



Dual Emission Shader (3D)

v1.0
Documentation

Contents

1	Introduction	2
2	Documentation	3
2.1	Universal Render Pipeline (URP)	3
2.1.1	Universal Renderer	3
3	Support	7

Section 1

Introduction

The Dual Emission shader (3D) is a reconstructed URP Lit shader that supports two independent emission sources. This allows you to use two different emission maps, two different emission colors, and two different emission intensities on the same 3D mesh – for example, fiery orange cracks and cold cyan runes glowing on the same weapon, without splitting the mesh into several materials.

In addition to the regular HDR emission color controls, each emission has its own extra intensity option. This makes it possible to increase each emission's strength beyond the usual HDR color intensity range.

WARNING

This shader supports only Universal Render Pipeline (URP) on Unity 6.0+. Built-in Render Pipeline (BIR) and High-Definition Render Pipeline (HDRP) are not supported.

The shader preserves the familiar URP Lit material options, such as albedo, alpha cutoff, normal maps, metallic settings, and ambient occlusion.

Feel free to check out the demo scenes included in the asset for inspiration.

Section 2

Documentation

2.1 Universal Render Pipeline (URP)

2.1.1 Universal Renderer

The Universal Renderer is the default URP Renderer when creating a 3D project.

Setup

The following paragraph presents the steps necessary to make the Dual Emission (3D) shader work in your project. Visit the following link for image-aided instructions:

<https://docs.project-gamedev.com/dual-emission-shader-3d/setup.html>

The Dual Emission (3D) shader is, in principle, the same as the regular URP Lit shader you would use in your Unity project, only reconstructed and expanded. The setup is thus the same as with any other lit shader:

- Create a new material for your object(s).
- Select the material; then, select **Project G.A.M.E.D.E.V./Essential 3D Shaders/Dual Emission Shader (3D)** as the shader source for this material.
- Assign the new material to your object(s).

That's it! The shader is now active. To make your mesh glow, assign a **Primary Emission Map** and/or a **Secondary Emission Map**, and pick the respective emission colors – both emission colors are black (i.e., no emission) by default. Feel free to play around with the material properties to find a dual emission setup that you are happy with.

Keep in mind that certain post-processing settings (most notably **Bloom** and **Tonemapping**) will have a profound effect on the emissive parts of your mesh. The demo scenes use the post-processing profile included in the **Post-Processing Profiles** folder, which you can use as a starting point for your own scenes.

i NOTE

Both emission maps are sampled with the same UV coordinates as the albedo texture. The easiest way to create them is to start from your albedo texture (or a UV layout exported from your modelling software), paint the areas that should glow in white, and fill everything else with black – one map per emission color.

Shader Options

The material properties are split into two categories in the Inspector – **Main Surface** and **Emission** – and are listed below in the same order.

Albedo Texture The main (albedo/color) texture used by the shader.

Albedo Color An additional color multiplied with the one already present in the albedo texture. Use this to further customize the look of your mesh.

Alpha Cutoff This is a threshold value – all parts of the texture whose alpha value is less than the alpha cutoff will not be rendered.

Ambient Occlusion A value between 0 and 1 that controls how much ambient (indirect) light reaches the surface. At **1**, the surface receives the full ambient lighting; lower values darken it, as if the surface were occluded. Unlike in the standard URP Lit shader, this is a single value for the whole material rather than a texture.

Metallic Map The standard metallic map option in the default URP Lit shader. The metallic map is a black-and-white texture that determines the surface type of the object. It is used to make certain parts of the mesh reflect light like a metal would (metallic), and others like sand would (smooth). The shader reads the metallic value from the texture's red channel, so disable **sRGB (Color Texture)** in the texture's import settings. If no metallic map is assigned, the entire surface is treated as fully metallic.

Metallic This value determines how “metallic” the surface should be, i.e., whether it should reflect light like a metal or a wet surface, or should absorb it like sand or another rough, non-metallic surface.

Normal Map The standard normal map option in the default URP Lit shader. The normal map is used to simulate finer surface detail by containing information on how the light should be reflected by the surface. Make sure the texture's **Texture Type** is set to **Normal map** in its import settings.

Normal Strength This value is multiplied with the normal map and is used to make the effect stronger or weaker. A value of **1** applies the normal map at its original strength, and **0** disables it. A negative value will mirror the 3D-effect (i.e., protruding surfaces will now be concave, and vice versa).

Primary Emission Map A black-and-white map that determines which parts of the mesh will emit light in the primary emission color. The white pixels of the texture are areas where the emission will have an effect, and the black pixels are the areas with no emission.

Primary Emission Color The emission color for the primary emission. As it is an HDR color, an RGB-value as well as an intensity can be specified. Keep in mind that **different post-processing effects (most notably bloom)** will have a **profound** effect on the look of your emission.

Primary Emission Extra Intensity Since the default HDR color has a limited intensity (-10 to 10), this shader offers an additional intensity option that allows you to make your primary emission even stronger. A value of **1** (the default) leaves the emission unchanged, values above 1 strengthen it, and values below 1 weaken it. The intensity grows exponentially, so small steps (e.g., from 1.1 to 1.5) already make a clear difference. Consult with 2.1.1 on how this option interacts with the viewing angle.

Secondary Emission Map An additional emission map that allows you to make different areas of the mesh emit light in a different color and with a different intensity than the primary emission. Has the same properties as the primary emission map option.

Secondary Emission Color The emission color for the secondary emission. Has the same properties as the primary emission color option.

Secondary Emission Extra Intensity The additional intensity applied to the secondary emission. Has the same properties as the primary emission extra intensity option.

API

Visit <https://docs.project-gamedev.com/dual-emission-shader-3d/api.html> for the Dual Emission Shader (3D) API documentation. This API allows you to modify all shader values in-code, from your own scripts, if necessary.

Advanced

i NOTE

The two emissions are added on top of each other. Where the primary and the secondary emission maps overlap, their colors mix additively and become brighter (e.g., red and blue produce a bright magenta). Keep the white areas of the two maps apart if you want two clearly separated colors.

i NOTE

The RGB values of each emission map are multiplied with the respective emission color, so gray pixels produce a weaker emission, and colored pixels tint the emission color. The alpha channel of the emission maps is ignored – transparent pixels whose color is not black will still glow, so paint all non-emissive areas in opaque black.

i NOTE

The extra intensity options are weighted by the viewing angle. Surfaces facing the camera receive the full effect, while surfaces seen edge-on (along the silhouette of the mesh) remain almost unchanged. As a result, values above 1 make the emission brightest where the surface faces the camera, while values below 1 fade the camera-facing emission first and keep it along the silhouette – at 0, the emission turns into a rim glow.

Section 3

Support

In case you experience any problems with the shader, can't manage to set it up properly, or would simply like to propose an improvement, don't hesitate to contact us at contact@project-gamedev.com, or fill out the form at <https://project-gamedev.com/contact.html>.

If you are satisfied with this product, please consider leaving a review on the Unity Asset Store page (<https://assetstore.unity.com/packages/slug/412498>).

Best of luck with your game development journey!