

Hatching Post Processing

By Joseph Y.

—

Overview

The hatching/cross-hatching post-processing effect is an image effect that replicates the cross-hatching/hatching texture overlay in games like How2Escape. It is designed to run on the Unity URP pipeline and Unity 2019.3.9f1. This effect utilizes URP's render feature to integrate it into the render pipeline and ShaderGraph to create the shader effect.

The shader graph for this effect is based on IronStar Interactive's Image Effect Shader with Shader Graph (<https://www.youtube.com/watch?v=FpvJAG6R99k&t=6s>). The shader graph was modified to remove unnecessary features such as a paper texture overlay, eliminating the need for a custom node via subgraph and adding additional features like inverted hatching color and extra brightness to make the hatching more subtle. As a side effect, the scene may appear a bit brighter. Some parts of the blit/render feature code from IronStar were retained due to the need to process the code for rendering in the render pass, but the remaining code is original.

Installation

Before you begin:

1. Ensure you use the Universal Render Pipeline (URP) and Unity 2019.3.9f1.
2. Ensure you have installed the necessary packages, including ShaderGraph, Visual Scripting, and Post-Processing.
3. Ensure that the color space is set to linear in the project settings.

The effect is contained in a Unity package that you can import into Unity. To do this, navigate to Assets > Import Packages, select the Unity package, and open it. A popup will appear showing the contents of the package. Choose to import everything from the package and click OK. They will be imported into the Unity project under the Post Processing folder/Hatching Post Processing folder.

After importing, you should see a set of specific items. These include textures labeled as HatchX (where X is a number), 3 C# scripts (1 for the render feature, 1 for the render pass, and another for creating texture arrays), the hatching effect shader graph, 1 Texture Array with the hatching textures, and a material using the shader graph.

If you are not using the provided material, you can create a new material by right-clicking on the shader graph and selecting "Create Material." This will generate a material that uses the shader graph. To use a custom texture array, refer to [Using custom hatching textures](#).

To apply the effect, navigate to the universal renderer being used in your project and locate the "Add Render Feature" button. Next, click the "Add Render Feature" button and choose the hatch render feature. After that, input the material containing the hatching effect into the material section in the render feature's settings. If done correctly, you should be able to see the effect in both the game and scene view, confirming that it is working.

Adjusting the effect

To adjust the visual effects, you can go to the render feature's material and modify the parameters in the inspector window. Here are the details about the parameters:

1. **MainTexture:** This is the input field used to utilize Unity's `_MainTexture` property for shader use. It's required to be left exposed and should be ignored.
2. **Color Intensity:** This parameter adjusts the impact of colors on the effect. Lower values will make it look black and white, while setting it to 1 will make it look coloured.
3. **Extra Brightness & Extra Brightness Strength:** These parameters are used to make the hatching effect less prominent by brightening the hatching part in the shader graph code. They also make the visual a little bit brighter.
4. **Hatch Textures:** These are the textures for cross-hatching. They utilize a texture array to contain our hatching textures.
5. **Hatching Texture Tiling:** This parameter affects the tiling of the hatching textures in the effect.
6. **Hatch Texture Count:** This parameter affects the hatching texture count.
7. **Invert Hatch Colour & Invert Strength:** This parameter inverts the hatching color, and its strength can be adjusted.
8. **Animation Cycles Per Second:** This modifies the number of cycles that the hatches animate per second.

Using custom hatching textures

The hatching effect comes with a default texture array that can be used. This array consists of six different textures, ranging from the lightest (no hatching) to the darkest (lots of hatching). The effect utilizes the texture array to obtain the hatching texture based on the pixel's brightness, which ranges from 0 to 5.

In Unity, there isn't a built-in feature to convert selected textures to a texture array, so you'll need an external script. The hatching post-processing Unity package includes a script for creating a texture array, which can be found in Dmitry Denisov's Github Repo (<https://github.com/DmtDenisov/Texture-Array-Unity>). Alternatively, you can create your own script or use an existing one.

To create a texture array using the provided script:

1. Go to the "Tools" tab and select the "Texture Array Creator."
2. A window will appear with fields and options. Adjust the size to match the number of textures you are using. The number of elements at the bottom will change accordingly. Each element will have an input square for inserting textures. It is important to place the darkest or most densely hatched texture at element 0 and the lightest or least hatched texture at the last element. Place the rest of the textures in elements in a range from darkest to lightest (element 0 being the start of the range and the last element being the end of the range).
3. For the effect to work effectively, ensure that the background in the texture is white or transparent. It is recommended that you use black for the cross-hatchings and white for the background.
4. Once satisfied, click the button at the bottom to create the texture array. By default, it will be created and saved in the Assets/ directory with the name "NewTextureArray" unless you specify otherwise.

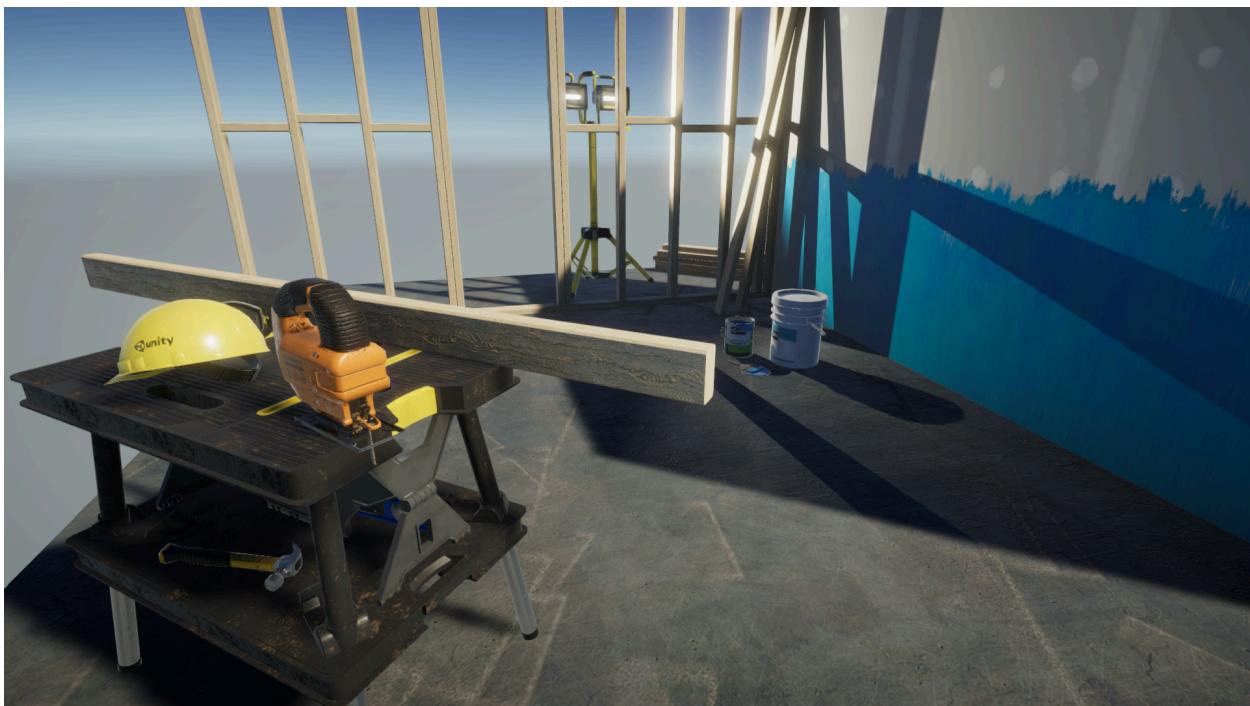
To change the texture array:

1. Go to the render feature's material inspector window.
2. Update or input the part of the hatching texture with the new texture array.

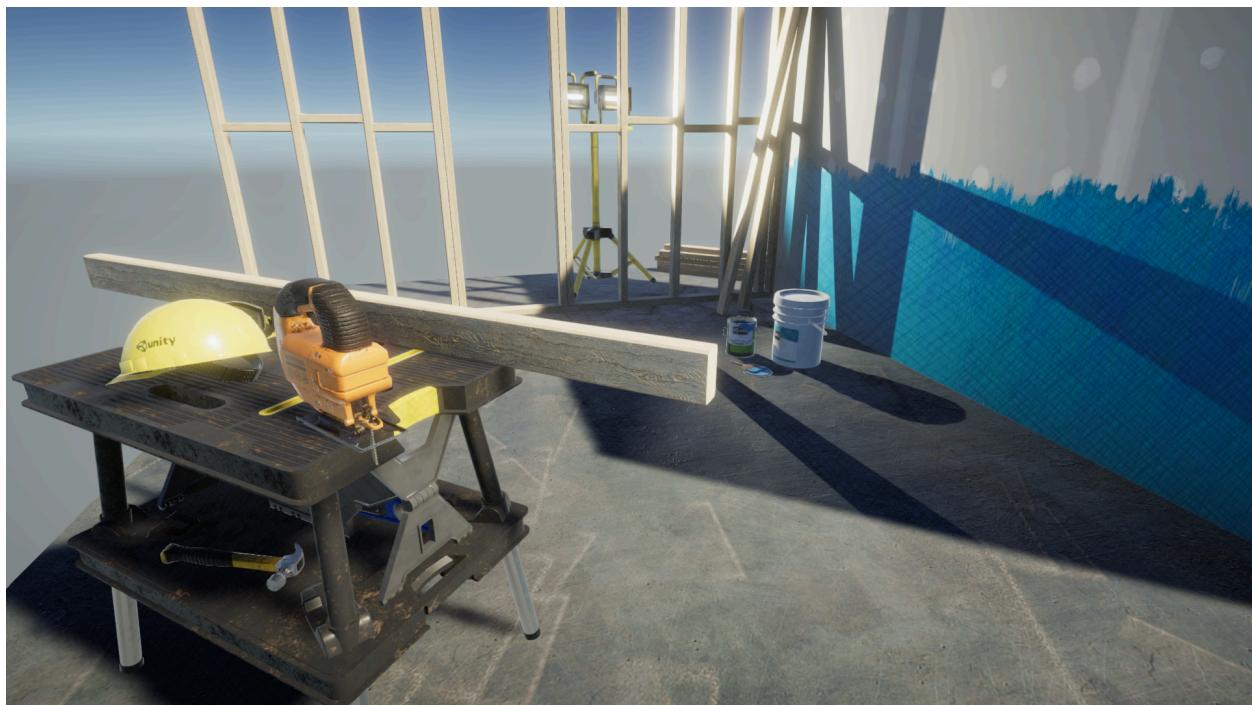
Update Log

V0.9: The base implementation of the hatching post processing.

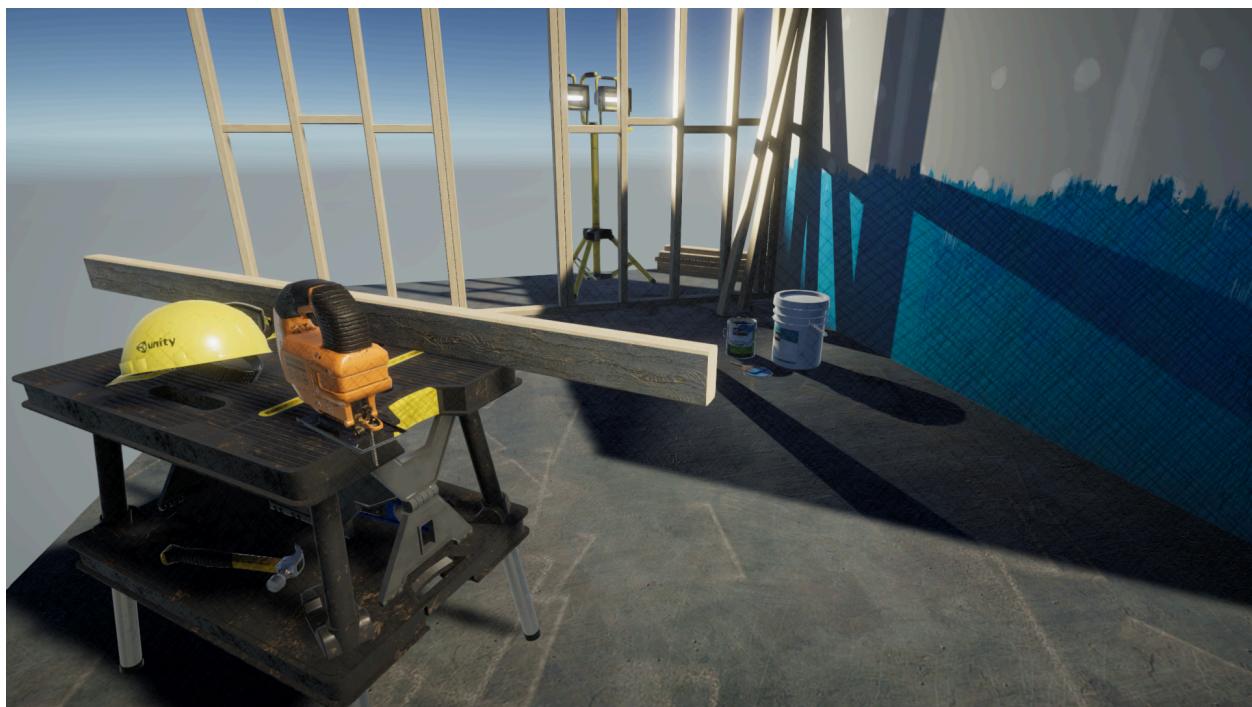
Example Screenshots



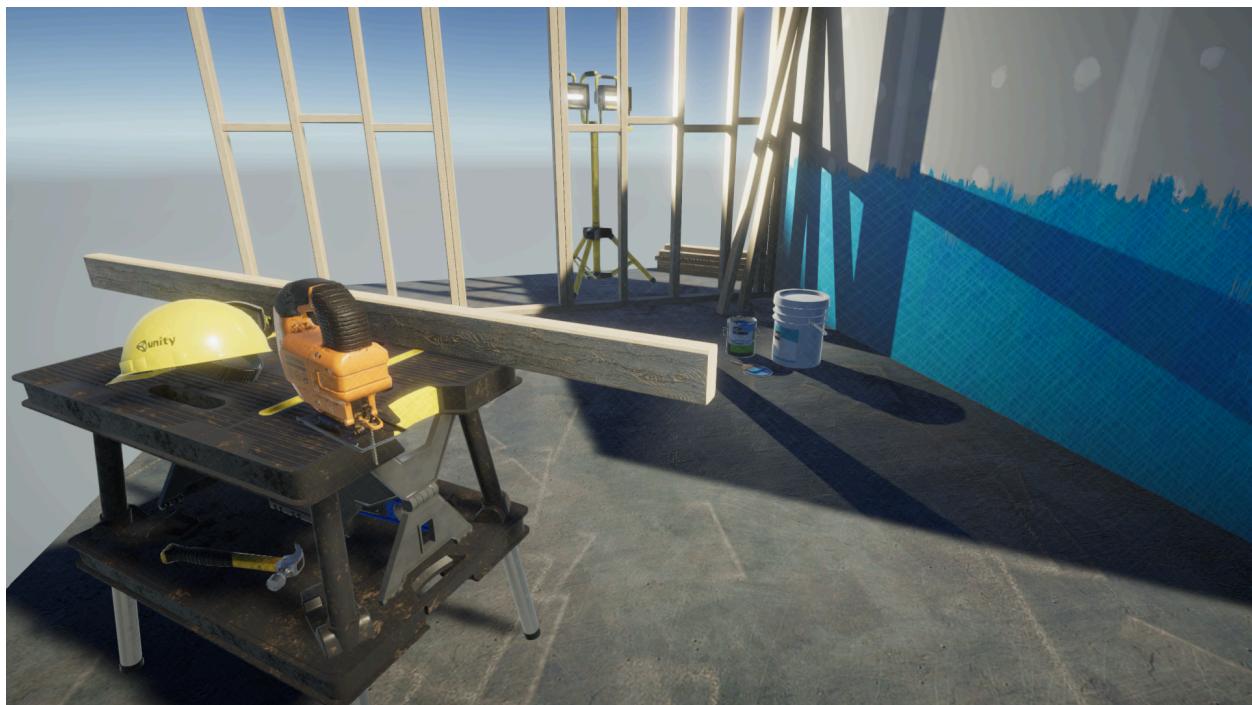
Without the effect



Screenshot of the Effect (With default parameters)



Effect without the extra brightness boost



Effect with inverted hatching colour



Effect with adjusted parameters to mimic the original effect