

Pixel Nostalgia v2.0

DFA Studios

May 31, 2019

Contents

1	Getting Started	2
2	Pixel Nostalgia	3
2.1	Post Processing Stack	3
2.1.1	Resolution Settings	3
2.1.2	Color Depth Settings	3
2.1.3	Dithering Settings	4
2.1.4	Extra Effects	6
2.1.5	Disable for Scene Camera	7
3	For versions Unity 5.6.1 to Unity 2017.2	7
4	For Unity 2017.3 and higher	7

1 Getting Started

We've gone full 360, and some of the most popular games on the market are pixel art games. The success of games like Super Meat Boy, Hotline Miami, Fez, and Shovel Knight, can partly be credited to their aesthetic. Pixel art is a constraint, and these constraints bring out the best in artists. The limited palette, dimensions, and animations an artist can work with mean that the result is unique, and an instant classic. This asset is an image effect that can be applied to your unity camera, and will let you render your games, simulating some of the hardware restrictions these artists worked with. Limiting the resolution of your game, or the colors we can draw, are a step in the right direction towards creating the next big pixel art game!



Figure 1: A sand dune. Settings are RGB232, 8x8 Dither, 1400 Separation, 1/4 Camera resolution.

Below is an example of a game with modern graphics. The effect can be applied to emulate older consoles, or different color depths. In Figure 2, there is more depth allocated to the green channel, and the blue channel is a single bit. These were chosen to get higher contrast in the sky colors, and more variation in the grass and foliage. The custom color depth settings are RGB231 - something that never existed, but you can use it.



Figure 2: The Sega Genesis preset applied to [Sonic Generations](#)

2 Pixel Nostalgia

2.1 Post Processing Stack

Pixel Nostalgia works directly with the Post Processing Stack, meaning it works with Lightweight Rendering Pipeline, High-Definition Rendering Pipeline, and the Default Rendering Pipeline. Once you've added the effect to the post-processing stack, each option you can tweak is organized into one of 4 sub-menus. If you're just getting started with the Unity Post Processing Stack, here is a comprehensive starting guide from Unity Technologies: <https://github.com/Unity-Technologies/PostProcessing/wiki/Quick-start>

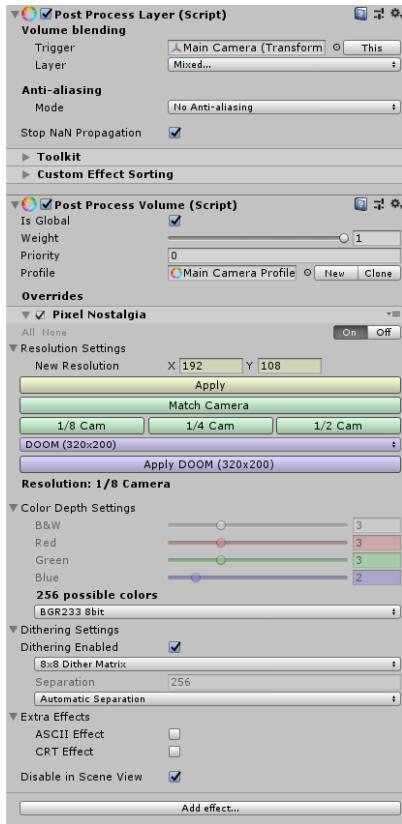


Figure 3: Resolution, Color Depth, Dithering, and Effects

camera by the height you would like to use. This will give you a width that you can use with that height value.

2.1.1 Resolution Settings

The resolution settings available are color coded based on what they do. The three color coded options will be outlined here:

- **Yellow Options:** These options allow you to explicitly set the resolution that Pixel Nostalgia should use. This resolution is how many pixels there will be vertically, and horizontally. If you're going for a very low-resolution game, the values under "New Resolution" should be quite small. Clicking the "Apply" button switches this to be the resolution the effect uses. The numbers can't be smaller than 1x1.
- **Green Options:** These are the camera options. In case you're unsure of a nice resolution to use, try exploring these first. The first option "Match Camera" will use the cameras size. This means nothing will change. Using the fractions below will do what they suggest, ie: "1/8 Cam" will use a resolution that is 1/8th the horizontal and vertical resolution of your camera. This is nice for matching the aspect ratio. ie: All of the pixels will be square.
- **Purple Options:** This dropdown menu contains some common resolutions. The N64 option for example, emulates the default resolution of the Nintendo 64 console. If you'd like to explore these, the "Apply chosen_resolution" button will switch the effect to use this new resolution. The DOS ASCII option is a good resolution to try with the ASCII effect.

The yellow and purple options may not match the aspect ratio of your screen. The aspect ratio can be computed as follows: $\text{aspectRatio} = \text{width}/\text{height}$. If you'd like to maintain square pixels, you can multiply the aspect ratio of your

By default, most consumer monitors can display 24bit True Color. This means that the color is represented by 24 bits evenly divided between the red channel, the blue channel, and the green channel. Mixing Red-Green-Blue can approximate any color in the spectrum. With less bits to represent each channel however, you start losing colors that you can represent. In fact with just 1 bit, you're left with only two options: 1 or 0! If you have one bit for a red channel, this means you have two options: No red, or entirely red. This wikipedia article can explain in more detail:

https://en.wikipedia.org/wiki/Color_depth. It's worth learning if you want to get the most out of this effect. Here are the options.

- **BW:** How many bits are used when rendering in black and white.
- **Red:** How many bits represent the red channel.
- **Green:** How many bits represent the green channel.
- **Blue:** How many bits represent the blue channel.
- **Presets:** Presets are common color depths that you can choose between. If you're looking at BGR233, you can tell it is 8 bit. We know this because the blue channel has 2 bits, the green channel has 3 bits, and the red channel has 3 bits. If we total the bits used among all channels, we have $2 + 3 + 3 = 8$.

If the Preset "Custom" or "Black and White" is chosen, you can modify these bit depth values with the sliders, and experiment with custom colors.

2.1.3 Dithering Settings

Dithering is applied to help blend colors. This effect is comparable to stippling done by pencil artists, or Ben-Day dots used by comic book printers. The concept is simple: if you're looking at a pattern of pixels from a distance, you stop seeing individual colors and instead see the average of the colors.



Figure 4: A puppy face made up of only the following colors: **White, Red, Yellow, Magenta, Black.**

Figure 4 is a bit extreme because there are only 8 possible colors (1 bit per channel). But if we zoom out and look at the puppy face, we start to see more colors than we're drawing it with.

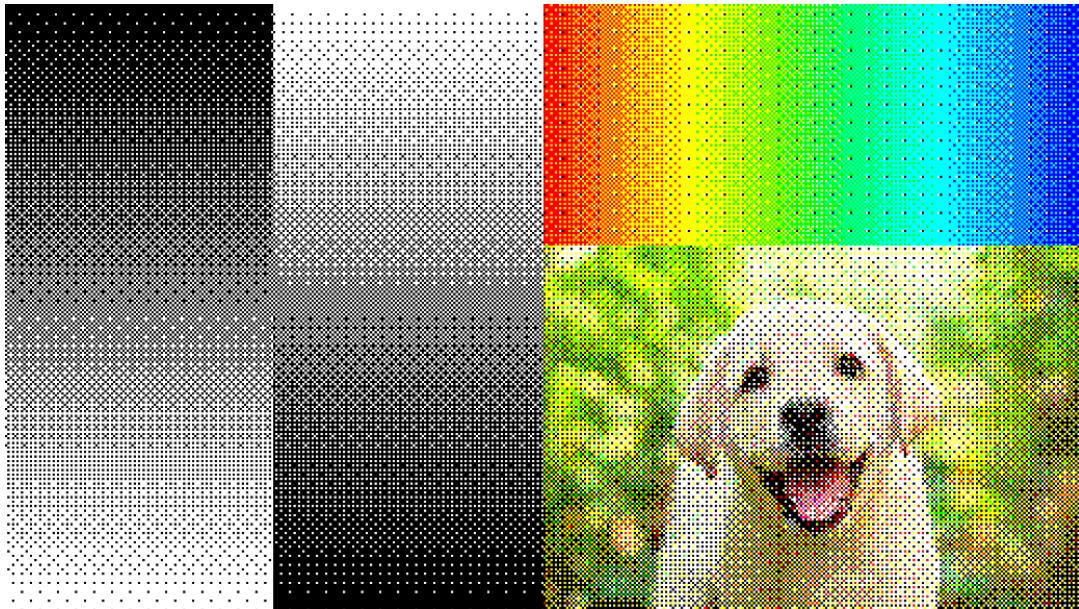


Figure 5: The same puppy, but zoomed out.

The gradients on the left from black to white, use only 2 colors. Black, and white. The pattern chosen comes from an effect called Dithering, and we have a few options to control this effect.

- **Do Dithering:** Obviously, this is an optional effect. Dithering is common in games, but it might not be what you want. It's all about the aesthetic you want.
- **Dither Matrix:** There are many types of dithering. This effect is an ordered dither, and it uses a matrix to look up the pattern. The matrices are either 2x2, 3x3, 4x4, or 8x8. The 8x8 matrix has 64 possible shades it can represent with 2 colors, the 4x4 has 16, 3x3 has 9, and 2x2 has 4. In the demo scene you can experiment to see the effects they have.
- **Dither Separation:** Assuming you have the Custom Separation dropdown selected, this slider lets you control the reach of the dither. A larger number here will decrease the feathering that occurs as a direct result of dithering. Try experimenting with the gradients in the demo scene. Swap between different sized matrices to see the effects. It's not recommended to go lower than the automatic separation value, only higher.

2.1.4 Extra Effects

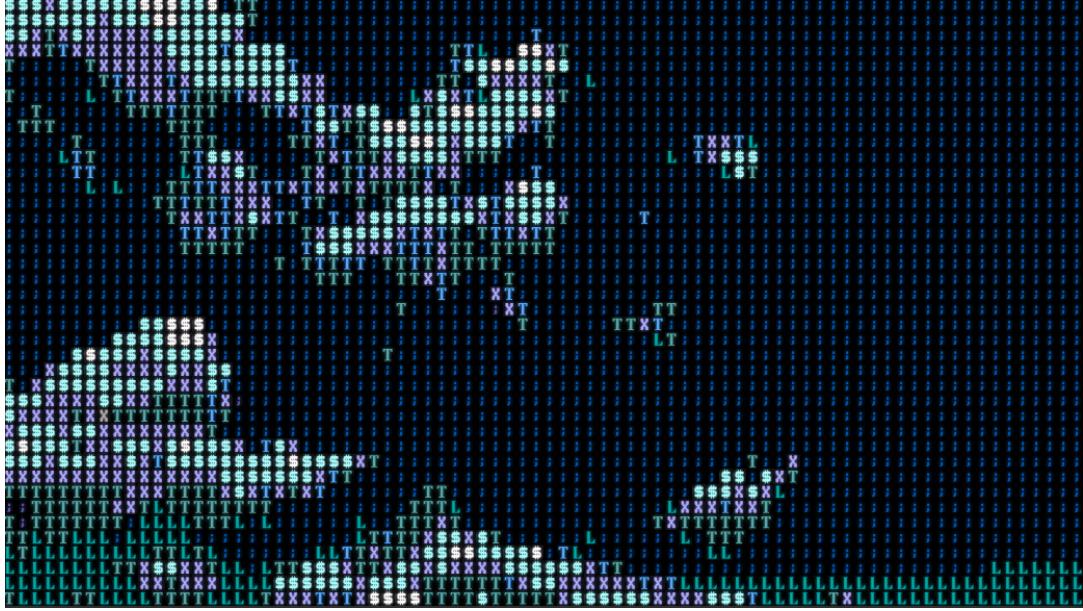


Figure 6: It's hard to distinguish, but these are clouds. ASCII is not ideal for an entire game, but it's a fun effect, even included by some AAA titles.

Two extra effects are added to PixelNostalgia. Keeping with the theme of retro gaming and old school hardware, the first effect is an ASCII effect. ASCII stands for American Standard Code for Information Interchange. It's basically letters and numbers. If you want to draw the scene using letters and numbers, try this toggle out. It works best with very low resolutions, because you will be able to see the characters (letters and numbers are referred to as ASCII characters).

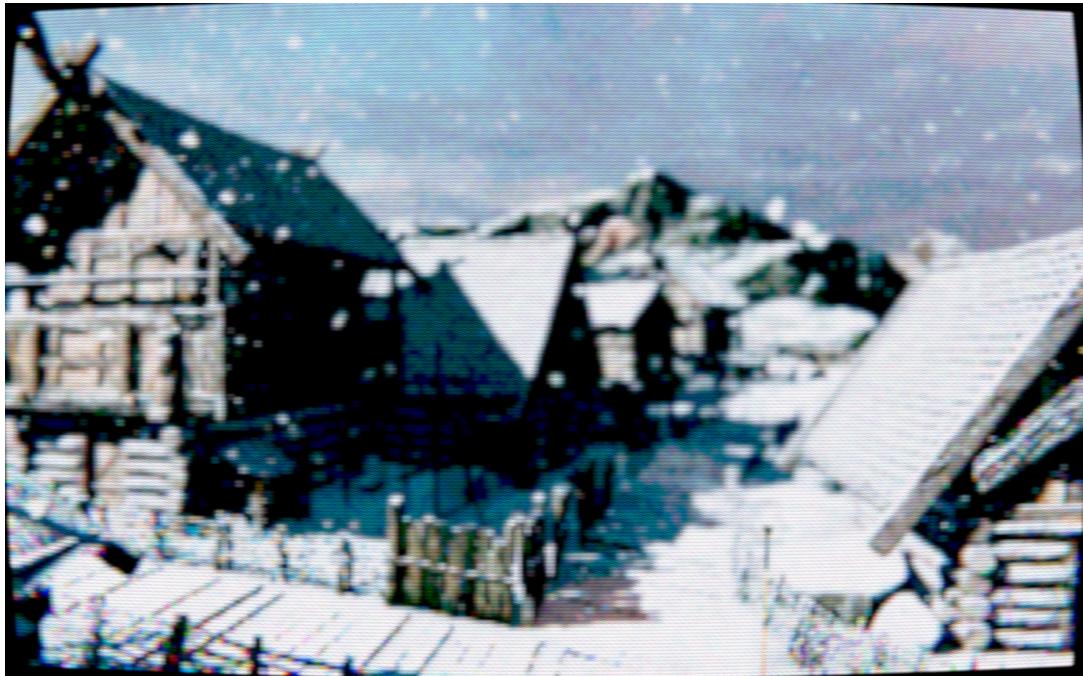


Figure 7: CRT effect on the viking village scene.

The second effect you can turn on is a CRT effect. This mimics the curvature, blur, and

separation of pixels that you get from an old CRT monitor.

2.1.5 Disable for Scene Camera

Having the effect in the scene view is nice, if you want to rapidly see what something looks like with PixelNostalgia turned on. If you'd like to turn it off for the scene camera, just enable the "*Disable In Scene*" checkbox.

3 For versions Unity 5.6.1 to Unity 2017.2

For Unity 5.6.1 through to Unity 2017.2, the post processing stack needs to be downloaded from the UnityTechnologies GitHub page. Newer versions can use the Package Manager. <https://github.com/Unity-Technologies/PostProcessing>

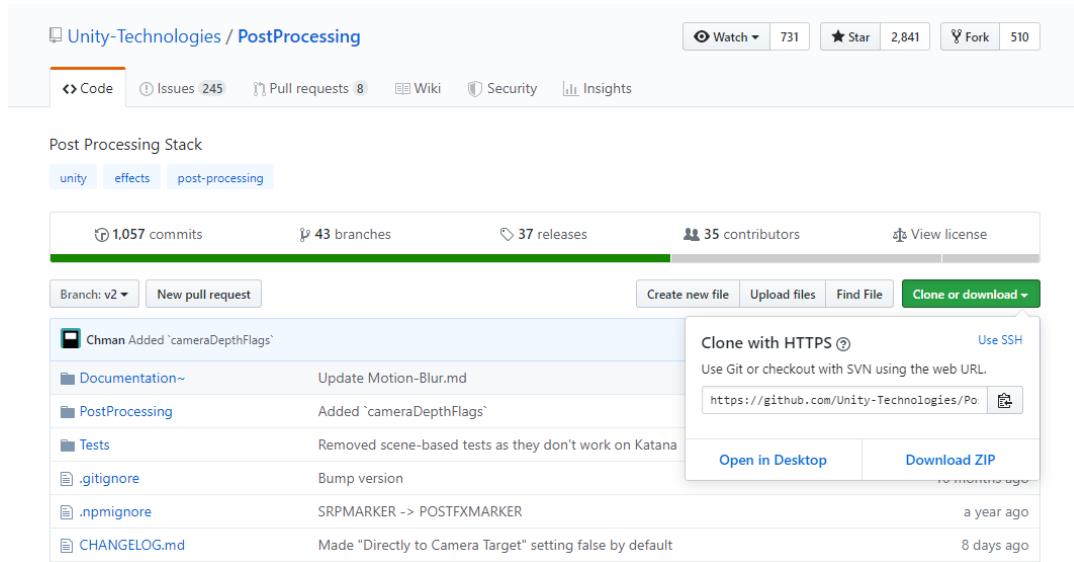


Figure 8: Where to click, to download the Post Processing Stack from GitHub.

Once downloaded, the 'PostProcessing' folder can be added to your project. At this point, it's safe to comment out the second line of the Pixel Nostalgia code files. `PixelNostalgia.cs`, and `PixelNostalgiaEditor.cs`. At this point, you can use it as a regular Post Processing Stack effect.

4 For Unity 2017.3 and higher

For Unity 2017.3+, the post processing stack can be downloaded from the Package Manager. Once it is installed, it is safe to comment out the fourth line of the Pixel Nostalgia code files. `PixelNostalgia.cs`, and `PixelNostalgiaEditor.cs`. At this point, you can use it as a regular Post Processing Stack effect.

If you enjoyed the asset, please leave a review! If you have any issues, please send me an email so I can get a fix out as soon as possible. Enjoy!

[Website](#) | [eMail](#) | [YouTube](#)