Freezing Effect （URP & LWRP）

Introduction

If you want to turn your game character or game object into a piece of ice, then this asset is prefect for you. This is a combination of shader, script and audio. You don’ t need to write any shader yourself, you can easily achieve such a cool freezing effect, and it contains animation effects that can be controlled by scripts, and it also contains animation sound effects, and it can be easily integrated into your project, The shader graph file exposes many parameters that you can adjust yourself. Due to the cool effect achieved by using the shader, all performance losses are small, and the mobile platform can also run perfectly.

PS：Only support URP（Universal Renderer Pipeline）and LWRP (Light Weight Renderer Pipeline)

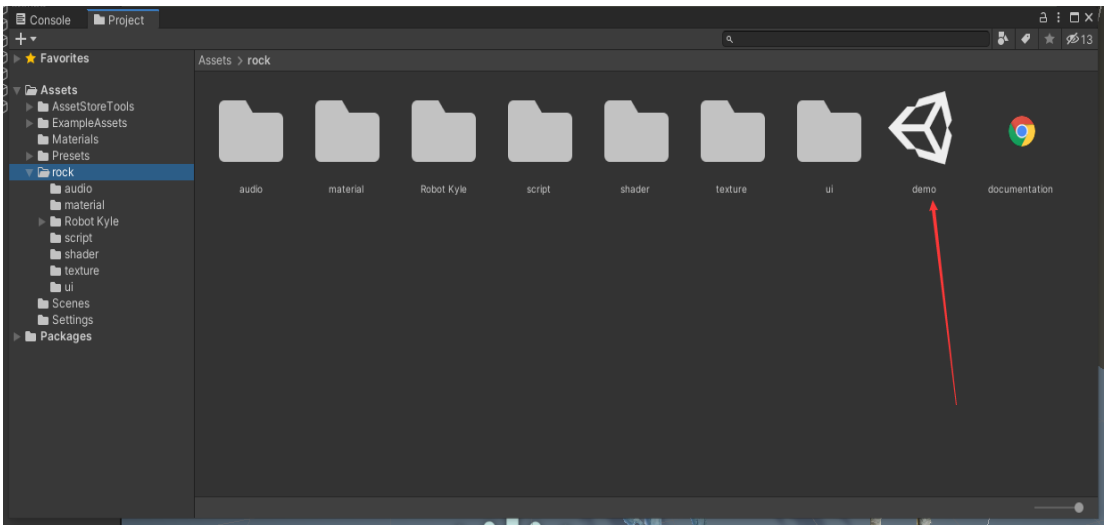
Main features：

1. Easy to integrate into your project, no need to write any shader and code
2. Only support URP（Universal Renderer Pipeline）and LWRP (Light Weight Renderer Pipeline)
3. The script controls the shader to create dynamic effects, don’ t worry about it, like other shader effects, it can only be played in a loop
4. You can modify the attributes exposed by the shader and script to easily get the effect you want
5. Provide you with 9 different ice textures, 11 adjusted materials
6. Contains animation sound effects
7. Configurable shader properties
8. Support Android, IOS, Windows and other platforms
9. Complete documentation, clear demo
10. Cheap

How To Use：

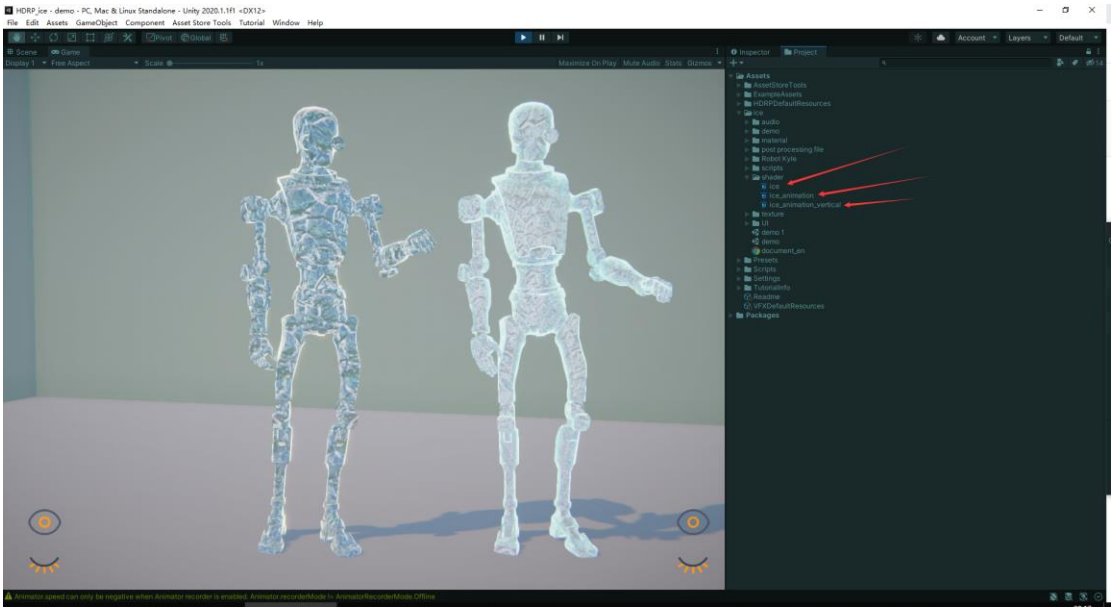
How To Run

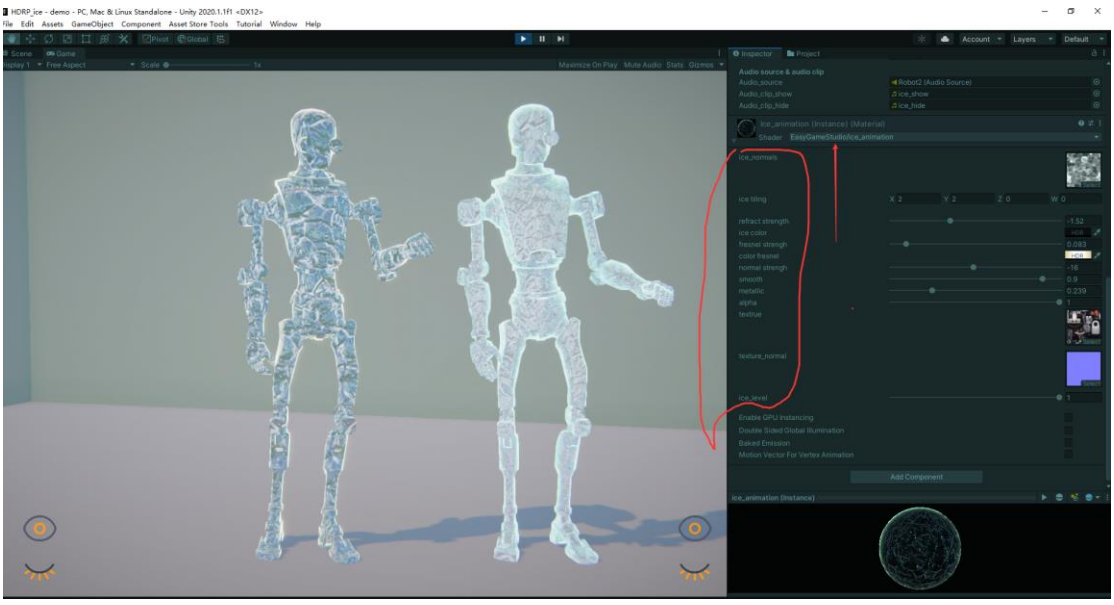
Find demo.unity in the resource package and click Run. The path address is shown in the figure below:



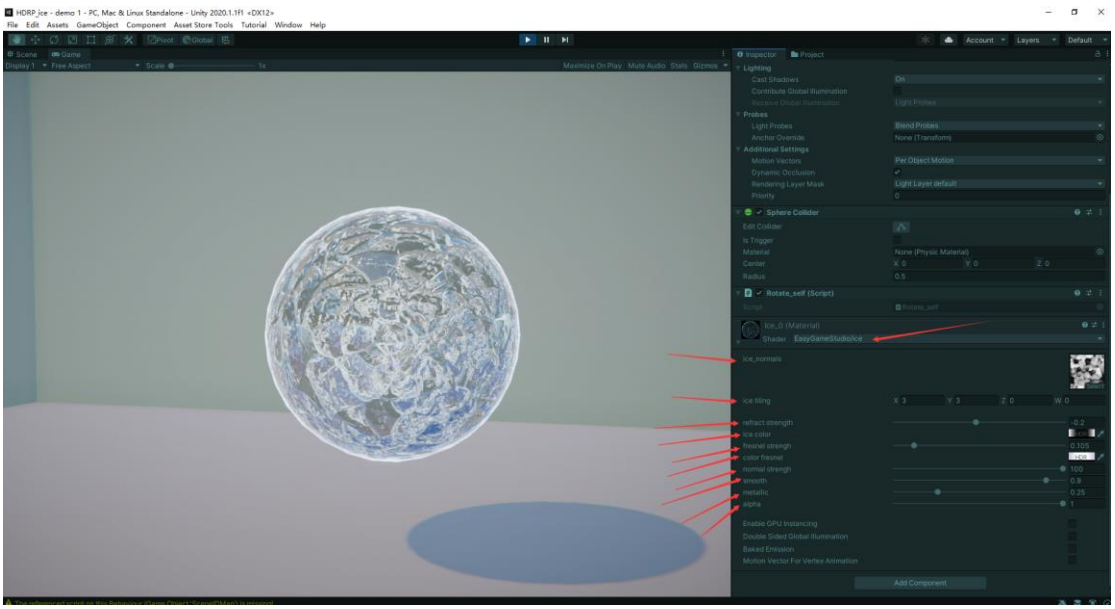
How to apply to your own game objects

1. This shader can only act on objects that contain "MeshRenderer" or "SkinnedMeshRenderer" components
2. Set the shader for the game object in the first step-"EasyGameStuio/ice_animation" or EasyGameStuio/ice_animation_vertical", as shown in the figure below:

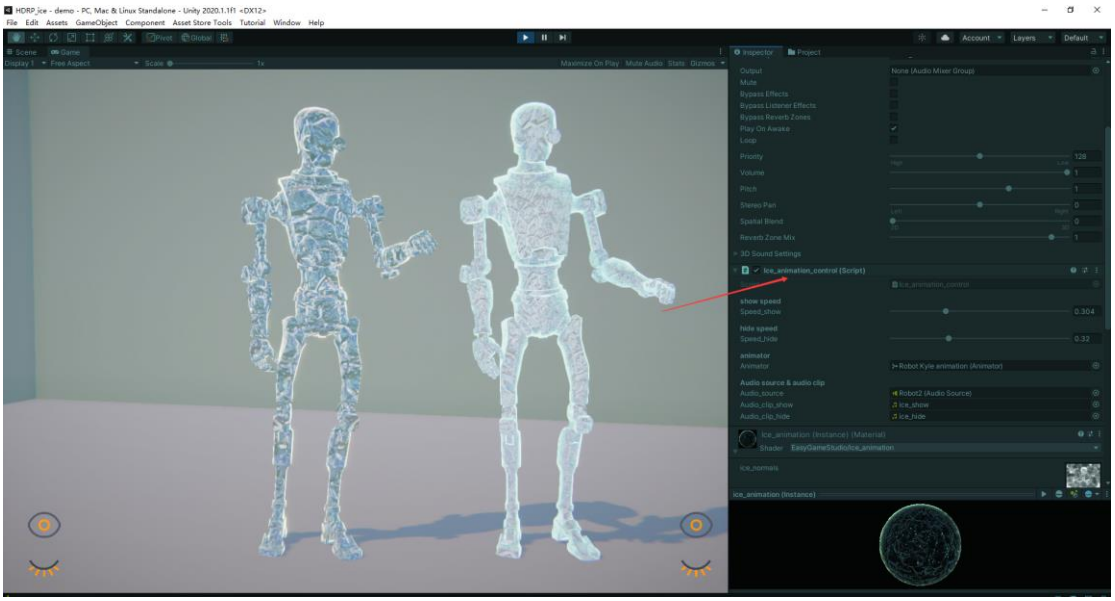


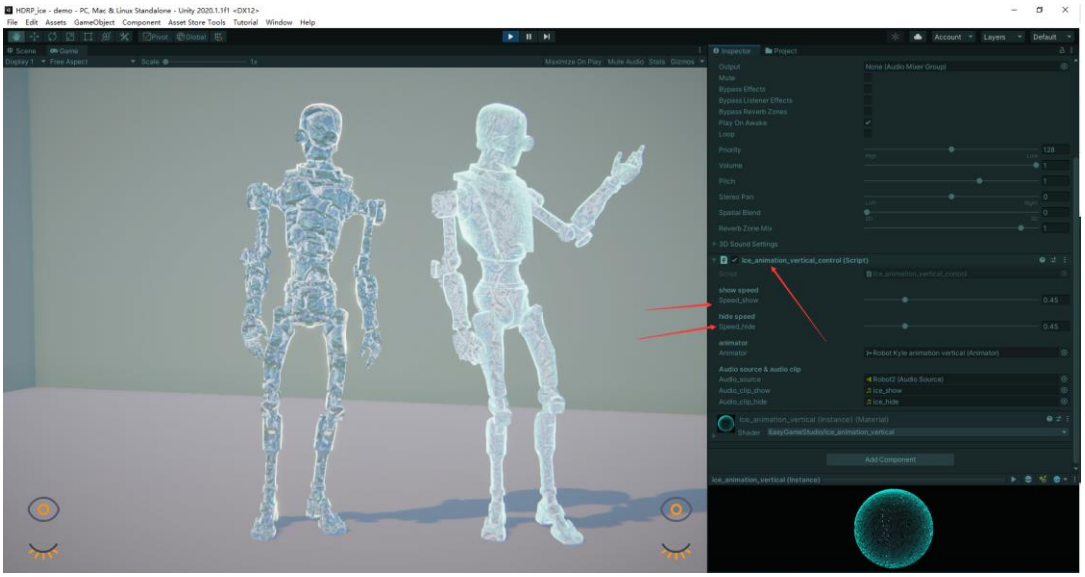


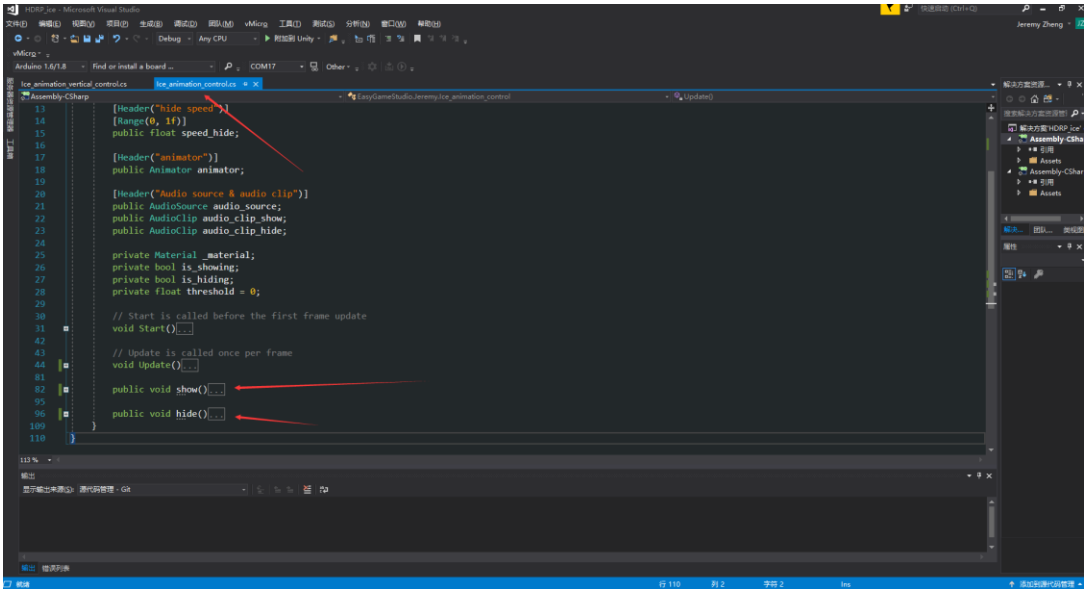
3. How to adjust the shader parameters to get the effect you want more: select the game object , expand the shader option, and try to modify the shader parameters, as shown in the figure below



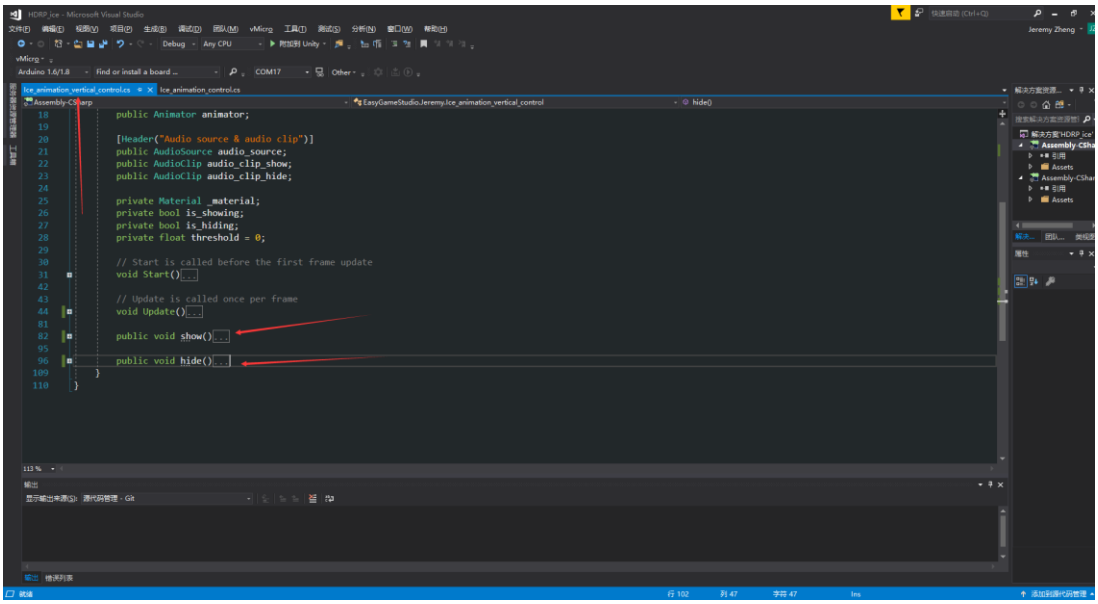
4. If you need to play animation display or hidden animation, you need to add a script to the above game object, and then call the "show" and "hide" functions in the script. The parameters exposed in the script can adjust the animation The playback speed, as shown in the figure below:







```
[Header("hide speed")]
[Range(0, 1f)]
public float speed_hide;

[Header("animator")]
public Animator animator;

[Header("Audio source & audio clip")]
public AudioSource audio_source;
public AudioClip audio_clip_show;
public AudioClip audio_clip_hide;

private Material _material;
private bool is_showing;
private bool is_hiding;
private float threshold = 0;

// Start is called before the first frame update
void Start()...

// Update is called once per frame
void Update()...

public void show()...

public void hide()...
}
```



```
public Animator animator;

[Header("Audio source & audio clip")]
public AudioSource audio_source;
public AudioClip audio_clip_show;
public AudioClip audio_clip_hide;

private Material _material;
private bool is_showing;
private bool is_hiding;
private float threshold = 0;

// Start is called before the first frame update
void Start()...

// Update is called once per frame
void Update()...

public void show()...

public void hide()...
}
```

PS: It needs to be used together with post processing, the post processing effect file that comes with the HDRP project is used in the demo

Contact Us:

If you have any questions or suggestions during use, please feel free to contact us

Email: sgzxzj13@163.com