



VELVARA FLAMES
CUSTOMIZABLE STYLIZED SHADER



Velvara Flames Documentation

Thank you for your purchase! Welcome to the Velvara Flames Pack. This is a series of stylized shaders meant to help you give a very distinctive look to your game.

This package contains the main Flame shader, as well as other support shaders for Heat distortion, Burning objects, Glowing embers and a Simple Billboard. This guide will only go in-depth explaining the Flame Shader. Support shaders are meant to be used with their default settings.

***Note: Heat Distortion shader is not currently compatible with mobile.**

You will also find +15 game ready prefabs and prefabs variants so you can jump start your game right away. This package also includes “light” versions of these prefabs optimized for mobile.

Even though you can use any of the assets provided right out of the box, the main value of this shader is the possibility to make your very own assets and flame meshes with very little image editing or 3D software knowledge!

Because of how versatile the flame shader is (you can use to make candles, campfires, torches or even massive fire tornadoes), there is no standard audio files that can fit every possibility. So you will have to find or create the audio files that fit your flame creations better.

This documentation will cover the **First Steps**, **Flame Shader Guide**, a **Custom Color Guide** and **Custom Mesh Creation** tips.

If you have any questions or would just like to send a message, please contact me at:

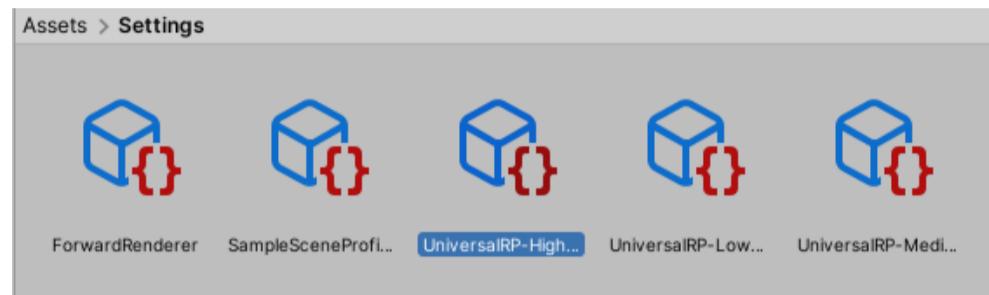
velvara.art@gmail.com

**I currently live in a place with unstable internet connection, so it may take anywhere from 1 to 48 hours for me to reply an email. Be sure that I will do my best to reply your email as soon as possible. Thank you for understanding.*

First Steps

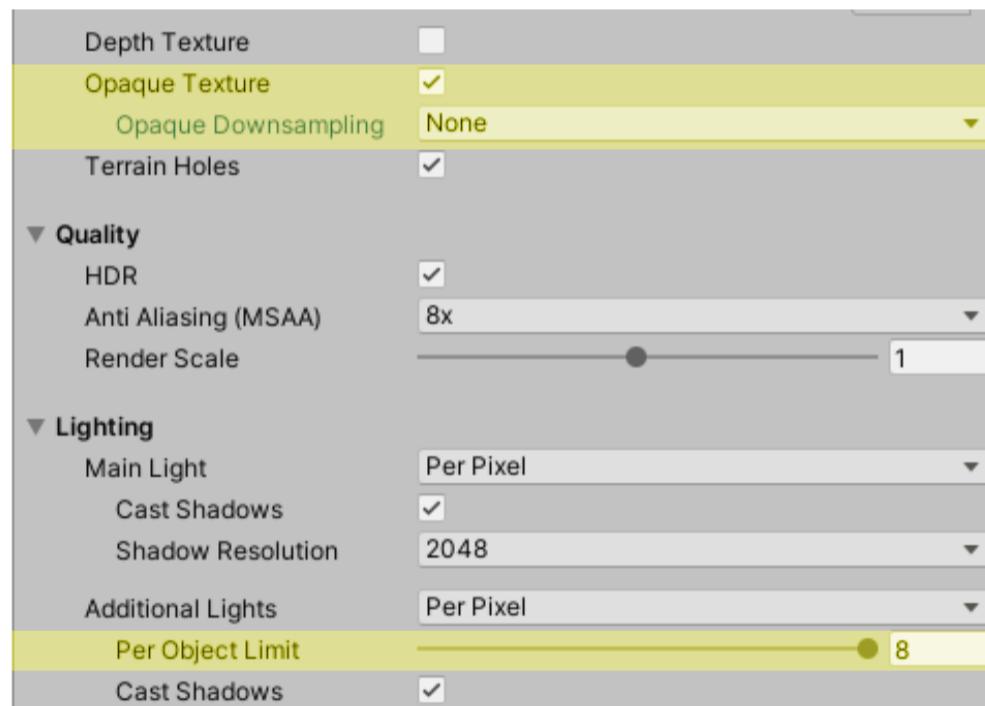
The very first step to take to make some of the shaders compatible with your project is to enable Opaque Textures in both your URP Asset and any cameras in your game.

The URP Asset settings can be accessed in **Assets > Settings**. Here you'll find Unity's built-in URP Assets. To make sure which one you're currently using, you can go to **Edit > Project Settings**, then select the "**Graphics**" tab to see which URP Asset is your project using currently.

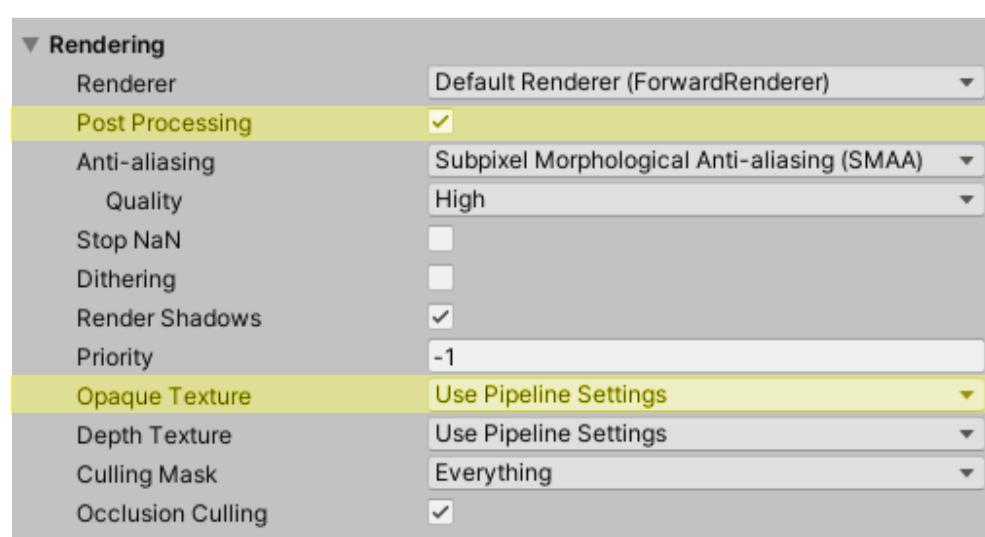


In your URP Asset settings, be sure to check the "**Opaque Texture**" checkbox. Under **Opaque Downsampling**, choose the option that suits your game better (Choosing "None" will not downsample, which will look the best at a performance cost)

***Optional:** This step is not required for any component of the package to work. However, if you want to experience the sample scenes as they were meant to, go into the "Lighting" group in your URP Assets and raise the **Per Object Limit** to **8**.



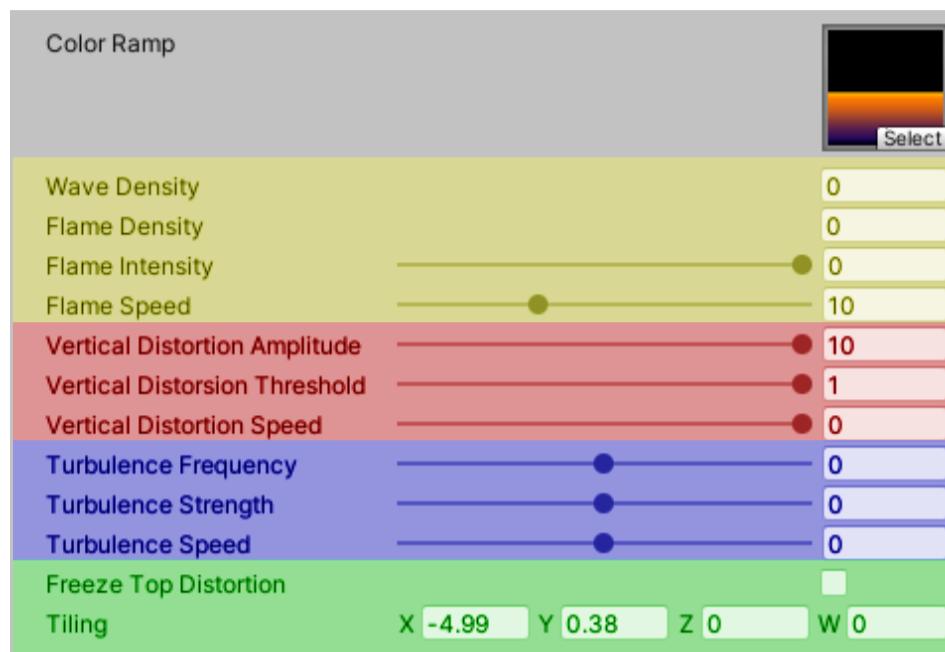
Just to make sure everything works as intended, go to your camera settings and check that "**Opaque Texture**" is either set to either "**On**" or "**Use Pipeline Settings**". Optionally, you can also check the "**Post Processing**" checkbox to experience the example scenes as they were meant to.



Flame Shader Guide

To tweak or create new flame materials, it's important to understand what each shader parameter does. So let's start by dividing these in 4 groups:

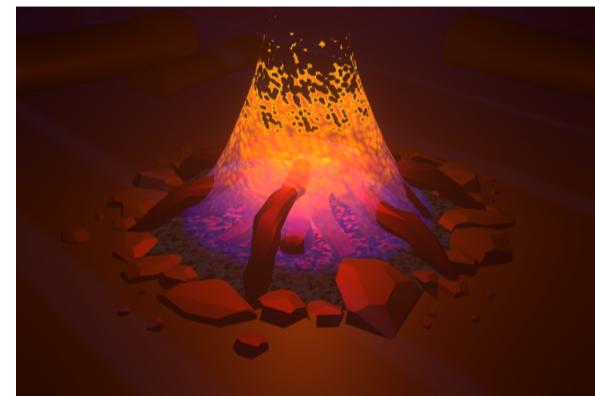
Texture Dynamics, **Vertical Distortion**, **Turbulence** and **Others**.



Color Ramp: This is the image texture that will define the flame color. You can try out 5 different pre-built color ramps in **FlamePack > Materials > Color Ramps**

Texture Dynamics: This section deals with how the color ramp will be rendered throughout the mesh.

Wave/Flame Density: The fire effect is created by using a noise to define large tongues of fire (waves), and an “inner noise” that is applied to the wave to give them more detail.



The first image has Waves with no Flame Density. The second image has Flame Density applied to the waves. Notice how you can kind of see the pattern left by the waves?

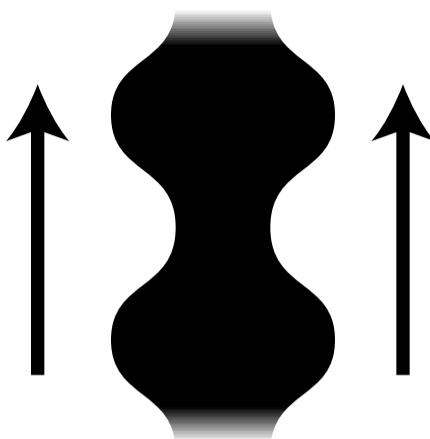
**These values were exaggerated for demonstration purposes. Balancing these values will yield no discernible pattern.*

Flame Intensity/Speed: Flame intensity defines how much will the noise from the Wave/Flame Density affect the Color Ramp. At 0, you'll see the Color Ramp applied to the mesh as it is. Flame Speed will define how fast the noise travels through the Color Ramp.



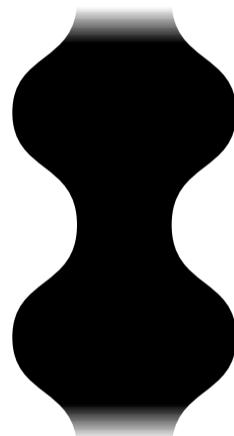
Flame Shader Guide

Vertical Distortion: The flame mesh will have its vertex displaced by waves traveling upwards. This section covers these distortion's settings.

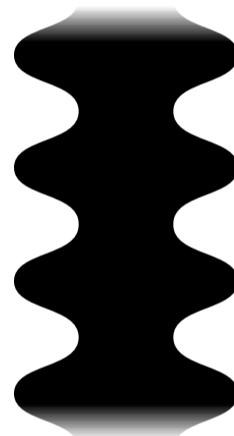


This is a representation of how the mesh's vertex will be displaced as the waves travel upwards.

Vertical Distortion Amplitude: This will define how many waves will travel through the mesh. Higher values will increase the amount of waves that will displace the mesh vertices.



Lower values



Higher values

Vertical Distortion Threshold: This will define how much displacement will the waves apply to the mesh. A value of 0 means the waves won't displace the vertices at all.



The first image has a Threshold of 0, so the mesh is not distorted. the second image is fully distorted by the waves at a value of 1.

Vertical Distortion Speed: This value defines the speed at which the waves travel upwards through the mesh.

Flame Shader Guide

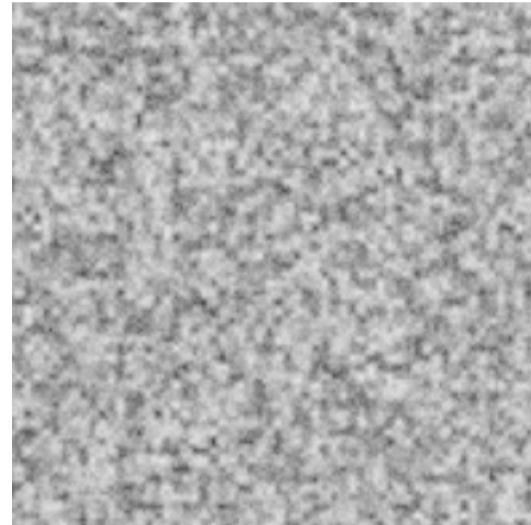
Turbulence: Vertical distortion is very regular on its own. So turbulence applies another layer of random distortion to the vertices that break the wave monotony and gives flames a more organic ebb and flow.

To this effect, a noise is added to the vertical distortion. This section covers the settings of this noise.

Turbulence Frequency: This will define the size of the noise. Black and white areas define how much a vertex will get displaced. Lower values will mean less jittery, more peaceful ebb and flow. Higher values will make for a more turbulent, windy flame.



Lower values will displace larger groups of vertices.



Higher values will displace smaller groups of vertices more erratically.

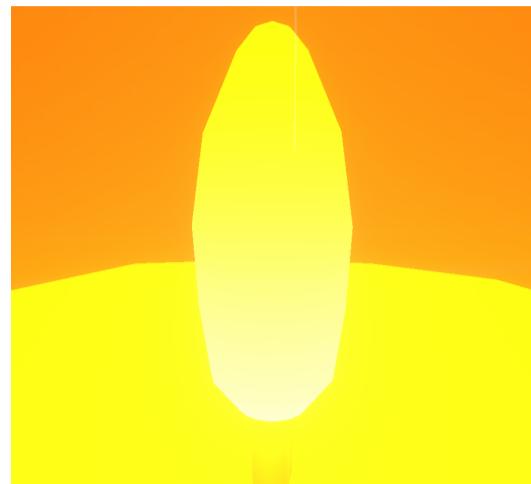
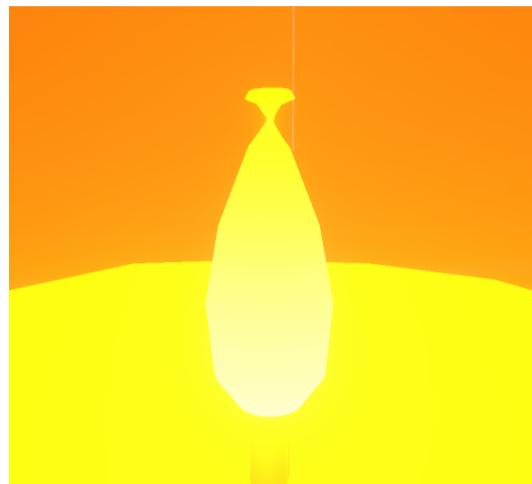
Turbulence Strength: This will define how far will the vertices get displaced by the noise. A value of 0 will prevent the turbulence noise from displacing any vertices.

*Note that vertices will still get displaced by Vertical Distortion if there is any already on effect.

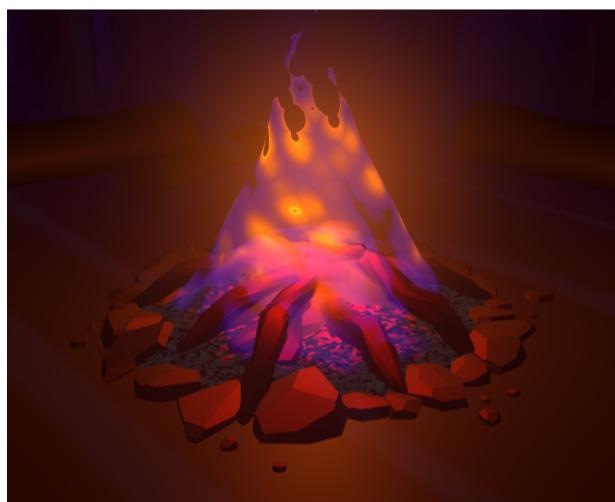
Turbulence Speed: This will define how fast will the noise travel through the mesh.

Flame Shader Guide

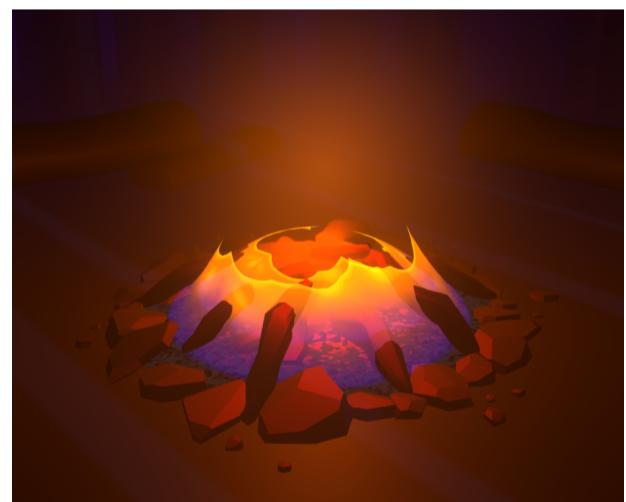
Freeze Top Distortion: If your mesh is closed on its top side, checking this will nullify any distortion applied to the top so that vertices don't cross each other. Useful for candles.



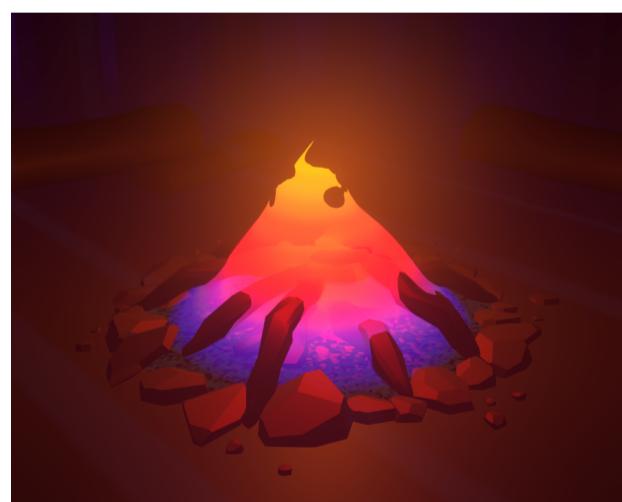
Tiling: Tweaking the Y tiling value is not recommended until you've set up the rest of the parameters. Higher values will make for taller, less intense flames (might introduce unwanted transparent areas). Lower values will make the flames shorter (and will decrease the effect of the Texture Dynamics group). Used sparingly with Flame Intensity can yield cartoony, flat colored flames.



Y tiling at 0.5



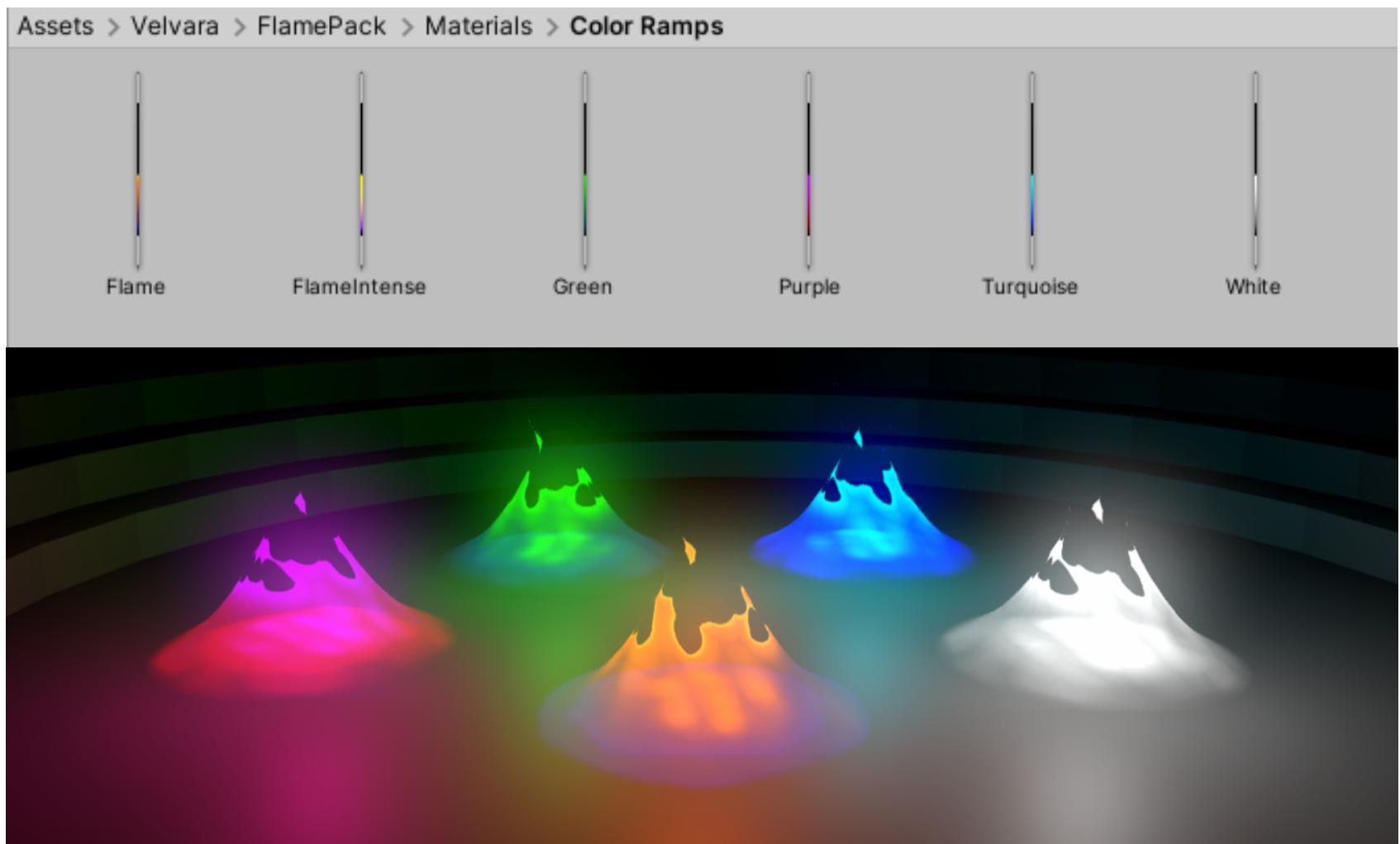
Y tiling at 2



Tiling paired up with Flame Intensity 0 to create a flat look.

Custom Color Guide

This Shader Pack comes with 5 built-in color ramps for you to use. However, you might need another color for your flame FX, and this guide will help you through the process.



Each color ramp is a 1x1024 px texture. In order to create your own, open your preferred image/vector editing software and create a 1024x1024 document.

Then, import any of the color ramps included in the Flame Pack and stretch it horizontally (it's not necessary, but will allow you to have a better view of the colors you are using)

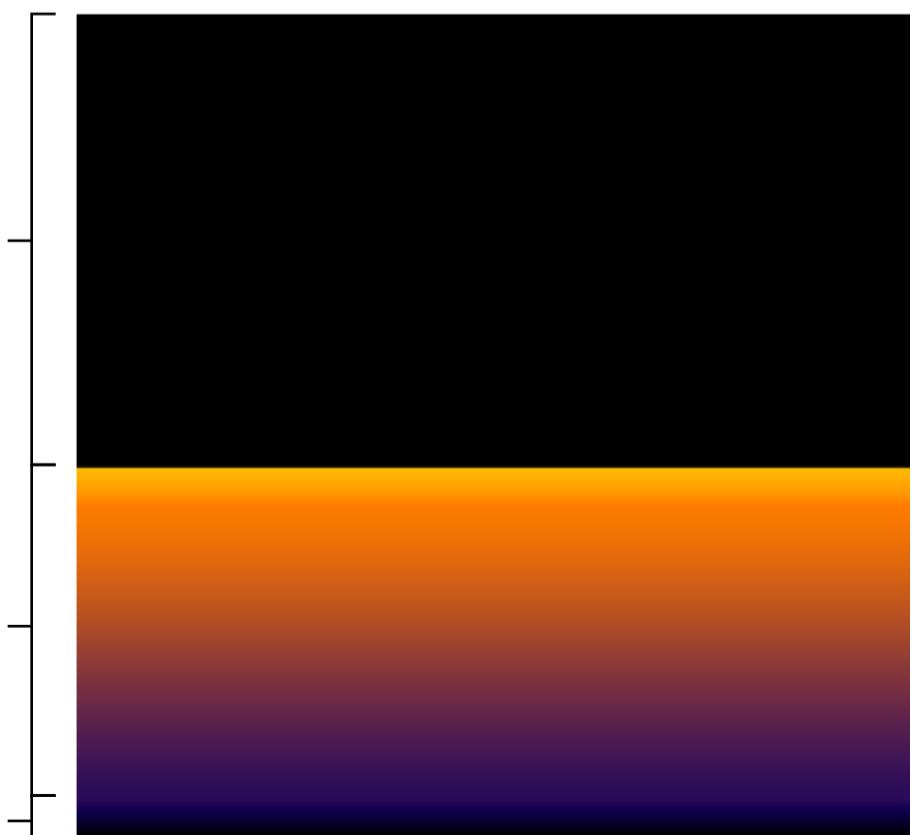
Now you're ready to create your own custom gradient! Remember to save it back as a 1x1024 PNG for reduced file size.

There are just a couple of things to bear in mind when creating your own Color Ramp:

This area should always remain black. It's there to provide space for the flames to reach taller.

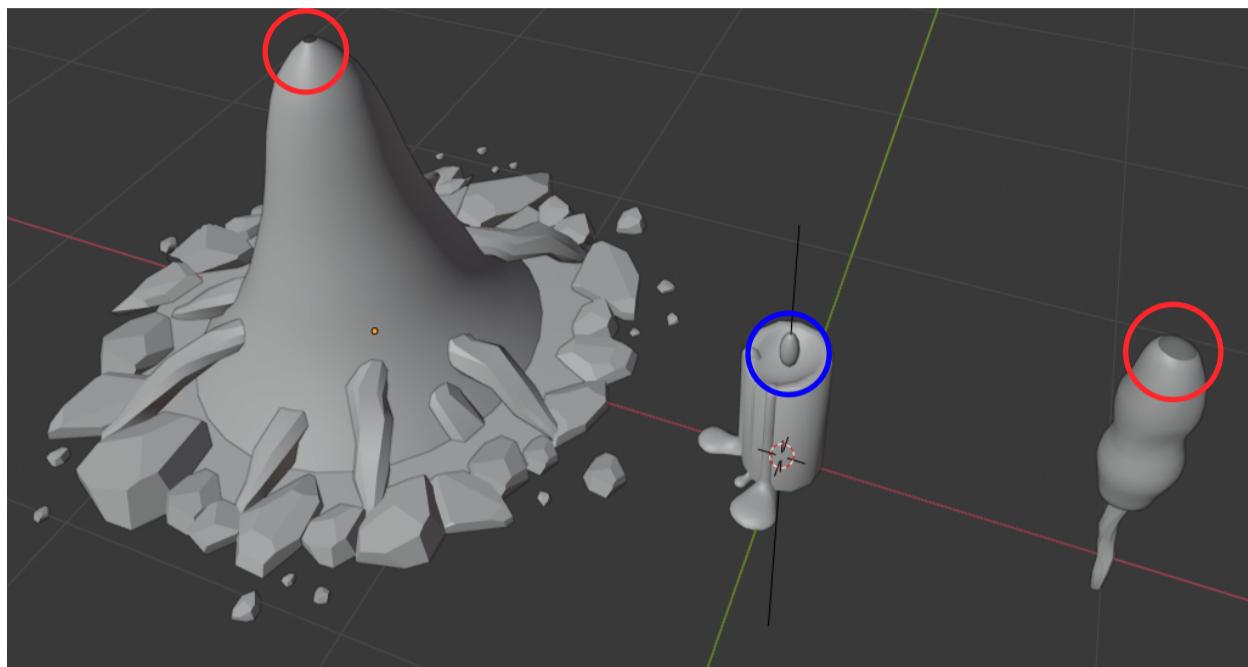
Unleash your creativity here! Gradients that end in darker tones are recommended.

Fading to black at the end will make flames start more naturally as it makes the base of the mesh invisible



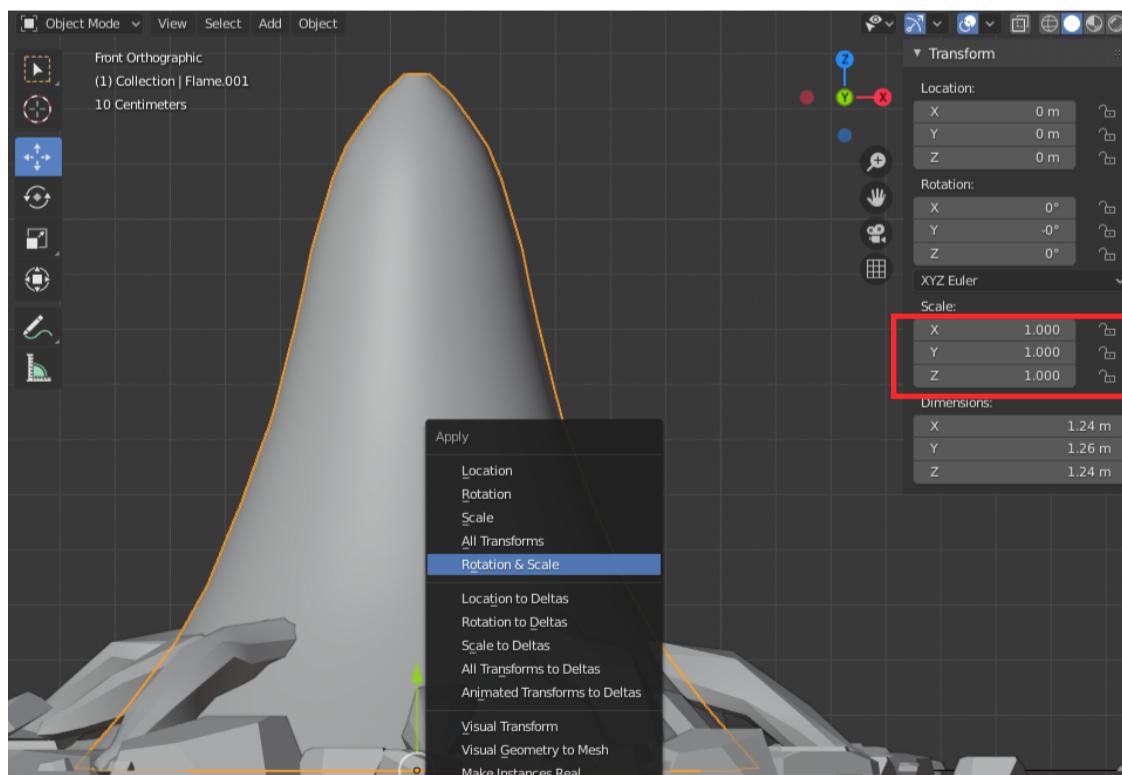
Custom Mesh Creation Tips

Even if you are a beginner at 3D modelling, creating your own custom flame mesh is very easy! These examples are made in Blender, but you can use your preferred 3D software tools as well.



Models: There is no real guide as to how to make a flame mesh, as it'll depend entirely on what you're making the mesh for. I would only suggest that you import the meshes provided in this pack to use as a base reference when making your own. However, there are a couple of tips that might help you in the process, regardless of the model you're making.

Always use **open ended meshes** at the top for wild flames (like the torch and campfire), and only use **closed meshes** for flames that will have very slight or almost no vertex displacement, such as candles. Wider openings are less likely to make vertices cross each other during displacement, which is hard to avoid even when checking the "Freeze Top Displacement" box.



It is also very important that you freeze the object's position and scale, so regardless of its size, it's always at a scale of 1. Failing to do this will result in the shader not properly displacing vertices, or seemingly not displacing them at all regardless of the settings you use.

Custom Mesh Creation Tips

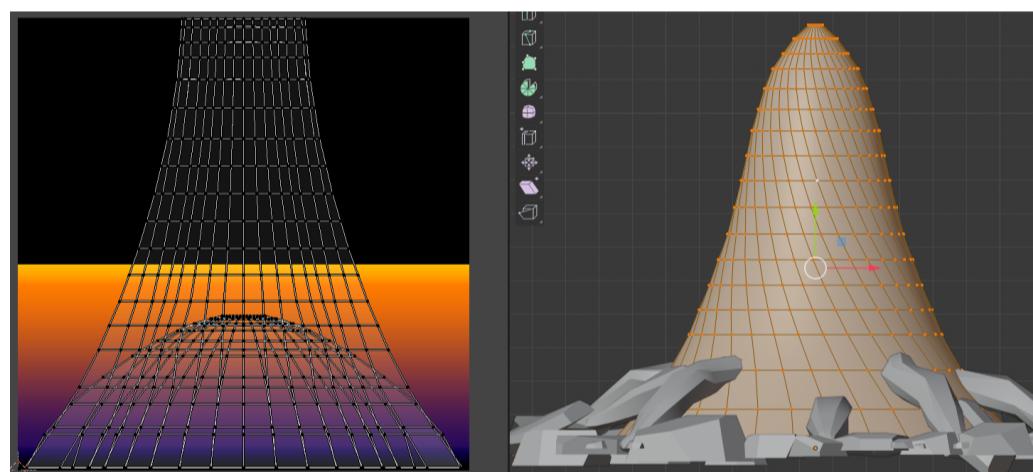
Unwrapping: The Flame shader relies on vertical UV scrolling, so it's important to unwrap in a way that places the base of the mesh at the bottom.

Currently, the noise used for the shader is not tileable, so unwrapping by projecting from an orthographic view will maintain continuity around the mesh.

The resulting mirroring can be mitigated by evenly spacing out UV vertices at both borders of the unwrapped mesh.



Another trick to hide mirroring is to unwrap the mesh first, and then twist the mesh horizontally a little bit. This will cause the mirrored areas to flow diagonally, making it look far less obvious. It also gives the flame a cool simulated windy twirl look!



Exporting from Blender: If you are using Blender, it's important to check the "EXPERIMENTAL! Apply Transform" checkbox when exporting an FBX.

This is because Unity will import a blender FBX and rotate it, which will mess up what the mesh considers "up". This will cause sideways displacement or even no vertex displacement at all.



So... that's it for the documentation! Now you're ready to go out there and make the most out of this package!

I hope the information provided on this document was useful.
If there's anything missing, typos, or information you would like me to address in future versions, please don't forget to write an email and drop a few words at:

velvara.art@gmail.com



Happy Gamedeving!