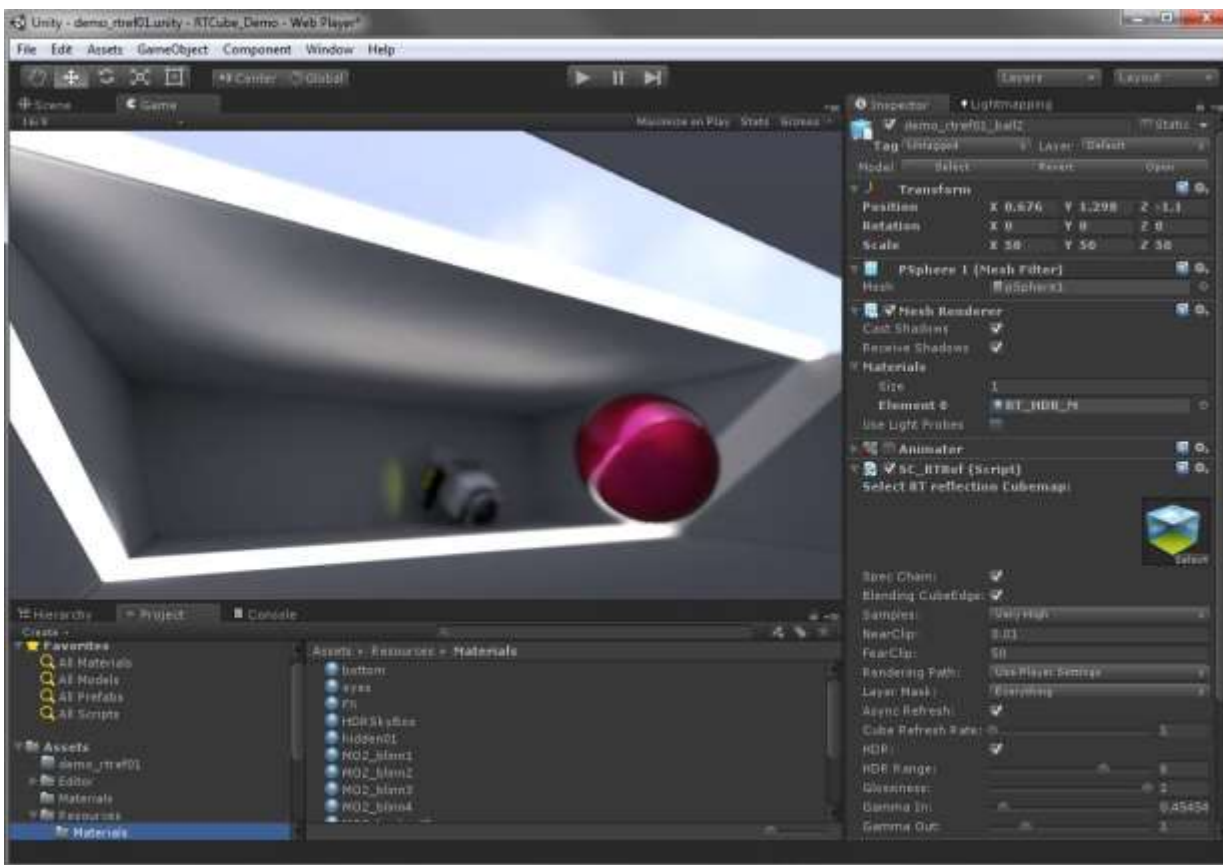# SkyCube RT 1.0 User Manual

Introduction:

SkyCube RT: Real-time IBL HDR Cubemap Rendering Tool & Shader Library is a tool with shader library for in-game real-time HDR/LDR image base rendering solution. It renders cubemap with specular chain in real-time and cost less than standard cubemap rendering. It is a perfect solution for the case when you want use the same environment map for different glossiness objects/materials.

It orients Unity4.x and compatible Unity5.x.

Please note, this tool has already been included in Skycube 3.5. If you had already purchased Skycube please ask refund and deleted Skycube RT from your project.
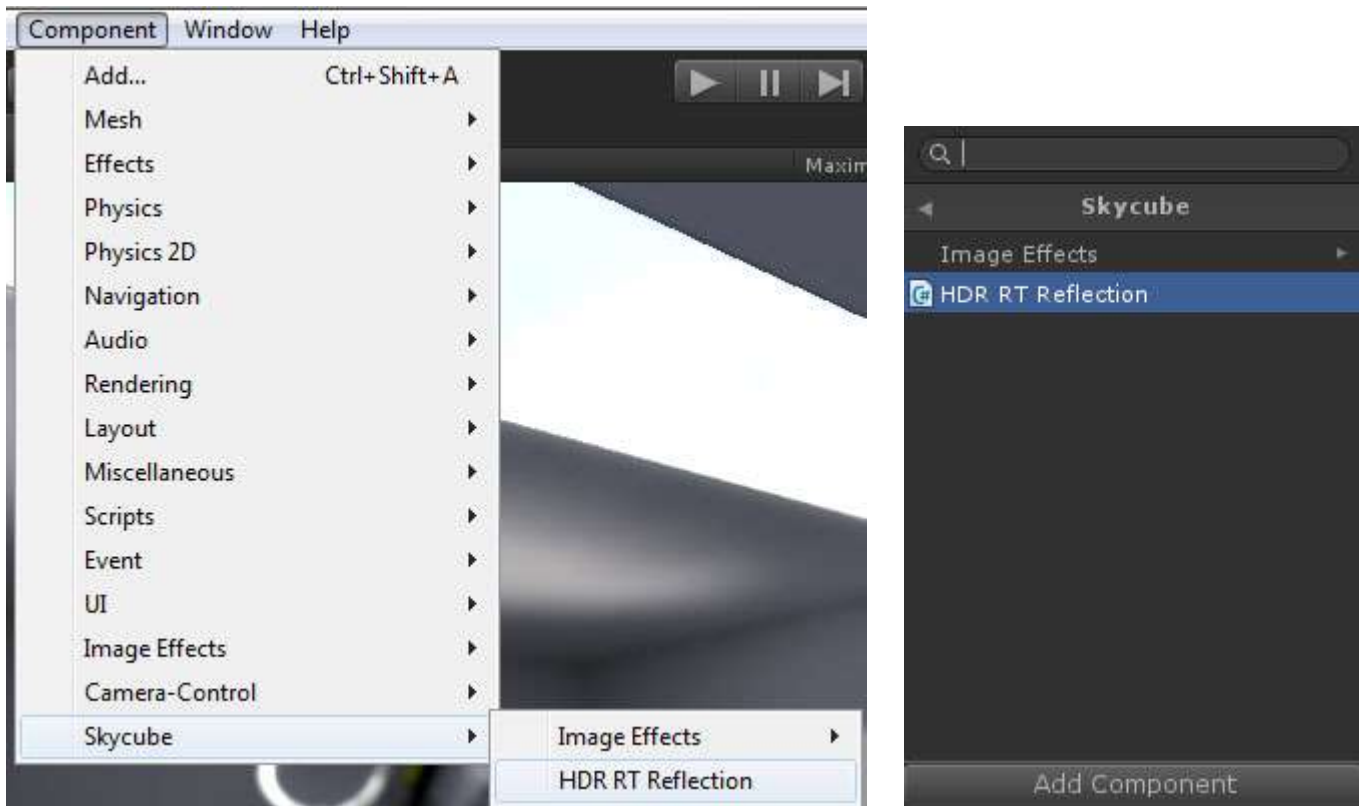

Usually standard rendering will sample environment base on each object, because the object in different position or has different glossiness in material. That means more objects will have more sample times and it will cost more also. And in the most case we only need to see real-time high gloss reflection on the object which close to camera and roughness reflection is not position sensitive. So why not share cubemap to all objects which also a kind of common way in current console game development.

Base on this idea, the tool will render cubemap with specular chain at the same time, it means you can access different roughness reflections in one cubemap and also can use them in one object just, for example, like multi-layer reflection material.

The specular chain base on physical glossiness algorithm, it more expensive than Unity5 which to use Gauss blur to simulate roughness reflection that less cost and less accuracy.
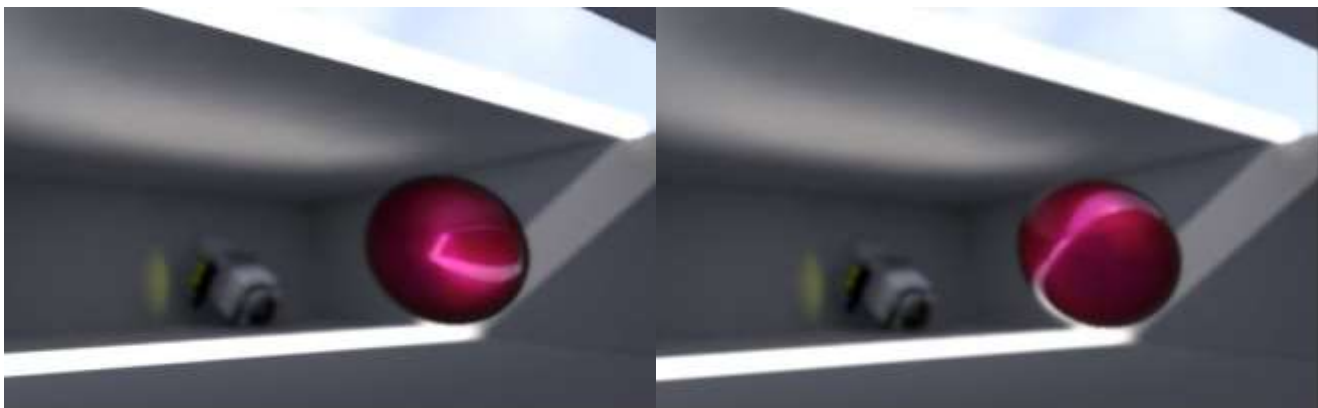
How to use it:

Simply select a Game object which you want to has reflection and then just adds a Component "HDR RT Reflection" from Component menu or Inspector window like below:

Then you will see SC_RTRef component in your game object.

This component will work in both editor and build modes, but in editor it will only work correctly just when you run your scene. You maybe will see cubemap not be updated when you move your object in editor scene window sometimes.

Before you test it, don't forget to add a target cubemap in the tool:

The cubemap can be any cubemap which should have 6 images (any images just don't use empty cubemap), the tool will not change those images (it will not save images to disk, it just save cubemap).
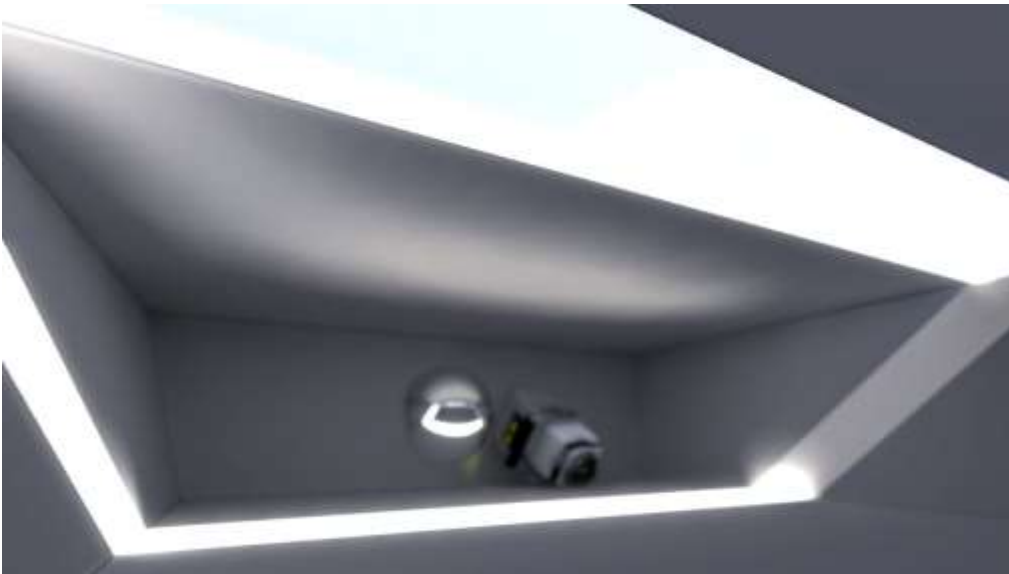
Please make sure you turn on **Mipmaps** and **Readable** in this cubemap, and confirm the **Face size**. The tool will work base on cubemap size, if you setup it bigger you will have batter quality but cost more in performance also.

Now, you can run your scene and check the result. But if you want the best result we still need to do some setup.

Before we deep in it you can check a Demo which can show you how it works and result:

[Demo: HDR Real-time Reflection with glossiness chain.](#)

In this demo, you can see real-time HDR reflection result with different IBL materials. The ball will keep moving and you can use W,A,S,D to move camera and also use mouse to navigate view direction to observe it. In this demo the cubemap updates one face in one frame and calculate specular chain once after all 6 faces updated.

Please note, if you want to open it in Chrome, you maybe need to enable NPAPI in chrome by:
chrome://flags/#enable-npapi
If you cannot view this demo in your browser you can also check the demo video: http://youtu.be/7GMmelHbB-U

Different exposure (use – and + key to control):

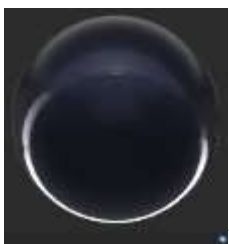Difference Material (use 1,2,3,4,5 key to select different materials):

Mirror

Mirror with different roughness
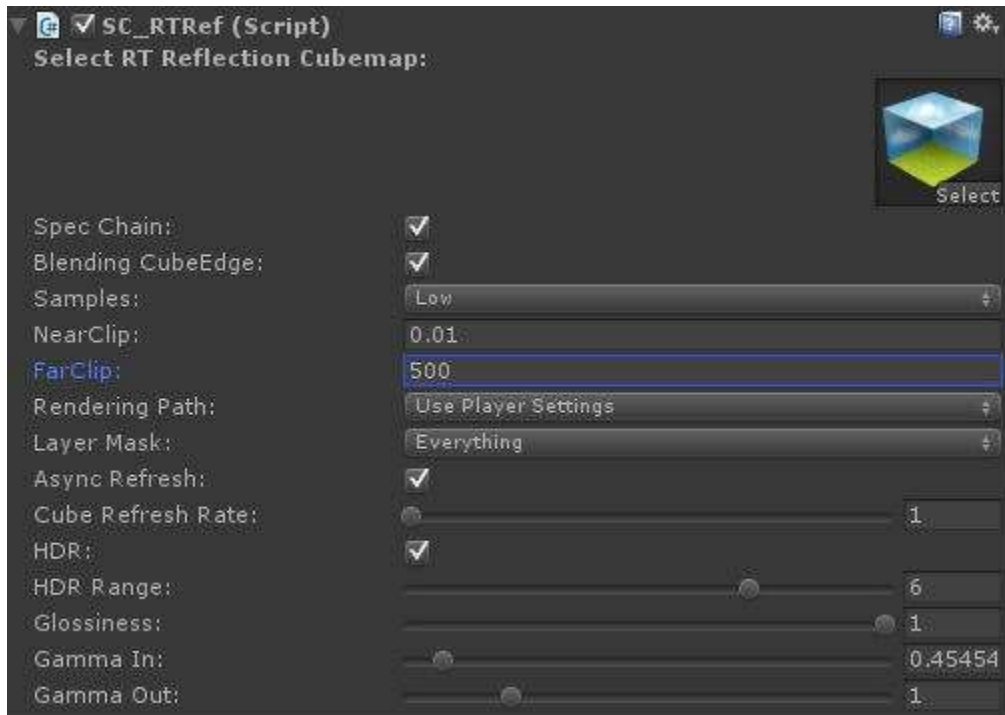
Different reflectance

Multi-Layer reflection (Plastic)

Multi-Layer reflection with color variation (Car Paint)

Settings:

OK, let's take a look the UI of Skycube RT:

- **Select RT reflection Cubemap:** Choose a cubemap for rendering.
- **Spec Chain:** Generate specular glossiness chain inside cubemap's mipmap The filter will sample each mipmap with glossiness reduction based on your Glossiness setting. The last mipmap's glossiness will always be 0.
- **Blending CubeEdge:** Since Unity Cubemap's 6 faces are not in the one texture and cannot be sampled continuous, in small size it will appear seams between each face. To avoid it you can turn on Blending cubemap edge option, but it will cost a lot performance (This is a function which inside Unity).
- **Samples:** Choose sample quality.
- **Near Clip:** The near camera clip.
- **Far Clip:** The far camera clip.
- **Rendering Path:** Selecting which rendering path will be used in Skycube. Usually it should be as same as your project, so just keep default "Use Player Settings" and it will use the same setting as your project.
- **Layer Mask:** Selecting which Layer you want rendering.
- **Async Refresh:** Asynchronization refresh. Turn on it, the glossiness sample will calculate only after 6 faces updated. You better turn on it when your Cubemap refresh rate less than 6 and glossiness less than 1, otherwise you will see flicker.
- **Cube Refresh Rate:** How many cubemap faces will be updated in one frame. It similar as Time Slicing in Unity5 reflection Probe but smoother than reflection Probe, since it updates cubemap much more frequency.
- **HDR:** Turn on or off HDR rendering. Do not forget turn on HDR in camera also.
- **HDR Range:** The HDR Lighting range in the scene (it will only available when HDR on), it is same as the lighting maximum intensity is 8 in Unity. Since we encode 32bits HDR to 8Bits LDR texture, the color gradation range of LDR texture is 256 and $2^8 = 256$, so the number 8 is the maximum intensity.
- **Glossiness:** The glossiness control, the value is similar Blinn lighting mode, the higher value the finer glossiness. The value range is from 0 to 1. When the Glossiness from 0.1 to 0.2 will get close result as irradiance.
- **Gamma In:** Gamma tweak for rendering scene. It will update to 0.454545 when you turn on HDR.
- **Gamma Out:** Gamma tweak for cubemap output.

Build in game:

If you want to put this tool in your game, you need to move shader folder under your Resources folder, since there are some hidden shaders will be needed when script running and the compile system will not including them since there is no material uses them. So you need manual move them to your Resources folder to let compile system include them before build game.
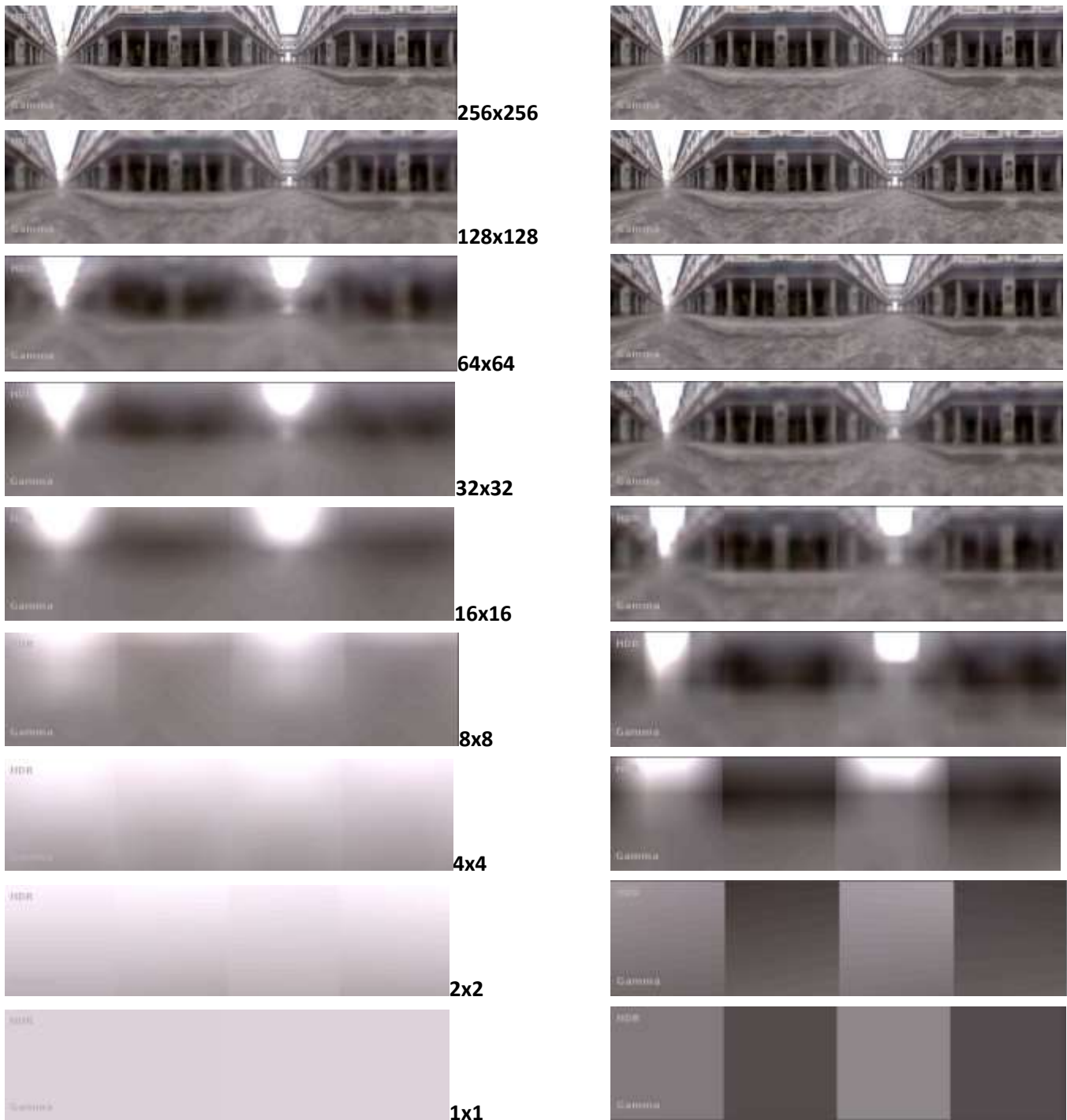
Specular Chain:

Specular Chain is a technique to pre-bake different glossiness reflection inside mipmaps and use mipmap offset function in shader to access them for blur reflection that avoid to re-sample cubemap in real-time rendering. Before this

technique some people tried to use default mipmap for low glossy surface reflection but it totally not physical accurate and with very obvious seam. There is a HDR cubemap comparison in below:

Left: Specular Glossiness Chain, Right: Default Mipmap

256x256

128x128

64x64

32x32

16x16

8x8

4x4

2x2

1x1

Usually we use Mipmap level from 0 to the third from the last one(4x4), because under 4x4 pixels in each cubemap face the pixels too small to get enough information. But in some case you still can use 2x2 for some special effect like Velvet Fresnel.

Since the cubemap in Unity always split in 6 textures, so in some case you still see little seam between cube faces in high contrast content. Usually you can solve this issue by higher sample.

Any question about this tool please contact me in mail: support@atomsdev.com