justingarza.net/contact

Description/Features

Awesome Negative Shader

- Negative UI objects
- Negative UI objects (in World Space)
- Negative Particles
- Negative Circle!

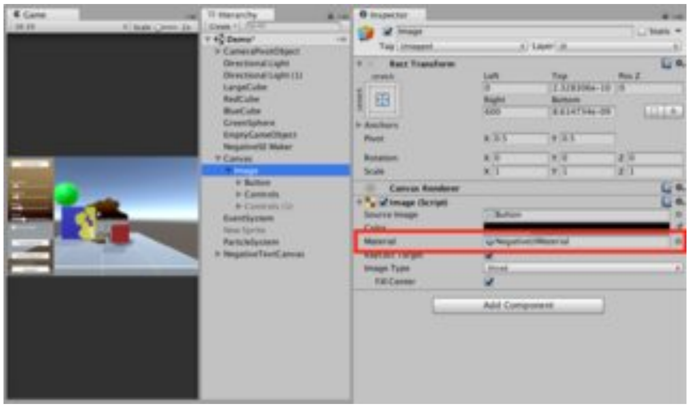
Terms of Use

You are free to add this asset to any game you'd like However,
please put my name in the credits, or in the special thanks section. :)
please do not re-distribute.

Table of Contents

Negative UI Objects

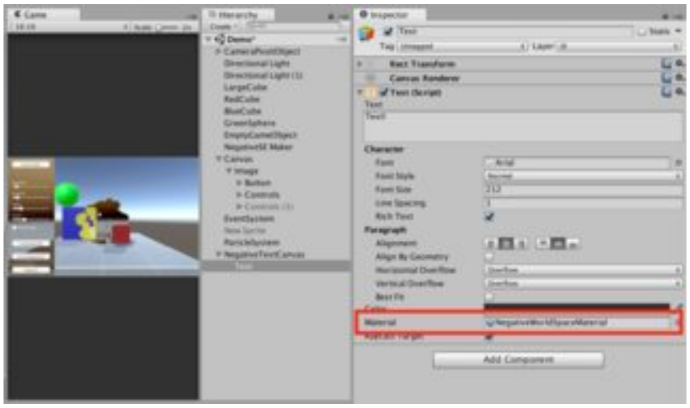
To create a Negative UI object just set the material of the image to the NegativeUIMaterial.mat. (or create a new material using the NegativeUI.shader)



Material: NegativeUIMaterial

Negative UI Objects (in World Space)

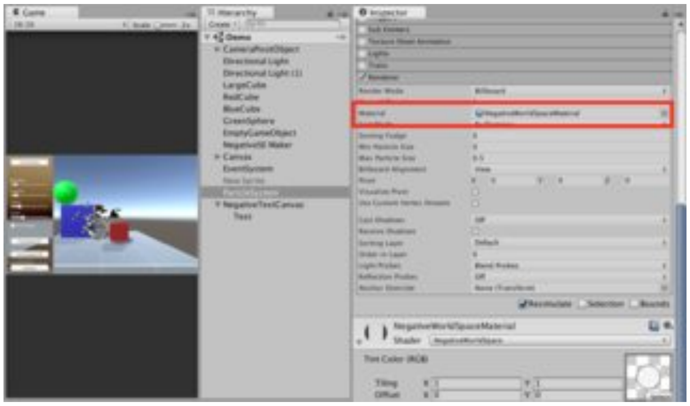
To create a Negative UI object (in World Space) just set the material of the image to the NegativeWorldSpaceMaterial.mat. (or create a new material using the NegativeWorldSpace.shader)



Material: NegativeWorldSpaceMaterial

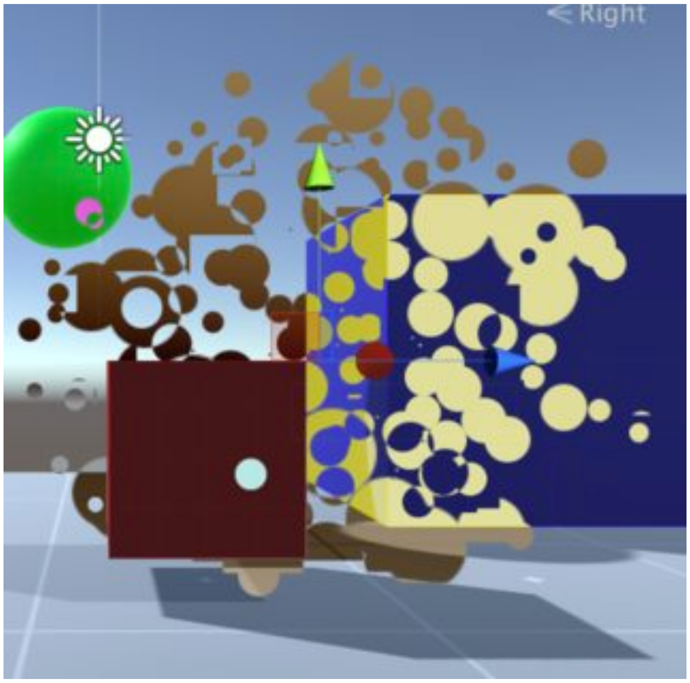
Negative Particles

you can also use the NegativeWorldSpaceMaterial.mat to create Negative Particles.



Render Mode Billboard
Material NegativeWorldSpaceMaterial
Sorting Fudge 0
Min Particle Size 0
Max Particle Size 0.5
Billboard Alignment View
Pivot X 0 Y 0 Z 0
Visualize Pivot
Use Custom Vertex Streams
Cast Shadows Off
Receive Shadows
Sorting Layer Default
Order in Layer 0
Light Probes Blend Probes
Reflection Probes Off
Anchor Override None (Transform)
NegativeWorldSpaceMaterial
Shader NegativeWorldSpace
Tint Color (RGB)
Tiling X 1 Y 1
Offset X 0 Y 0

Note:** Multiple negative particles using this material might not render correctly. Below is an image of how multiple negative particle systems might render.



Negative Circle

Just add this line of code when you want a NegativeCircle to occur on the screen.

```C#
//Display NegativeCircle
NegativeCircle.Get().StartIt([Position], [MaxRadius],[Delay],[Speed1], [Speed2]);

//Display NegativeCircle (if you are passing in a screen position)
NegativeCircle.Get().StartIt([Position],true, [MaxRadius],[Delay],[Speed1], [Speed2]);
//Display NegativeCircle (if you are passing in a GameObject)
NegativeCircle.Get().StartIt([Target], [MaxRadius],[Delay],[Speed1], [Speed2]);
```

Position: This is the Position in Vector2 for a screen position, or Vector3 for a world position, or a GameObject

IsScreenPosition: (Optional) Used to convey if the position is a screen position.

MaxRadius: This is how large the NegativeCircle can get before it stops growing.

Delay: The delay between the first and second circle.

Speed1: The speed of the first circle.

Speed2: The speed of the second circle.

Here is a few examples of code.

```C#
//Display NegativeCircle at the mouse’s position
NegativeCircle.Get().StartIt(Input.mousePosition,true,0.25f,0.5f,1f,1.5f);
//Display NegativeCircle at the a gameObject’s Position
NegativeCircle.Get().StartIt(gameObject.transform.position,0.25f,0.5f,1f,1.5f);
//Display NegativeCircle at the a gameObject’s Position, and follow the gameObject.
NegativeCircle.Get().StartIt(gameObject,0.125f,0.5f, 1.0f,1.5f);
```

An Alternative way to use the NegativeCircle is to pass in a different set of values.

```C#
//Display NegativeCircle
NegativeCircle.Get().StartIt([Position], [MaxRadius],[WaveSize],[Speed]);

//Display NegativeCircle (if you are passing in a screen position)
NegativeCircle.Get().StartIt([Position],true, [MaxRadius],[WaveSize],[Speed]);

//Display NegativeCircle (if you are passing in a GameObject)
NegativeCircle.Get().StartIt([Target], [MaxRadius],[WaveSize],[Speed]);
```

Position: This is the Position in Vector2 for a screen position, or Vector3 for a world position, or a GameObject

IsScreenPosition:** (Optional) Used to convey if the position is a screen position.

MaxRadius: This is how large the NegativeCircle can get before it stops growing.

WaveSize: This is the maximum length between the first and second circle. Similar to the Delay value.

Speed: The speed of the both circles.

Here is a few examples of code.

```C#
//Display NegativeCircle at the mouse’s position
NegativeCircle.Get().StartIt(Input.mousePosition,true,0.25f,0.5f,1f);
//Display NegativeCircle at the a gameObject’s Position
NegativeCircle.Get().StartIt(gameObject.transform.position,0.25f,0.5f,1f);
//Display NegativeCircle at the a gameObject’s Position, and follow the gameObject.
NegativeCircle.Get().StartIt(gameObject,0.125f,0.5f, 1.0f);
```

Read NegativeCircle.cs for more details.

Demo!

Now try the Demo!

[Click Here for Demo]