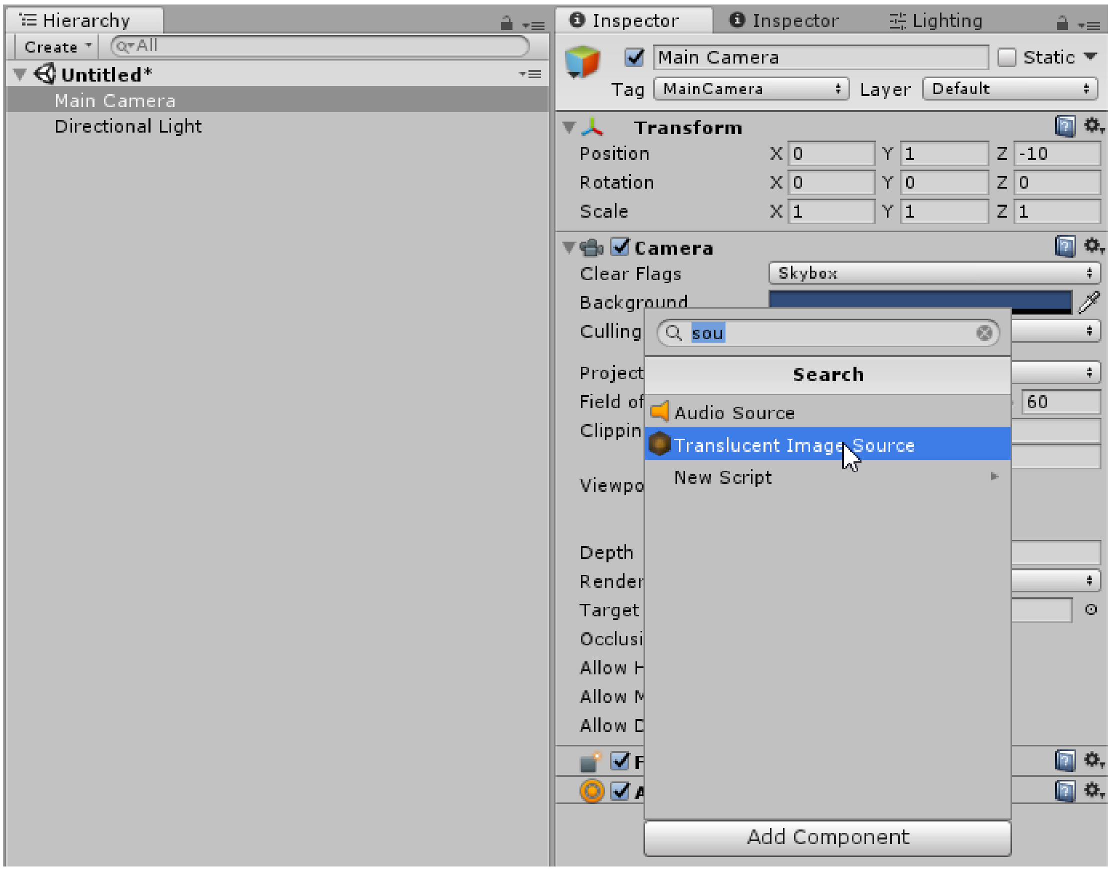
Table of Contents

Getting Started

1. Add `Translucent Image Source` to your main camera.



Main Camera

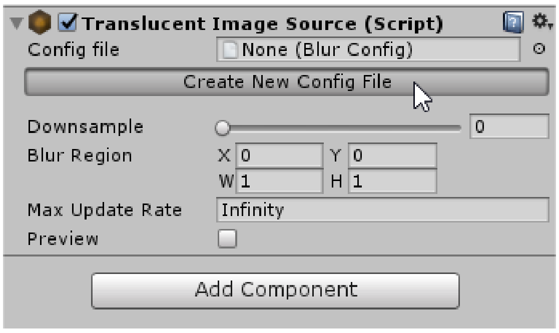
Tag MainCamera Layer Default

	X	Y	Z
Position	0	1	-10
Rotation	0	0	0
Scale	1	1	1

Clear Flags: Skybox

Field of: 60

2. Create a ***Blur Config*** asset, or assign an existing one.



Config file: None (Blur Config)

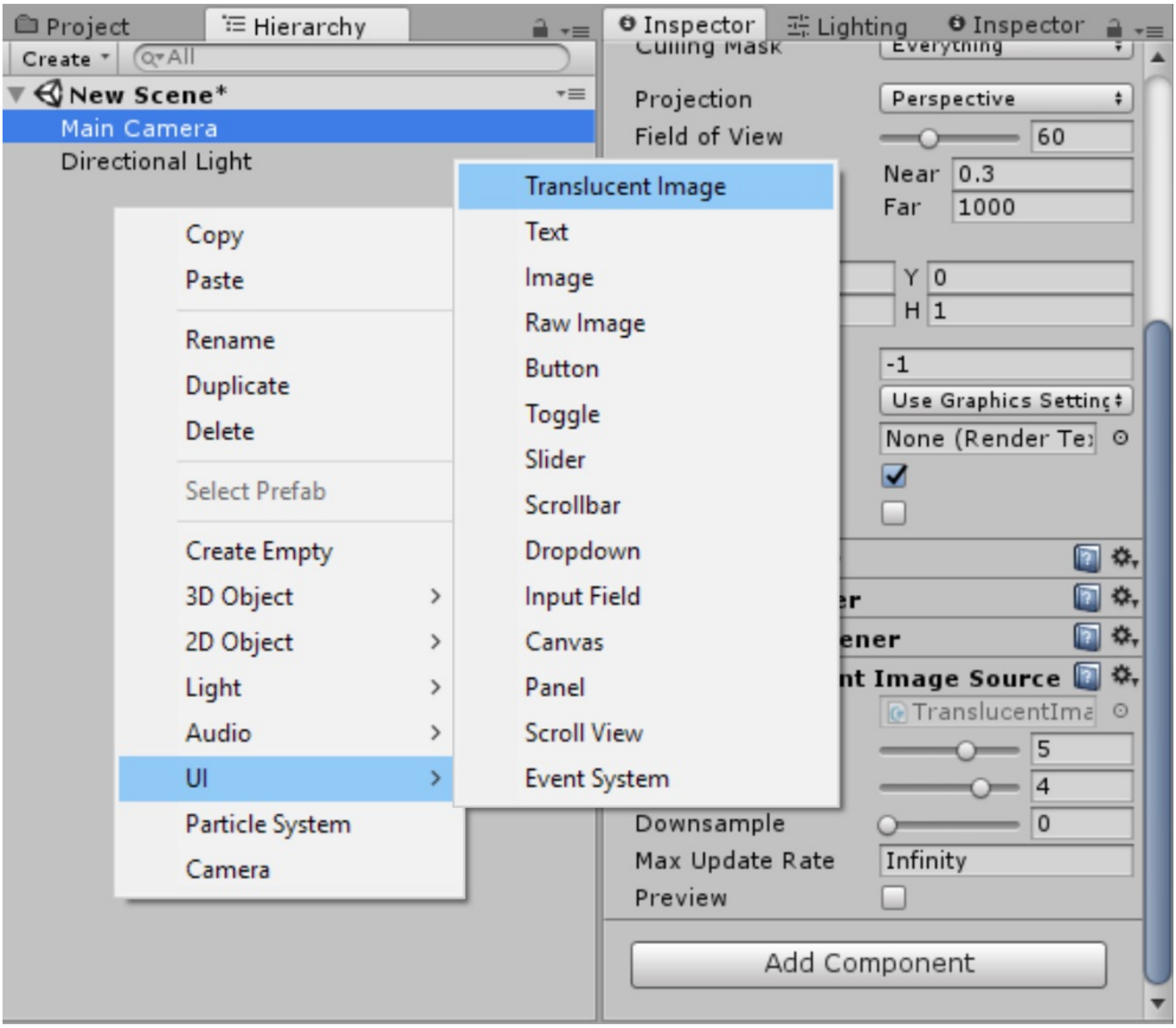
Downsample: 0

Blur Region: X 0 Y 0 W 1 H 1

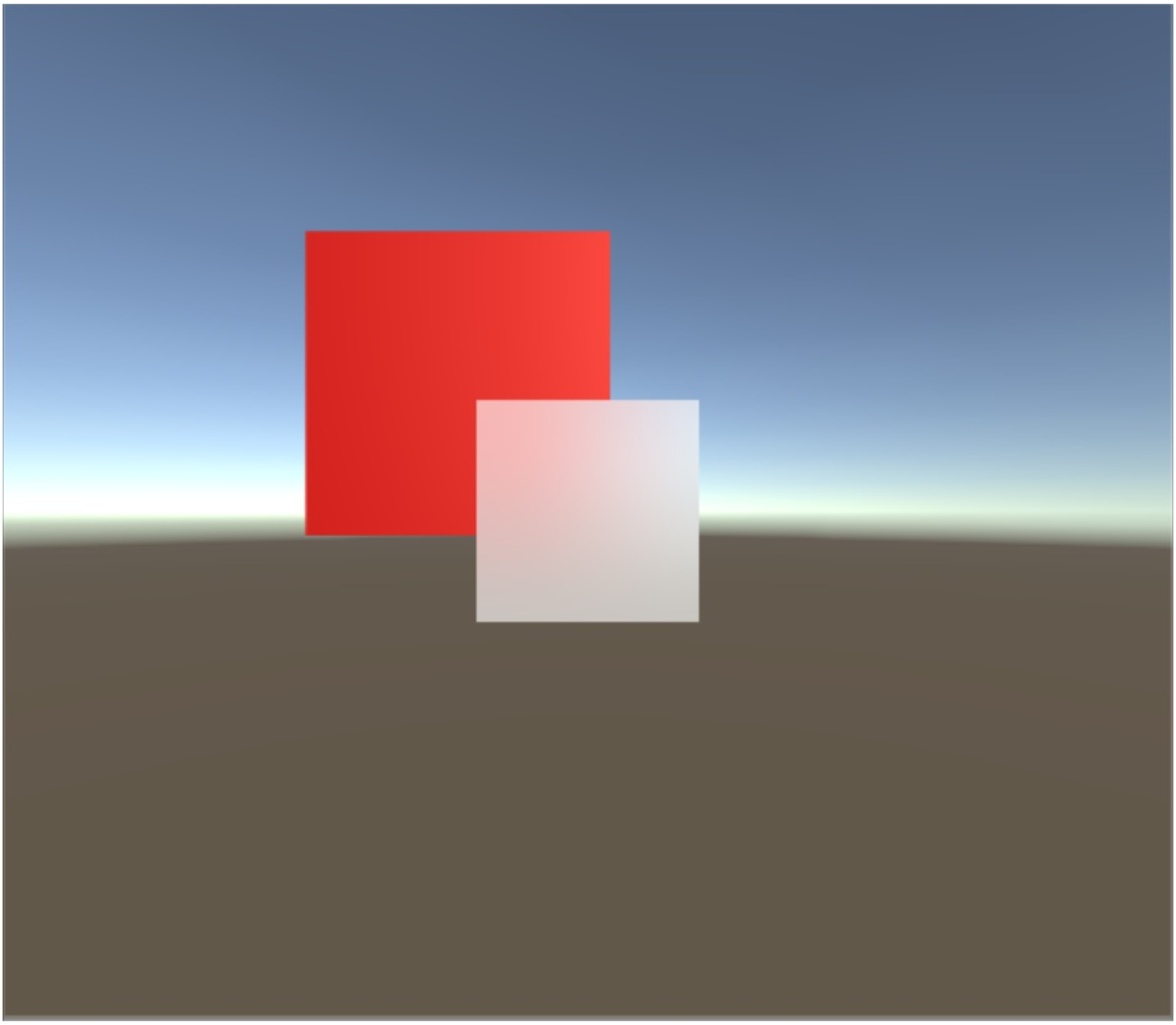
Max Update Rate: Infinity

Preview: □

3. Create `UI -> Translucent Image` as you would with normal UI Image.


Project
Hierarchy
Create
All
New Scene*
Main Camera
Directional Light
Copy
Paste
Rename
Duplicate
Delete
Select Prefab
Create Empty
3D Object
2D Object
Light
Audio
UI
Particle System
Camera
Translucent Image
Text
Image
Raw Image
Button
Toggle
Slider
Scrollbar
Dropdown
Input Field
Canvas
Panel
Scroll View
Event System
Inspector
Lighting
Inspector
Culling Mask
Everything
Projection
Perspective
Field of View
60
Near
0.3
Far
1000
Y
0
H
1
-1
Use Graphics Setting
None (Render Te
er
ener
nt Image Source
TranslucentIma
5
4
Downsample
0
Max Update Rate
Infinity
Preview
Add Component

4. That's it!



> [!NOTE]
> Sometime the effect does not shown up immediately. If that happen, just switch to play mode. The effect will continue to show even when exit play mode.

Customize

□ **Note**

This package was designed to be scalable. All properies that was said below to affect performance actually do so very little

There are 2 components that form the effect, both with their own parameter that affect the look of the effect:

Translucent Image Source

This component offers two modes of controlling the amount of blur: *Simple* and *Advanced*:

- ***Simple***:
  - **Strength**. Using this single property, you can (kinda) smoothly change the blur amount at runtime.
- ***Advanced***:
  - **Size**: How much blurriness you want. Doesn't affect performance, but will look bad if the number too big. Also reduce flickering.
  - **Iteration**: Increase blur quality and blurriness when it is increased.

There are also other properties that are independant of mode:

- **Max Depth**: Increase this property will:
  - Increase flickering when background moving
  - Increase blur level
  - Improve performance
- **Downsample**: Decrease the resolution before processing to increase performance. Side effect include increase blurriness and flickering.
- **Blur Region**: Select the region of the screen to blur. If your UI does not span the entire screen, it might be a good idea to limit this to only the part that you use to increase performance and reduce power usage.

  > □ **Tip**
  >
  > It easier if you tune the `x` and `y` value before `w` and `h`

- **Max Update Rate**: How many time the effect update itself per second. Use this property to increase performance and decrease power usage. Set to 0 to pause, this can reduce power usage/ prevent overheat when you don't need dynamically updating background - like in a pause menu for example.
- **Preview**: preview the effect in full-screen without creating a Translucent Image

Translucent Image

- **Source Image**: The sprite to use for this image.
- **Material**: Multiple Translucent Image using the same material can only have different color, but they can batch dynamically to only take one draw call.

  > □ **Warning**
  >
  > Material used here must use the shader UI/TranslucentImage

- **Color, Raycast Target, Image Type**: same as built-in Image.
- **Source**: Translucent Image Source component. This is where the image gets the blurred screen. It will automatically being set to the first one found, so you should make sure there one in your scene before creating any Translucent Image. You can always override this to change which camera will be blurred.

- **Vibrancy**: How colorful you want the background to be, 0 mean black and white, negative value will invert the color. This is great for enhancing the detail behind the image, or making death screen.
- **Brightness**: Brighten or darken the background.
- **Flatten**: Make your Translucent Image more contrast-y against the background. Useful when you can't predict the color of the background.

LWRP

Requirements

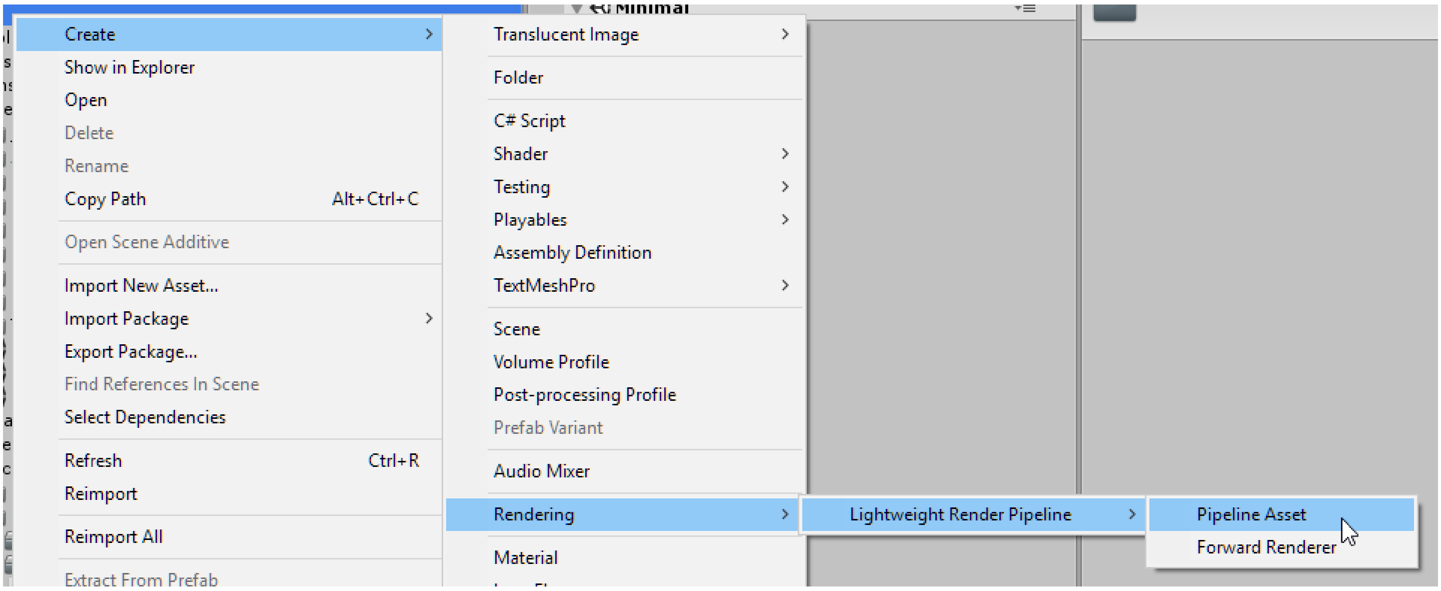
LWRP support was tested with LWRP version `5.7.2`. LWRP on Unity 2018 is not supported, as it is still a preview package.

The files required for LWRP support can be found by importing the unity package at: `Assets/Le Tai's Asset/TranslucentImage/LWRP support`. They are not included by default since they would cause errors for projects without the LWRP package.

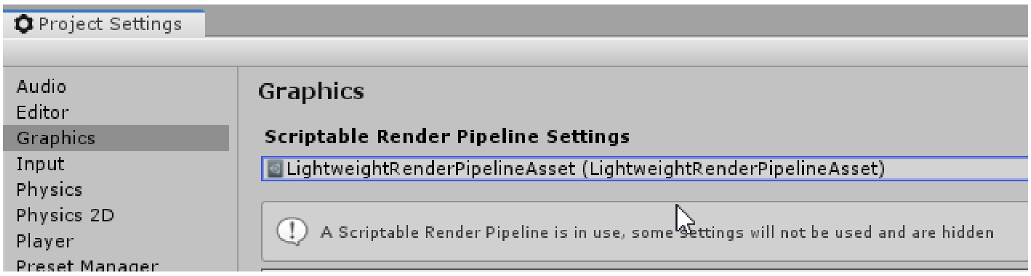
Tutorial

First, we need an ***LWRPipeline asset*** that you can control. You might already set up one for other purposes, in which case, you can skip the next 2 steps.

Start by creating an ***LWRPipeline asset*** like so:



Then assign it to your **Graphics Settings**:



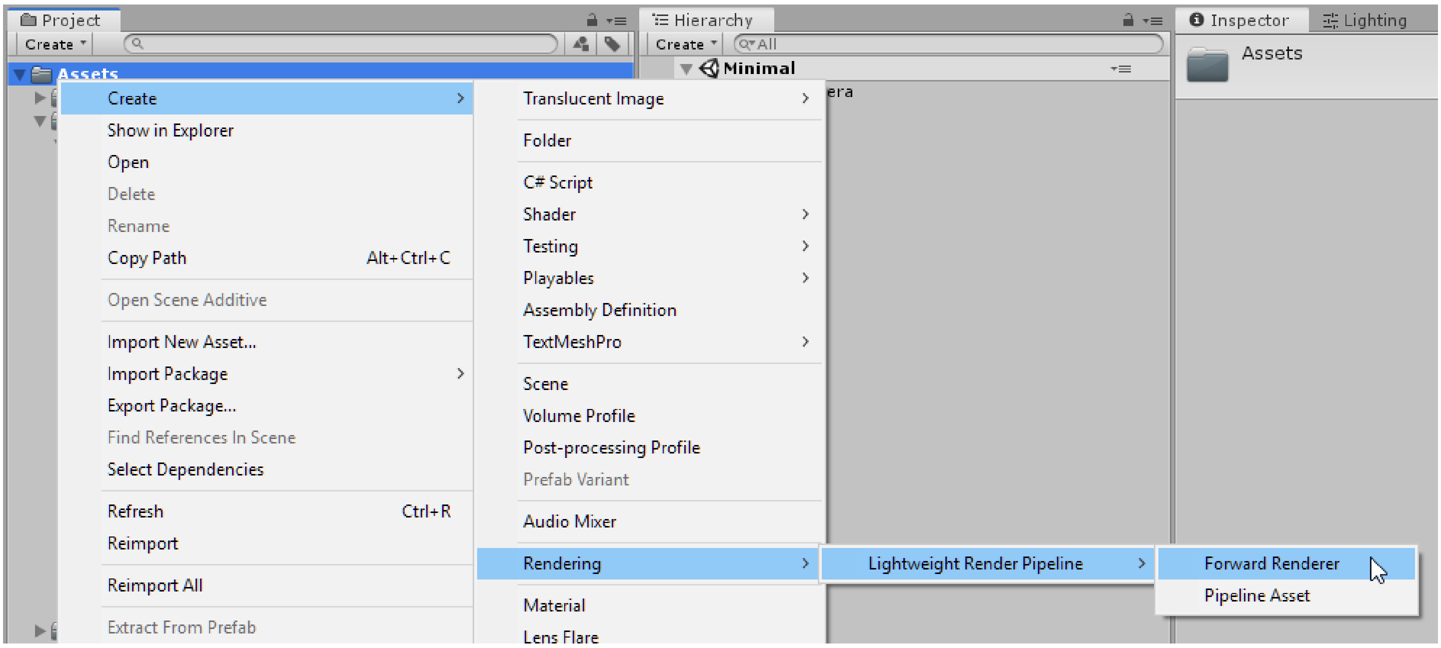
Graphics

Scriptable Render Pipeline Settings

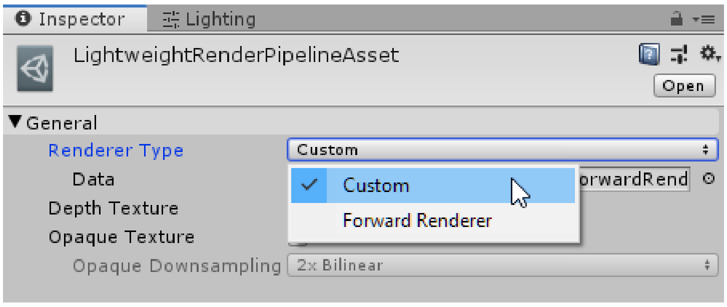
LightweightRenderPipelineAsset (LightweightRenderPipelineAsset)

A Scriptable Render Pipeline is in use, some settings will not be used and are hidden

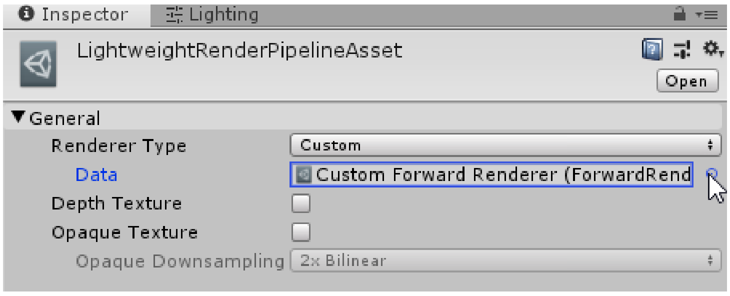
Next, you need a Custom ***Forward Renderer*** Asset. You might also have this already, if not, create one:



And assign it to the ***LWRPipeline asset*** you just created:

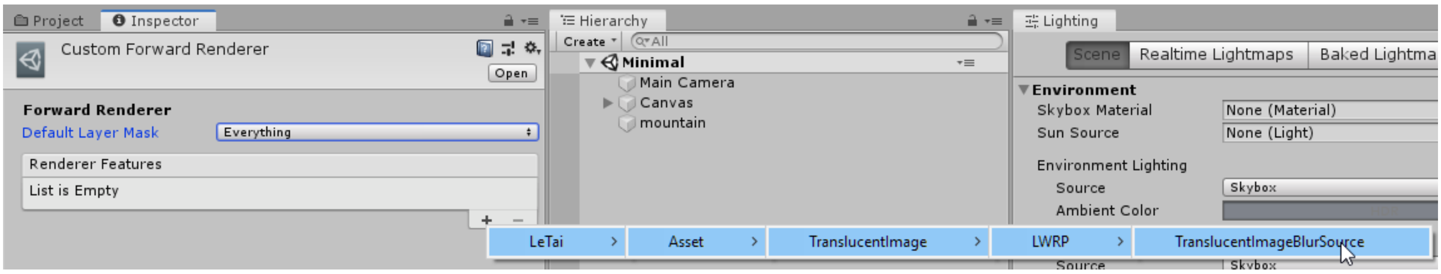


Renderer Type: Custom
Data: orwardRend
Depth Texture
Opaque Texture
Opaque Downsampling: 2x Bilinear



Renderer Type: Custom
Data: Custom Forward Renderer (ForwardRend
Depth Texture ☐
Opaque Texture ☐
Opaque Downsampling: 2x Bilinear

Last, add *Translucent Image Blur Source* as a *Render Feature* of your Custom ***Forward Renderer***



Forward Renderer
Default Layer Mask: Everything
Renderer Features
List is Empty

Environment
Skybox Material: None (Material)
Sun Source: None (Light)
Environment Lighting
Source: Skybox
Ambient Color

Limitations

LWRP does not support multiple cameras overlay each other. Consequently, multiple blur layers for Translucent Images are also not available.

Only the `Minimal` Demo scenes work in LWRP project for now, as the material used in other scenes is from the standard pipeline, and the main Demo scene uses multiple cameras, which is also not supported by LWRP.

Frequently Asked Questions

Will this asset works well on my device?

The asset should run on any device. Performance wise, it depends on your project, but here some general rule of thumb:

- PC/Mac/Console: Should run well on everything.
- Android: There's too many of them with too much difference in power. The only way to know for sure is to test the demo on your target devices. On the Samsung Galaxy S7 Edge, the demo run at 60FPS at any setting.
- IOS: Apple A8 and later should hit 60FPS. A7 can hit 30FPS.

Can I smoothly animate the blur level?

The strength property allows for some smoothness, but not fully, no. If you just need to fade in and out, use the alpha value of the Color. You can also use Canvas Group as normal Images.

Can I blur other UI?

TL;DR: Kind of. See the demo. Except for LWRP

Explanation

If the blur algorithm runs once for each UI on the screen, it will get too slow very fast. In fact, if you use a Mac or Window 10 computer, you will notice the blur effect is disabled for windows that aren't in focus.

Therefore, the blurring is done once per camera. This gives us a much higher performance. However, every TranslucentImage share the same source will have the exact same background - which means they will not "see" others below them.

Solution

If you need to blur UIs, for example, for a fully UI games, this can be done by setting up another *Camera* and *Canvas*. Note that this only work for Standard Render Pipeline, as LWRP does not support stacking Camera setup.

The setup:

- *Main Camera*: Render the scene but NOT UIs
- *UI Camera*:
    - Render ONLY UIs.
    - Clear flag set to "Depth only".
    - Should have higher Depth than the *Main Camera*.
    - Add a TranslucentImageSource here.
- *Overlay Canvas*:
    - In Screen Space Overlay mode (Render on top of everything).
    - This is where we put our TranslucentImage (TI for short).
    - These TIs have source set to *UI Camera*.
- *Camera Canvas*:
    - In Screen Space Camera mode (Render Camera set to the *UI Camera*).
    - This is where we put all our normal UIs.

You can also have TIs in the *Camera Canvas*. Just add another TranslucentImageSource to the *Main Camera*, and set all the TIs in the *Camera Canvas* to that Source.

Have another question?

Contact me

Support

If you need assistance regarding the asset or have a feature request, feel free to contact me by the form below or through my support email.

Support request

Search Articles

Namespace LeTai.Asset.TranslucentImage

Classes

BlurConfig

Extensions

ScalableBlur

ScalableBlurConfig

ShaderProperties

TranslucentImage

Dynamic blur-behind UI element

TranslucentImageSource

Common source of blur for Translucent Images.

Interfaces

IBlurAlgorithm

Enums

BlurAlgorithmType

Enum BlurAlgorithmType

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public enum BlurAlgorithmType
```

Fields

NAME	DESCRIPTION
ScalableBlur	

Class BlurConfig

Inheritance

System.Object

UnityEngine.Object

UnityEngine.ScriptableObject

BlurConfig

ScalableBlurConfig

Inherited Members

UnityEngine.ScriptableObject.SetDirty()

UnityEngine.ScriptableObject.CreateInstance(System.String)

UnityEngine.ScriptableObject.CreateInstance(System.Type)

UnityEngine.ScriptableObject.CreateInstance<T>()

UnityEngine.Object.GetInstanceID()

UnityEngine.Object.GetHashCode()

UnityEngine.Object.Equals(System.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Instantiate<T>(T)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)

UnityEngine.Object.Destroy(UnityEngine.Object)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object)

UnityEngine.Object.FindObjectsOfType(System.Type)

UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)

UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)

UnityEngine.Object.DestroyObject(UnityEngine.Object)

UnityEngine.Object.FindSceneObjectsOfType(System.Type)

UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)

UnityEngine.Object.FindObjectsOfType<T>()

UnityEngine.Object.FindObjectOfType<T>()

UnityEngine.Object.FindObjectsOfTypeAll(System.Type)

UnityEngine.Object.FindObjectOfType(System.Type)

UnityEngine.Object.ToString()

UnityEngine.Object.name

UnityEngine.Object.hideFlags

System.Object.Equals(System.Object, System.Object)

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public class BlurConfig : ScriptableObject
```

Class Extensions

Inheritance

System.Object
Extensions

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public static class Extensions
```

Methods

BlitFullscreenTriangle(CommandBuffer, RenderTargetIdentifier, RenderTargetIdentifier, Material, Int32)

Declaration

```
public static void BlitFullscreenTriangle(this CommandBuffer cmd, RenderTargetIdentifier source, RenderTargetIdentifier destination, Material material, int pass)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.CommandBuffer	cmd	
UnityEngine.Rendering.RenderTargetIdentifier	source	
UnityEngine.Rendering.RenderTargetIdentifier	destination	
UnityEngine.Material	material	
System.Int32	pass	

ToMinMaxVector(Rect)

Declaration

```
public static Vector4 ToMinMaxVector(this Rect self)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rect	self	

Returns

TYPE	DESCRIPTION
UnityEngine.Vector4	

Interface IBlurAlgorithm

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public interface IBlurAlgorithm
```

Methods

Blur(RenderTexture, Rect, ref RenderTexture)

Declaration

```
void Blur(RenderTexture source, Rect sourceCropRegion, ref RenderTexture blurredTexture)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.RenderTexture	source	
UnityEngine.Rect	sourceCropRegion	
UnityEngine.RenderTexture	blurredTexture	

Init(BlurConfig)

Declaration

```
void Init(BlurConfig config)
```

Parameters

TYPE	NAME	DESCRIPTION
BlurConfig	config	

Class ScalableBlur

Inheritance

System.Object
ScalableBlur

Implements

IBlurAlgorithm

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public class ScalableBlur : IBlurAlgorithm
```

Methods

Blur(RenderTexture, Rect, ref RenderTexture)

Declaration

```
public void Blur(RenderTexture source, Rect sourceCropRegion, ref RenderTexture blurredTexture)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.RenderTexture	source	
UnityEngine.Rect	sourceCropRegion	
UnityEngine.RenderTexture	blurredTexture	

BlurAtDepth(Int32, ref RenderTexture, ref RenderTexture)

Declaration

```
protected virtual void BlurAtDepth(int depth, ref RenderTexture baseTexture, ref RenderTexture target)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	depth	
UnityEngine.RenderTexture	baseTexture	
UnityEngine.RenderTexture	target	

ConfigMaterial(Single, Vector4)

Declaration

```
protected void ConfigMaterial(float radius, Vector4 cropRegion)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Single	radius	
UnityEngine.Vector4	cropRegion	

Init(BlurConfig)

Declaration

```
public void Init(BlurConfig config)
```

Parameters

TYPE	NAME	DESCRIPTION
BlurConfig	config	

Implements

IBlurAlgorithm

Class ScalableBlurConfig

Inheritance

System.Object

UnityEngine.Object

UnityEngine.ScriptableObject

BlurConfig

ScalableBlurConfig

Inherited Members

UnityEngine.ScriptableObject.SetDirty()

UnityEngine.ScriptableObject.CreateInstance(System.String)

UnityEngine.ScriptableObject.CreateInstance(System.Type)

UnityEngine.ScriptableObject.CreateInstance<T>()

UnityEngine.Object.GetInstanceID()

UnityEngine.Object.GetHashCode()

UnityEngine.Object.Equals(System.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Instantiate<T>(T)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)

UnityEngine.Object.Destroy(UnityEngine.Object)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object)

UnityEngine.Object.FindObjectsOfType(System.Type)

UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)

UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)

UnityEngine.Object.DestroyObject(UnityEngine.Object)

UnityEngine.Object.FindSceneObjectsOfType(System.Type)

UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)

UnityEngine.Object.FindObjectsOfType<T>()

UnityEngine.Object.FindObjectOfType<T>()

UnityEngine.Object.FindObjectsOfTypeAll(System.Type)

UnityEngine.Object.FindObjectOfType(System.Type)

UnityEngine.Object.ToString()

UnityEngine.Object.name

UnityEngine.Object.hideFlags

System.Object.Equals(System.Object, System.Object)

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
[CreateAssetMenu(fileName = "New Scalable Blur Config", menuName = "Translucent Image/ Scalable Blur Config")]
public class ScalableBlurConfig : BlurConfig
```

Properties

Iteration

Half the number of time to process the image. It is half because the real number of iteration must alway be even. Using half also make calculation simpler

Declaration

```
public int Iteration { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Must be non-negative

MaxDepth

Clamp the minimum size of the intermediate texture. Reduce flickering and blur

Declaration

```
public int MaxDepth { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Must larger than 0

Radius

Distance between the base texel and the texel to be sampled.

Declaration

```
public float Radius { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Single	

Strength

User friendly property to control the amount of blur

Declaration

```
public float Strength { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Single	Must be non-negative

Methods

SetAdvancedFieldFromSimple()

Calculate size and iteration from strength

Declaration

```
protected virtual void SetAdvancedFieldFromSimple()
```

Class ShaderProperties

Inheritance

System.Object
ShaderProperties

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public static class ShaderProperties
```

Fields

blurRadius

Declaration

```
public static int blurRadius
```

Field Value

TYPE	DESCRIPTION
System.Int32	

blurTextureCropRegion

Declaration

```
public static int blurTextureCropRegion
```

Field Value

TYPE	DESCRIPTION
System.Int32	

intermediateRT

Declaration

```
public static int[] intermediateRT
```

Field Value

TYPE	DESCRIPTION
System.Int32[]	

Methods

Init()

Declaration

```
public static void Init()
```

Init(Int32)

Declaration

```
public static void Init(int stackDepth)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	stackDepth	

Class TranslucentImage

Dynamic blur-behind UI element

Inheritance

System.Object
UnityEngine.Object
UnityEngine.Component
UnityEngine.Behaviour
UnityEngine.MonoBehaviour
UnityEngine.EventSystems.UIBehaviour
UnityEngine.UI.Graphic
UnityEngine.UI.MaskableGraphic
UnityEngine.UI.Image
TranslucentImage

Implements

UnityEngine.UI.ICanvasElement
UnityEngine.UI.IClippable
UnityEngine.UI.IMaskable
UnityEngine.UI.IMaterialModifier
UnityEngine.ISerializationCallbackReceiver
UnityEngine.UI.ILayoutElement
UnityEngine.ICanvasRaycastFilter
UnityEngine.UI.IMeshModifier

Inherited Members

UnityEngine.UI.Image.s_ETC1DefaultUI
UnityEngine.UI.Image.DisableSpriteOptimizations()
UnityEngine.UI.Image.OnBeforeSerialize()
UnityEngine.UI.Image.OnAfterDeserialize()
UnityEngine.UI.Image.SetNativeSize()
UnityEngine.UI.Image.OnPopulateMesh(UnityEngine.UI.VertexHelper)
UnityEngine.UI.Image.UpdateGeometry()
UnityEngine.UI.Image.UpdateMaterial()
UnityEngine.UI.Image.OnCanvasHierarchyChanged()
UnityEngine.UI.Image.CalculateLayoutInputHorizontal()
UnityEngine.UI.Image.CalculateLayoutInputVertical()
UnityEngine.UI.Image.IsRaycastLocationValid(UnityEngine.Vector2, UnityEngine.Camera)
UnityEngine.UI.Image.sprite
UnityEngine.UI.Image.overrideSprite
UnityEngine.UI.Image.type
UnityEngine.UI.Image.preserveAspect
UnityEngine.UI.Image.fillCenter
UnityEngine.UI.Image.fillMethod
UnityEngine.UI.Image.fillAmount
UnityEngine.UI.Image.fillClockwise
UnityEngine.UI.Image.fillOrigin
UnityEngine.UI.Image.eventAlphaThreshold
UnityEngine.UI.Image.alphaHitTestMinimumThreshold
UnityEngine.UI.Image.useSpriteMesh
UnityEngine.UI.Image.defaultETC1GraphicMaterial
UnityEngine.UI.Image.mainTexture

UnityEngine.UI.Image.hasBorder

UnityEngine.UI.Image.pixelsPerUnit

UnityEngine.UI.Image.material

UnityEngine.UI.Image.minWidth

UnityEngine.UI.Image.preferredWidth

UnityEngine.UI.Image.flexibleWidth

UnityEngine.UI.Image.minHeight

UnityEngine.UI.Image.preferredHeight

UnityEngine.UI.Image.flexibleHeight

UnityEngine.UI.Image.layoutPriority

UnityEngine.UI.MaskableGraphic.m_ShouldRecalculateStencil

UnityEngine.UI.MaskableGraphic.m_MaskMaterial

UnityEngine.UI.MaskableGraphic.m_StencilValue

UnityEngine.UI.MaskableGraphic.GetModifiedMaterial(UnityEngine.Material)

UnityEngine.UI.MaskableGraphic.Cull(UnityEngine.Rect, System.Boolean)

UnityEngine.UI.MaskableGraphic.SetClipRect(UnityEngine.Rect, System.Boolean)

UnityEngine.UI.MaskableGraphic.OnValidate()

UnityEngine.UI.MaskableGraphic.OnTransformParentChanged()

UnityEngine.UI.MaskableGraphic.RecalculateClipping()

UnityEngine.UI.MaskableGraphic.RecalculateMasking()

UnityEngine.UI.MaskableGraphic.UnityEngine.UI.IClippable.get_gameObject()

UnityEngine.UI.MaskableGraphic.onCullStateChanged

UnityEngine.UI.MaskableGraphic.maskable

UnityEngine.UI.Graphic.s_DefaultUI

UnityEngine.UI.Graphic.s_WhiteTexture

UnityEngine.UI.Graphic.m_Material

UnityEngine.UI.Graphic.m_SkipLayoutUpdate

UnityEngine.UI.Graphic.m_SkipMaterialUpdate

UnityEngine.UI.Graphic.m_OnDirtyLayoutCallback

UnityEngine.UI.Graphic.m_OnDirtyVertsCallback

UnityEngine.UI.Graphic.m_OnDirtyMaterialCallback

UnityEngine.UI.Graphic.s_Mesh

UnityEngine.UI.Graphic.m_CachedMesh

UnityEngine.UI.Graphic.m_CachedUvs

UnityEngine.UI.Graphic.SetAllDirty()

UnityEngine.UI.Graphic.SetLayoutDirty()

UnityEngine.UI.Graphic.SetVerticesDirty()

UnityEngine.UI.Graphic.SetMaterialDirty()

UnityEngine.UI.Graphic.OnRectTransformDimensionsChange()

UnityEngine.UI.Graphic.OnBeforeTransformParentChanged()

UnityEngine.UI.Graphic.OnDestroy()

UnityEngine.UI.Graphic.OnCullingChanged()

UnityEngine.UI.Graphic.Rebuild(UnityEngine.UI.CanvasUpdate)

UnityEngine.UI.Graphic.LayoutComplete()

UnityEngine.UI.Graphic.GraphicUpdateComplete()

UnityEngine.UI.Graphic.OnPopulateMesh(UnityEngine.Mesh)

UnityEngine.UI.Graphic.OnRebuildRequested()

UnityEngine.UI.Graphic.Reset()

UnityEngine.UI.Graphic.Raycast(UnityEngine.Vector2, UnityEngine.Camera)

UnityEngine.UI.Graphic.PixelAdjustPoint(UnityEngine.Vector2)

UnityEngine.UI.Graphic.GetPixelAdjustedRect()

UnityEngine.UI.Graphic.CrossFadeColor(UnityEngine.Color, System.Single, System.Boolean, System.Boolean)

UnityEngine.UI.Graphic.CrossFadeColor(UnityEngine.Color, System.Single, System.Boolean, System.Boolean, System.Boolean)

UnityEngine.UI.Graphic.CrossFadeAlpha(System.Single, System.Single, System.Boolean)

UnityEngine.UI.Graphic.RegisterDirtyLayoutCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.UnregisterDirtyLayoutCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.RegisterDirtyVerticesCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.UnregisterDirtyVerticesCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.RegisterDirtyMaterialCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.UnregisterDirtyMaterialCallback(UnityEngine.Events.UnityAction)

UnityEngine.UI.Graphic.UnityEngine.UI.ICanvasElement.get_transform()

UnityEngine.UI.Graphic.defaultGraphicMaterial

UnityEngine.UI.Graphic.color

UnityEngine.UI.Graphic.raycastTarget

UnityEngine.UI.Graphic.useLegacyMeshGeneration

UnityEngine.UI.Graphic.depth

UnityEngine.UI.Graphic.rectTransform

UnityEngine.UI.Graphic.canvas

UnityEngine.UI.Graphic.canvasRenderer

UnityEngine.UI.Graphic.defaultMaterial

UnityEngine.UI.Graphic.materialForRendering

UnityEngine.UI.Graphic.workerMesh

UnityEngine.EventSystems.UIBehaviour.Awake()

UnityEngine.EventSystems.UIBehaviour.IsActive()

UnityEngine.EventSystems.UIBehaviour.OnCanvasGroupChanged()

UnityEngine.EventSystems.UIBehaviour.IsDestroyed()

UnityEngine.MonoBehaviour.IsInvoking()

UnityEngine.MonoBehaviour.CancelInvoke()

UnityEngine.MonoBehaviour.Invoke(System.String, System.Single)

UnityEngine.MonoBehaviour.InvokeRepeating(System.String, System.Single, System.Single)

UnityEngine.MonoBehaviour.CancelInvoke(System.String)

UnityEngine.MonoBehaviour.IsInvoking(System.String)

UnityEngine.MonoBehaviour.StartCoroutine(System.String)

UnityEngine.MonoBehaviour.StartCoroutine(System.String, System.Object)

UnityEngine.MonoBehaviour.StartCoroutine(System.Collections.IEnumerator)

UnityEngine.MonoBehaviour.StartCoroutine_Auto(System.Collections.IEnumerator)

UnityEngine.MonoBehaviour.StopCoroutine(System.Collections.IEnumerator)

UnityEngine.MonoBehaviour.StopCoroutine(UnityEngine.Coroutine)

UnityEngine.MonoBehaviour.StopCoroutine(System.String)

UnityEngine.MonoBehaviour.StopAllCoroutines()

UnityEngine.MonoBehaviour.print(System.Object)

UnityEngine.MonoBehaviour.useGUILayout

UnityEngine.MonoBehaviour.runInEditMode

UnityEngine.Behaviour.enabled

UnityEngine.Behaviour.isActiveAndEnabled

UnityEngine.Component.GetComponent(System.Type)

UnityEngine.Component.GetComponent<T>()

UnityEngine.Component.GetComponent(System.String)

UnityEngine.Component.GetComponentInChildren(System.Type, System.Boolean)

UnityEngine.Component.GetComponentInChildren(System.Type)

UnityEngine.Component.GetComponentInChildren<T>(System.Boolean)

UnityEngine.Component.GetComponentInChildren<T>()

UnityEngine.Component.GetComponentsInChildren(System.Type, System.Boolean)

UnityEngine.Component.GetComponentsInChildren(System.Type)

UnityEngine.Component.GetComponentsInChildren<T>(System.Boolean)

UnityEngine.Component.GetComponentsInChildren<T>(System.Boolean, System.Collections.Generic.List<T>)

UnityEngine.Component.GetComponentsInChildren<T>()

UnityEngine.Component.GetComponentsInChildren<T>(System.Collections.Generic.List<T>)

UnityEngine.Component.GetComponentInParent(System.Type)

UnityEngine.Component.GetComponentInParent<T>()

UnityEngine.Component.GetComponentsInParent(System.Type, System.Boolean)

UnityEngine.Component.GetComponentsInParent(System.Type)

UnityEngine.Component.GetComponentsInParent<T>(System.Boolean)

UnityEngine.Component.GetComponentsInParent<T>(System.Boolean, System.Collections.Generic.List<T>)

UnityEngine.Component.GetComponentsInParent<T>()

UnityEngine.Component.GetComponents(System.Type)

UnityEngine.Component.GetComponents(System.Type, System.Collections.Generic.List<UnityEngine.Component>)

UnityEngine.Component.GetComponents<T>(System.Collections.Generic.List<T>)

UnityEngine.Component.GetComponents<T>()

UnityEngine.Component.CompareTag(System.String)

UnityEngine.Component.SendMessageUpwards(System.String, System.Object, UnityEngine.SendMessageOptions)

UnityEngine.Component.SendMessageUpwards(System.String, System.Object)

UnityEngine.Component.SendMessageUpwards(System.String)

UnityEngine.Component.SendMessageUpwards(System.String, UnityEngine.SendMessageOptions)

UnityEngine.Component.SendMessage(System.String, System.Object)

UnityEngine.Component.SendMessage(System.String)

UnityEngine.Component.SendMessage(System.String, System.Object, UnityEngine.SendMessageOptions)

UnityEngine.Component.SendMessage(System.String, UnityEngine.SendMessageOptions)

UnityEngine.Component.BroadcastMessage(System.String, System.Object, UnityEngine.SendMessageOptions)

UnityEngine.Component.BroadcastMessage(System.String, System.Object)

UnityEngine.Component.BroadcastMessage(System.String)

UnityEngine.Component.BroadcastMessage(System.String, UnityEngine.SendMessageOptions)

UnityEngine.Component.transform

UnityEngine.Component.gameObject

UnityEngine.Component.tag

UnityEngine.Object.GetInstanceID()

UnityEngine.Object.GetHashCode()

UnityEngine.Object.Equals(System.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Instantiate<T>(T)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)

UnityEngine.Object.Destroy(UnityEngine.Object)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object)

UnityEngine.Object.FindObjectsOfType(System.Type)

UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)

UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)

UnityEngine.Object.DestroyObject(UnityEngine.Object)

UnityEngine.Object.FindSceneObjectsOfType(System.Type)
UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)
UnityEngine.Object.FindObjectsOfType<T>()
UnityEngine.Object.FindObjectOfType<T>()
UnityEngine.Object.FindObjectsOfTypeAll(System.Type)
UnityEngine.Object.FindObjectOfType(System.Type)
UnityEngine.Object.ToString()
UnityEngine.Object.name
UnityEngine.Object.hideFlags
System.Object.Equals(System.Object, System.Object)
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
public class TranslucentImage : Image, ICanvasElement, IClippable, IMaskable, IMaterialModifier, ISerializationCallbackReceiver,
ILayoutElement, ICanvasRaycastFilter, IMeshModifier
```

Fields

brightness

Brighten/darken them image

Declaration

```
[Tooltip("Brighten/darken them image")]
[Range(-1F, 1F)]
public float brightness
```

Field Value

TYPE	DESCRIPTION
System.Single	

flatten

Flatten the color behind to help keep contrast on varying background

Declaration

```
[Tooltip("Flatten the color behind to help keep contrast on varying background")]
[Range(0F, 1F)]
public float flatten
```

Field Value

TYPE	DESCRIPTION
System.Single	

source

Source of blur for this image

Declaration

```
public TranslucentImageSource source
```

Field Value

TYPE	DESCRIPTION
TranslucentImageSource	

spriteBlending

Declaration

```
[Tooltip("Blend between the sprite and background blur")]
[Range(0F, 1F)]
public float spriteBlending
```

Field Value

TYPE	DESCRIPTION
System.Single	

vibrancy

(De)Saturate them image, 1 is normal, 0 is grey scale, below zero make the image negative

Declaration

```
[Tooltip("(De)Saturate them image, 1 is normal, 0 is black and white, below zero make the image negative")]
[Range(-1F, 3F)]
public float vibrancy
```

Field Value

TYPE	DESCRIPTION
System.Single	

Methods

ModifyMesh(Mesh)

Declaration

```
public virtual void ModifyMesh(Mesh mesh)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Mesh	mesh	

ModifyMesh(VertexHelper)

Declaration

```
public virtual void ModifyMesh(VertexHelper vh)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.UI.VertexHelper	vh	

OnDidApplyAnimationProperties()

Declaration

```
protected override void OnDidApplyAnimationProperties()
```

Overrides

UnityEngine.UI.Graphic.OnDidApplyAnimationProperties()

OnDisable()

Declaration

```
protected override void OnDisable()
```

Overrides

UnityEngine.UI.Image.OnDisable()

OnEnable()

Declaration

```
protected override void OnEnable()
```

Overrides

UnityEngine.UI.Image.OnEnable()

Start()

Declaration

```
protected override void Start()
```

Overrides

UnityEngine.EventSystems.UIBehaviour.Start()

Implements

UnityEngine.UI.ICanvasElement
UnityEngine.UI.IClippable
UnityEngine.UI.IMaskable
UnityEngine.UI.IMaterialModifier
UnityEngine.ISerializationCallbackReceiver
UnityEngine.UI.ILayoutElement
UnityEngine.ICanvasRaycastFilter
UnityEngine.UI.IMeshModifier

Class TranslucentImageSource

Common source of blur for Translucent Images.

Inheritance

System.Object
UnityEngine.Object
UnityEngine.Component
UnityEngine.Behaviour
UnityEngine.MonoBehaviour
TranslucentImageSource

Inherited Members

UnityEngine.MonoBehaviour.IsInvoking()
UnityEngine.MonoBehaviour.CancelInvoke()
UnityEngine.MonoBehaviour.Invoke(System.String, System.Single)
UnityEngine.MonoBehaviour.InvokeRepeating(System.String, System.Single, System.Single)
UnityEngine.MonoBehaviour.CancelInvoke(System.String)
UnityEngine.MonoBehaviour.IsInvoking(System.String)
UnityEngine.MonoBehaviour.StartCoroutine(System.String)
UnityEngine.MonoBehaviour.StartCoroutine(System.String, System.Object)
UnityEngine.MonoBehaviour.StartCoroutine(System.Collections.IEnumerator)
UnityEngine.MonoBehaviour.StartCoroutine_Auto(System.Collections.IEnumerator)
UnityEngine.MonoBehaviour.StopCoroutine(System.Collections.IEnumerator)
UnityEngine.MonoBehaviour.StopCoroutine(UnityEngine.Coroutine)
UnityEngine.MonoBehaviour.StopCoroutine(System.String)
UnityEngine.MonoBehaviour.StopAllCoroutines()
UnityEngine.MonoBehaviour.print(System.Object)
UnityEngine.MonoBehaviour.useGUILayout
UnityEngine.MonoBehaviour.runInEditMode
UnityEngine.Behaviour.enabled
UnityEngine.Behaviour.isActiveAndEnabled
UnityEngine.Component.GetComponent(System.Type)
UnityEngine.Component.GetComponent<T>()
UnityEngine.Component.GetComponent(System.String)
UnityEngine.Component.GetComponentInChildren(System.Type, System.Boolean)
UnityEngine.Component.GetComponentInChildren(System.Type)
UnityEngine.Component.GetComponentInChildren<T>(System.Boolean)
UnityEngine.Component.GetComponentInChildren<T>()
UnityEngine.Component.GetComponentsInChildren(System.Type, System.Boolean)
UnityEngine.Component.GetComponentsInChildren(System.Type)
UnityEngine.Component.GetComponentsInChildren<T>(System.Boolean)
UnityEngine.Component.GetComponentsInChildren<T>(System.Boolean, System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentsInChildren<T>()
UnityEngine.Component.GetComponentsInChildren<T>(System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentInParent(System.Type)
UnityEngine.Component.GetComponentInParent<T>()
UnityEngine.Component.GetComponentsInParent(System.Type, System.Boolean)
UnityEngine.Component.GetComponentsInParent(System.Type)
UnityEngine.Component.GetComponentsInParent<T>(System.Boolean)
UnityEngine.Component.GetComponentsInParent<T>(System.Boolean, System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentsInParent<T>()
UnityEngine.Component.GetComponents(System.Type)

```
UnityEngine.Component.GetComponents(System.Type, System.Collections.Generic.List<UnityEngine.Component>)
UnityEngine.Component.GetComponents<T>(System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponents<T>()
UnityEngine.Component.CompareTag(System.String)
UnityEngine.Component.SendMessageUpwards(System.String, System.Object, UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessageUpwards(System.String, System.Object)
UnityEngine.Component.SendMessageUpwards(System.String)
UnityEngine.Component.SendMessageUpwards(System.String, UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessage(System.String, System.Object)
UnityEngine.Component.SendMessage(System.String)
UnityEngine.Component.SendMessage(System.String, System.Object, UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessage(System.String, UnityEngine.SendMessageOptions)
UnityEngine.Component.BroadcastMessage(System.String, System.Object, UnityEngine.SendMessageOptions)
UnityEngine.Component.BroadcastMessage(System.String, System.Object)
UnityEngine.Component.BroadcastMessage(System.String)
UnityEngine.Component.BroadcastMessage(System.String, UnityEngine.SendMessageOptions)
UnityEngine.Component.transform
UnityEngine.Component.gameObject
UnityEngine.Component.tag
UnityEngine.Object.GetInstanceID()
UnityEngine.Object.GetHashCode()
UnityEngine.Object.Equals(System.Object)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)
UnityEngine.Object.Instantiate(UnityEngine.Object)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform, System.Boolean)
UnityEngine.Object.Instantiate<T>(T)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform, System.Boolean)
UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)
UnityEngine.Object.Destroy(UnityEngine.Object)
UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)
UnityEngine.Object.DestroyImmediate(UnityEngine.Object)
UnityEngine.Object.FindObjectsOfType(System.Type)
UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)
UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)
UnityEngine.Object.DestroyObject(UnityEngine.Object)
UnityEngine.Object.FindSceneObjectsOfType(System.Type)
UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)
UnityEngine.Object.FindObjectsOfType<T>()
UnityEngine.Object.FindObjectOfType<T>()
UnityEngine.Object.FindObjectsOfTypeAll(System.Type)
UnityEngine.Object.FindObjectOfType(System.Type)
UnityEngine.Object.ToString()
UnityEngine.Object.name
UnityEngine.Object.hideFlags
System.Object.Equals(System.Object, System.Object)
System.Object.GetType()
System.Object.MemberwiseClone()
```

System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: LeTai.Asset.TranslucentImage

Assembly: LeTai.TranslucentImage.dll

Syntax

```
[ExecuteInEditMode]
[RequireComponent(typeof(Camera))]
[AddComponentMenu("Image Effects/Tai Le Assets/Translucent Image Source")]
public class TranslucentImageSource : MonoBehaviour
```

Remarks

It is an Image effect that blur the render target of the Camera it attached to, then save the result to a global read-only Render Texture

Fields

maxUpdateRate

Maximum number of times to update the blurred image each second

Declaration

```
public float maxUpdateRate
```

Field Value

TYPE	DESCRIPTION
System.Single	

preview

Render the blurred result to the render target

Declaration

```
public bool preview
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

Properties

BlurAlgorithmSelection

Declaration

```
public BlurAlgorithmType BlurAlgorithmSelection { get; set; }
```

Property Value

TYPE	DESCRIPTION
BlurAlgorithmType	

BlurConfig

Declaration

```
public BlurConfig BlurConfig { get; set; }
```

Property Value

TYPE	DESCRIPTION
BlurConfig	

BlurredScreen

Result of the image effect. Translucent Image use this as their content (read-only)

Declaration

```
public RenderTexture BlurredScreen { get; set; }
```

Property Value

TYPE	DESCRIPTION
UnityEngine.RenderTexture	

BlurRegion

Define the rectangular area on screen that will be blurred.

Declaration

```
public Rect BlurRegion { get; set; }
```

Property Value

TYPE	DESCRIPTION
UnityEngine.Rect	Between 0 and 1

Downsample

The rendered image will be shrinked by a factor of 2^{{this}} before bluring to reduce processing time

Declaration

```
public int Downsample { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Must be non-negative. Default to 0

Methods

Awake()

Declaration

```
protected virtual void Awake()
```

CreateNewBlurredScreen()

Declaration

```
protected virtual void CreateNewBlurredScreen()
```

OnRenderImage(RenderTexture, RenderTexture)

Declaration

```
protected virtual void OnRenderImage(RenderTexture source, RenderTexture destination)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.RenderTexture	source	
UnityEngine.RenderTexture	destination	

shouldUpdateBlur()

Declaration

```
public bool shouldUpdateBlur()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

Start()

Declaration

```
protected virtual void Start()
```

Namespace LeTai.Asset.TranslucentImage.LWRP

Classes

ScalableBlur

TranslucentImageBlurRenderPass

TranslucentImageBlurSource

Utilities

Interfaces

IBlurAlgorithm

Enums

BlurAlgorithmType

Enum BlurAlgorithmType

Namespace: LeTai.Asset.TranslucentImage.LWRP

Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public enum BlurAlgorithmType
```

Fields

NAME	DESCRIPTION
ScalableBlur	

Interface IBlurAlgorithm

Namespace: LeTai.Asset.TranslucentImage.LWRP

Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public interface IBlurAlgorithm
```

Methods

Blur(CommandBuffer, RenderTargetIdentifier, Rect, RenderTexture)

Declaration

```
void Blur(CommandBuffer cmd, RenderTargetIdentifier src, Rect srcCropRegion, RenderTexture target)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.CommandBuffer	cmd	
UnityEngine.Rendering.RenderTargetIdentifier	src	
UnityEngine.Rect	srcCropRegion	
UnityEngine.RenderTexture	target	

Init(BlurConfig)

Declaration

```
void Init(BlurConfig config)
```

Parameters

TYPE	NAME	DESCRIPTION
BlurConfig	config	

Class ScalableBlur

Inheritance

System.Object
ScalableBlur

Implements

IBlurAlgorithm

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage.LWRP

Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public class ScalableBlur : IBlurAlgorithm
```

Methods

Blur(CommandBuffer, RenderTargetIdentifier, Rect, RenderTexture)

Declaration

```
public void Blur(CommandBuffer cmd, RenderTargetIdentifier src, Rect srcCropRegion, RenderTexture target)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.CommandBuffer	cmd	
UnityEngine.Rendering.RenderTargetIdentifier	src	
UnityEngine.Rect	srcCropRegion	
UnityEngine.RenderTexture	target	

BlurAtDepth(CommandBuffer, Int32, RenderTexture)

Declaration

```
protected virtual void BlurAtDepth(CommandBuffer cmd, int depth, RenderTexture baseTexture)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.CommandBuffer	cmd	
System.Int32	depth	

TYPE	NAME	DESCRIPTION
UnityEngine.RenderTexture	baseTexture	

ConfigMaterial(Single, Vector4)

Declaration

```
protected void ConfigMaterial(float radius, Vector4 cropRegion)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Single	radius	
UnityEngine.Vector4	cropRegion	

Init(BlurConfig)

Declaration

```
public void Init(BlurConfig config)
```

Parameters

TYPE	NAME	DESCRIPTION
BlurConfig	config	

Implements

IBlurAlgorithm

Class TranslucentImageBlurRenderPass

Inheritance

System.Object
UnityEngine.Rendering.LWRP.ScriptableRenderPass
TranslucentImageBlurRenderPass

Inherited Members

UnityEngine.Rendering.LWRP.ScriptableRenderPass.ConfigureTarget(UnityEngine.Rendering.RenderTargetIdentifier, UnityEngine.Rendering.RenderTargetIdentifier)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.ConfigureTarget(UnityEngine.Rendering.RenderTargetIdentifier)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.ConfigureClear(UnityEngine.Rendering.ClearFlag, UnityEngine.Color)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.Configure(UnityEngine.Rendering.CommandBuffer, UnityEngine.RenderTextureDescriptor)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.FrameCleanup(UnityEngine.Rendering.CommandBuffer)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.Blit(UnityEngine.Rendering.CommandBuffer, UnityEngine.Rendering.RenderTargetIdentifier, UnityEngine.Rendering.RenderTargetIdentifier, UnityEngine.Material, System.Int32)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.RenderPostProcessing(UnityEngine.Rendering.CommandBuffer, UnityEngine.Rendering.LWRP.CameraData, UnityEngine.RenderTextureDescriptor, UnityEngine.Rendering.RenderTargetIdentifier, UnityEngine.Rendering.RenderTargetIdentifier, System.Boolean, System.Boolean)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.CreateDrawingSettings(UnityEngine.Rendering.ShaderTagId, UnityEngine.Rendering.LWRP.RenderingData, UnityEngine.Rendering.SortingCriteria)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.CreateDrawingSettings(System.Collections.Generic.List<UnityEngine.Rendering.ShaderTagId>, UnityEngine.Rendering.LWRP.RenderingData, UnityEngine.Rendering.SortingCriteria)
UnityEngine.Rendering.LWRP.ScriptableRenderPass.renderPassEvent
UnityEngine.Rendering.LWRP.ScriptableRenderPass.colorAttachment
UnityEngine.Rendering.LWRP.ScriptableRenderPass.depthAttachment
UnityEngine.Rendering.LWRP.ScriptableRenderPass.clearFlag
UnityEngine.Rendering.LWRP.ScriptableRenderPass.clearColor
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage.LWRP
Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public class TranslucentImageBlurRenderPass : ScriptableRenderPass
```

Constructors

TranslucentImageBlurRenderPass()

Declaration

```
public TranslucentImageBlurRenderPass()
```

Methods

Execute(ScriptableRenderContext, ref RenderingData)

Declaration

```
public override void Execute(ScriptableRenderContext context, ref RenderingData renderingData)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.ScriptableRenderContext	context	
UnityEngine.Rendering.LWRP.RenderingData	renderingData	

Overrides

```
UnityEngine.Rendering.LWRP.ScriptableRenderPass.Execute(UnityEngine.Rendering.ScriptableRenderContext,
UnityEngine.Rendering.LWRP.RenderingData)
```

Class TranslucentImageBlurSource

Inheritance

System.Object

UnityEngine.Object

UnityEngine.ScriptableObject

UnityEngine.Rendering.LWRP.ScriptableRendererFeature

TranslucentImageBlurSource

Inherited Members

UnityEngine.ScriptableObject.SetDirty()

UnityEngine.ScriptableObject.CreateInstance(System.String)

UnityEngine.ScriptableObject.CreateInstance(System.Type)

UnityEngine.ScriptableObject.CreateInstance<T>()

UnityEngine.Object.GetInstanceID()

UnityEngine.Object.GetHashCode()

UnityEngine.Object.Equals(System.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Instantiate<T>(T)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)

UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform, System.Boolean)

UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)

UnityEngine.Object.Destroy(UnityEngine.Object)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)

UnityEngine.Object.DestroyImmediate(UnityEngine.Object)

UnityEngine.Object.FindObjectsOfType(System.Type)

UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)

UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)

UnityEngine.Object.DestroyObject(UnityEngine.Object)

UnityEngine.Object.FindSceneObjectsOfType(System.Type)

UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)

UnityEngine.Object.FindObjectsOfType<T>()

UnityEngine.Object.FindObjectOfType<T>()

UnityEngine.Object.FindObjectsOfTypeAll(System.Type)

UnityEngine.Object.FindObjectOfType(System.Type)

UnityEngine.Object.ToString()

UnityEngine.Object.name

UnityEngine.Object.hideFlags

System.Object.Equals(System.Object, System.Object)

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: LeTai.Asset.TranslucentImage.LWRP

Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public class TranslucentImageBlurSource : ScriptableRendererFeature
```

Methods

AddRenderPasses(ScriptableRenderer, ref RenderingData)

Declaration

```
public override void AddRenderPasses(ScriptableRenderer renderer, ref RenderingData renderingData)
```

Parameters

TYPE	NAME	DESCRIPTION
UnityEngine.Rendering.LWRP.ScriptableRenderer	renderer	
UnityEngine.Rendering.LWRP.RenderingData	renderingData	

Overrides

UnityEngine.Rendering.LWRP.ScriptableRendererFeature.AddRenderPasses(UnityEngine.Rendering.LWRP.ScriptableRenderer, UnityEngine.Rendering.LWRP.RenderingData)

Create()

Declaration

```
public override void Create()
```

Overrides

UnityEngine.Rendering.LWRP.ScriptableRendererFeature.Create()

RegisterSource(TranslucentImageSource)

When adding new Translucent Image Source to existing Camera at run time, the new Source must be registered here

Declaration

```
public void RegisterSource(TranslucentImageSource source)
```

Parameters

TYPE	NAME	DESCRIPTION
TranslucentImageSource	source	

Class Utilities

Inheritance

System.Object
Utilities

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: LeTai.Asset.TranslucentImage.LWRP

Assembly: LeTai.TranslucentImage.LWRP.dll

Syntax

```
public static class Utilities
```

Methods

SimplePingPong(Int32, Int32)

Declaration

```
public static int SimplePingPong(int t, int max)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	t	
System.Int32	max	

Returns

TYPE	DESCRIPTION
System.Int32	