

Image Overflow

rendering a sprite
outside of Rect Transform

Introduction

I strongly recommend reading this guide before using the asset.

1 Technical details:

- ImageOverflow is not a separate Image component or a replacement for the Image component. It is added to an existing Image component on its GameObject, serving as an auxiliary tool.
- The **Slice** and **PreserveAspectRatio** functions are currently **not supported**.
- The area extending beyond the RectTransform will not affect the position of the asset inside GridLayout, HorizontalLayout groups, or other AutoLayouts.
- When resizing or scaling the image, the sprite margins you set will remain the same.
- ImageOverflow is guaranteed to be **compatible with UnityEngine.UI.Image** component.
- ImageOverflow inherits from the "BaseMeshEffect" class.

2 If you encounter any errors while working with the asset, please write me about it at provided contacts or in the Issues section on GitHub.

I typically respond quickly to messages, offer assistance on an individual basis, and address any identified errors in the upcoming updates.

Discord Server: <https://discord.com/invite/ZsnDffV5eE>

Telegram Group: https://t.me/da_assets_publisher

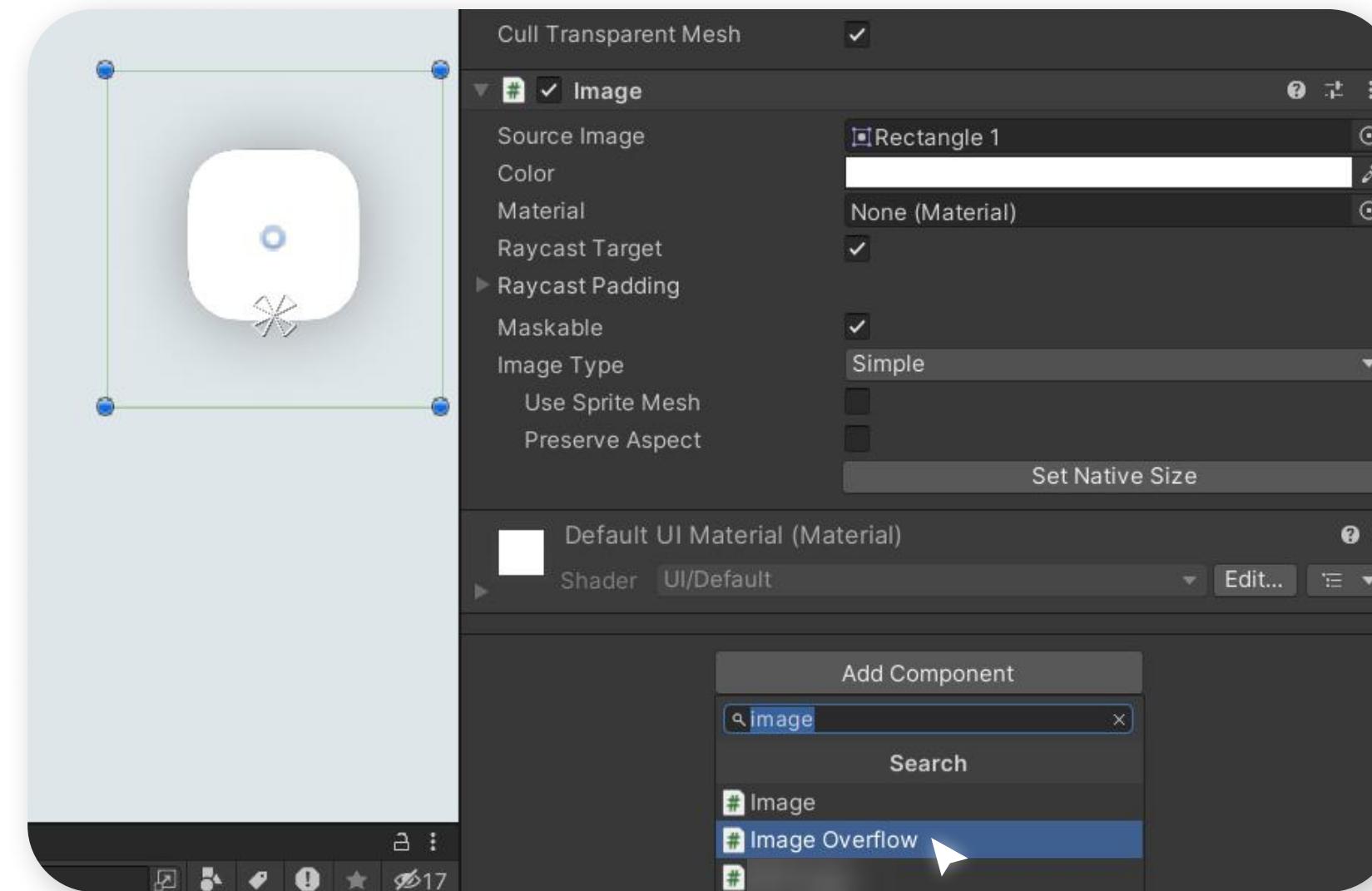
Email Support: da.assets.publisher@gmail.com

Website: <https://da-assets.github.io/site/>

Usage

1

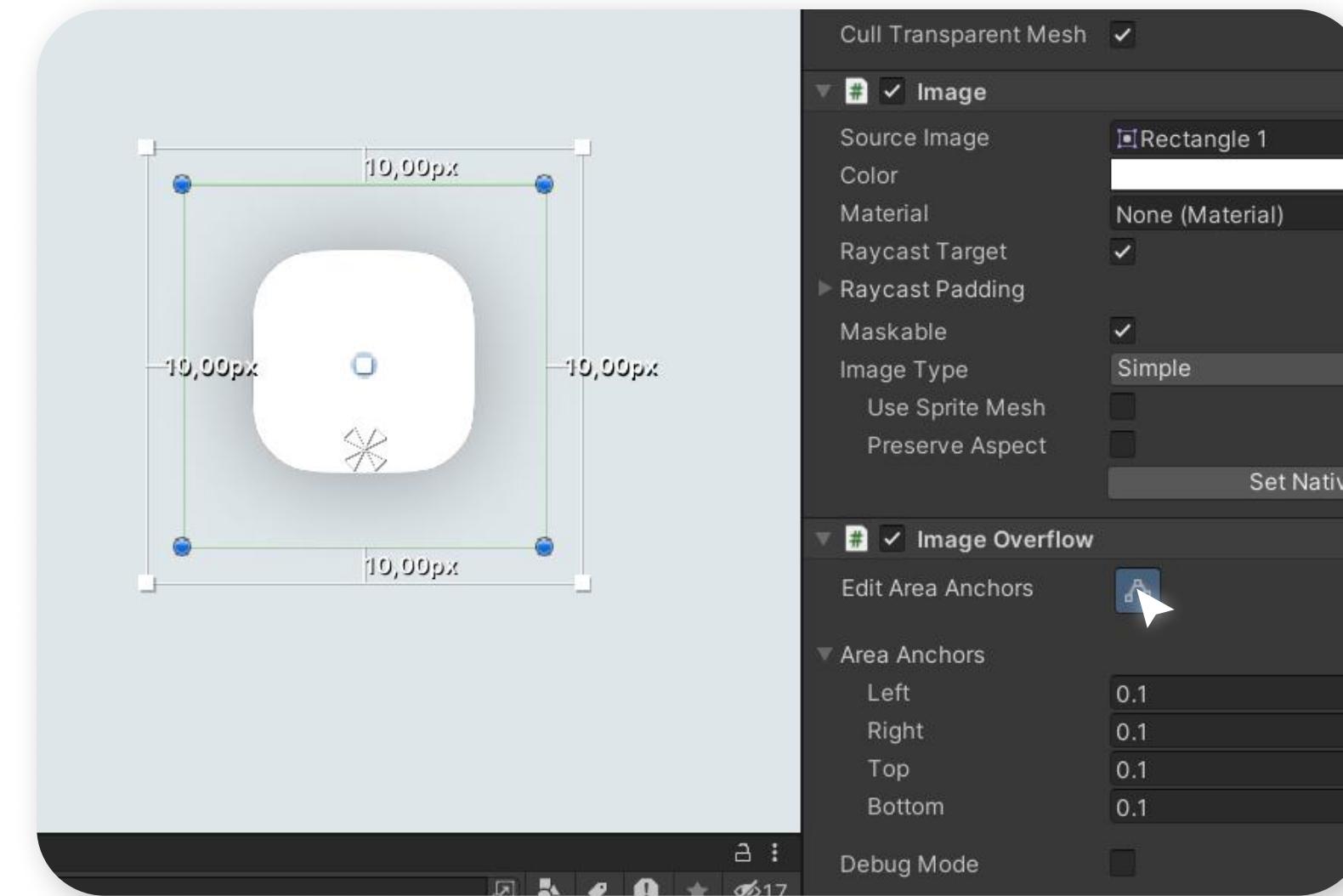
Add the "ImageOverflow" component alongside your Image component.
Please note that if you haven't assigned a sprite to the Image component, the "ImageOverflow" script will not work.



Usage

2

- To adjust the anchors, click the button that enables the anchor editing mode.
To disable the anchor editing mode, you can click this button again.



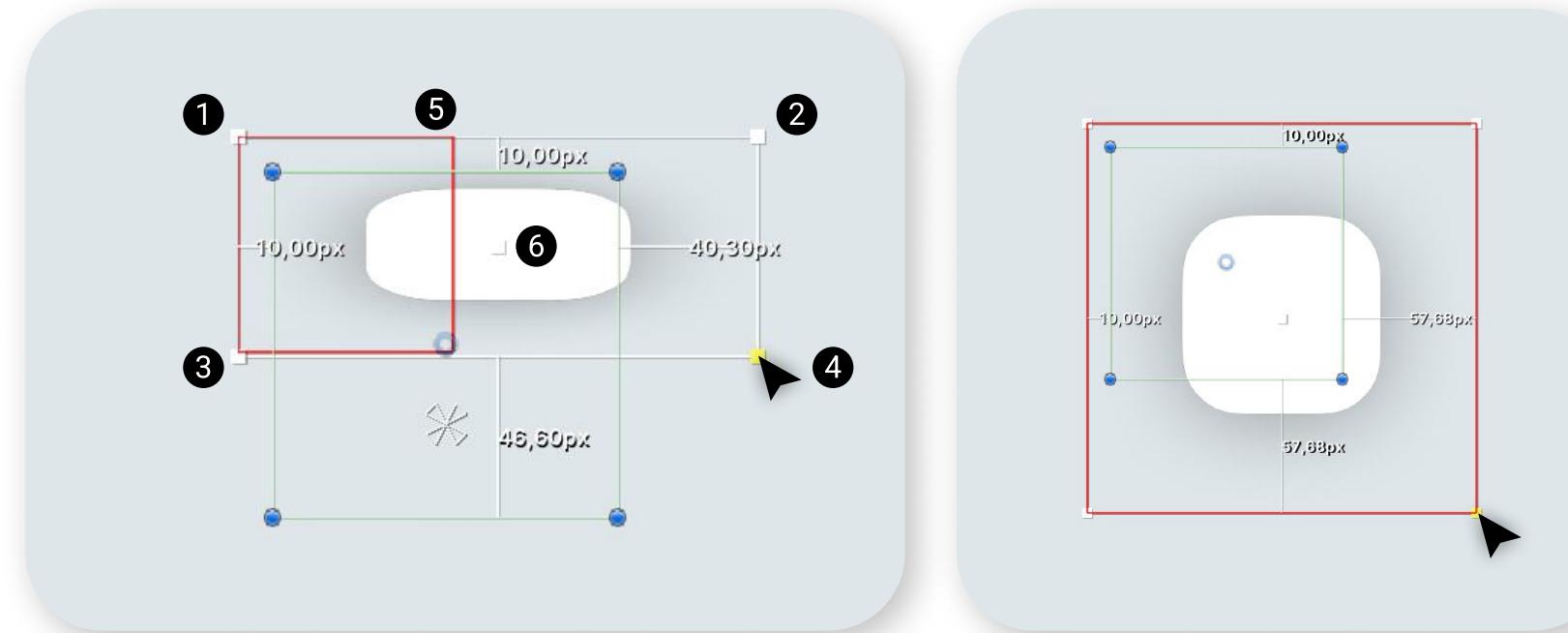
Usage

3

1-4 - gizmo squares you can use to adjust the margins.

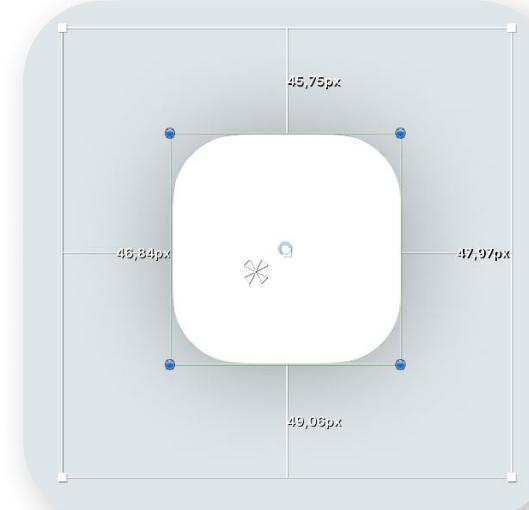
5 - the red outline shows what the object will look like if you adjust the margins while maintaining the sprite's proportions. To adjust the margins this way, move the gizmo squares while holding down the Shift key.

6 - the center square allows you to drag the object while maintaining its current size. Before moving the center square, adjust the margins on the other sides so that the square does not overlap with the round gizmo of the RectTransform; otherwise, it will be difficult to grab.



4

Example of adjusted margins. As we can see, the shadow, which is also part of the sprite, extends beyond the RectTransform boundaries.



Usage

3

Example of using 'ImageOverflow' for Image components inside a HorizontalLayoutGroup.
As we can see, the shadow extends beyond the RectTransform and does not affect the spacing between
GameObjects inside the HorizontalLayoutGroup.

