



## WHAT IS "RETRO SHADERS PRO FOR GODOT"?

*Retro Shaders Pro for Godot* is a collection of shader effects which emulate the look and feel of retro games. This style is commonly referred to as the "PSX style" due to its similarity to the look and feel of the PS1.

## SUPPORT

Sometimes things break! If you've encountered an error and this README doesn't have the answers (or if you have feature requests), then pop me an email at

[danielilett+retroshadersgodot@gmail.com](mailto:danielilett+retroshadersgodot@gmail.com) and I'll try to sort you out. Please:

- **Clearly** describe the problem you are having and what steps I can take to reproduce the error.
- Include the **Godot version you are using**.
- Also include the **package version** you are using (you can find this in the project root in a file called "version.txt").
- Include the **renderer** (i.e. Forward Plus, Compatibility, Mobile) – this setting can be found in *Project Settings -> Rendering -> Renderer*.
- **Attach images or short videos** where necessary to describe your problem.

Following these steps will help me fix your issue as quickly as possible!

## SETUP

**Folder Structure:** You will receive a zip folder, and upon extraction, the folder structure will be *addons/Retro Shaders Pro/everything else*. A demo scene can be found under *Retro Shaders Pro/Demo/Scenes/Demo.tscn*, which explains all the settings available to the shaders.

The shaders are included inside *Retro Shaders Pro/Shaders*, including the base Retro Lit shader, skybox shaders, and CRT shaders for post processing and for meshes.

## MESHES

The **Retro Lit** and **Retro Lit Transparent** shaders are intended to be used with regular meshes in your scene. To use these shaders, create a ShaderMaterial and drag one of these shaders onto the Shader field, then modify the shader parameters to your heart's content and attach the material to objects in your scene.

You may also use the **CRT Mesh** shader to apply the CRT-style effect to any regular mesh in your scene.

## SKYBOX

*Retro Shaders Pro* contains three Skybox shaders, each of which supply some options similar to the **Retro Lit** shader for limiting the resolution and color depth. To use a skybox, add a WorldEnvironment node to your scene and attach a material using one of these shaders to the Sky parameter in the environment. You might need to change the Background to use Sky mode to get it to show up.

The base **Retro Skybox** shader can be used to sample an equirectangular texture, which can be applied to the sky.

The **Retro Skybox Cubemap** version takes a cubemap as input and wraps it around the sky.

The **Retro Skybox Procedural** version generates a sky gradient and a cloud pattern which can be dynamically adjusted. An example of this kind of material is included in the demo scene.

## POST PROCESSING

This shader pack uses a fullscreen quad which samples the screen texture in order to render the CRT post process effect. You may find [this video](#) helpful to learn more about post processing in Godot if you're unsure about setting it up, although an example is included in the demo scene.

You should be able to create a ShaderMaterial using the CRT Post Process shader and then attach it to a quad mesh MeshInstance3D with a few tweaks:

- In Geometry, set its Extra Cull Margin set to the maximum value, 16384.0m.
- Make the mesh 2m x 2m in size.
- Enable flip faces on the mesh.

The shader will take care of putting it in front of the camera.

The latest version of this asset was created in Godot 4.5.

## ASSETS INCLUDED

The following assets are included in the asset pack:

### RETRO LIT

A PSX-style shader that can be attached to any mesh to instantly give it a classic PS1 aesthetic.

This shader can use affine texture mapping for sampling textures (as this was a limitation of the original PS1 hardware), which causes objects to appear warped when viewing them at extreme angles, or when viewing large triangles.

#### Retro Properties

- **Base Color** – The albedo color of the object.
- **Base Texture** – An albedo texture to apply more color detail than *Base Color* alone.
- **Resolution Limit** – Sets an upper bound on the resolution of *Base Texture*.
- **Snaps Per Meter** – The vertices of the mesh will snap to this number of snap points per meter along each axis (in view space, i.e., relative to the camera).
- **Color Depth** – Each color channel is constrained to this many possible values. Low values may darken your image because a floor function is applied.
- **Color Depth Offset** – Applies a slight offset to the colors to avoid the darkening issue.
- **Affine Strength** – Blend between affine and perspective-correct mapping.
- **Filtering Mode** – Choose between pixelated point filtering, regular bilinear filtering, and Nintendo 64 style 3-point bilinear filtering.
- **Lighting Mode** – Choose between unlit, standard lit, and texel-aligned lighting (where shadows adhere to the pixel grid created by the base texture).
- **Use Dithering** – Toggle the dithering effect, which ‘blends’ between color values according to a Bayer matrix pattern when the Color Depth is reduced.
- **Use Vertex Colors** – Choose whether to multiply the base color using vertex colors.

### RETRO OUTLINE

A simple hull outline effect which supports many of the same retro-style features of the other effects in the pack. Hull outlines were a common way to achieve a toon look in some PS2 and late PS1 games.

- **Base Color** – The albedo color of the outlines.
- **Thickness** – How far the mesh vertices are extended along their vertex normals in world space units.
- **Snaps Per Meter** – The vertices of the mesh will snap to this number of snap points per meter along each axis (in view space, i.e., relative to the camera).

### RETRO SKYBOX

This skybox shader reads from an equirectangular texture and unwraps it into a skybox.

- **Base Color** - The albedo color of the sky.
- **Base Texture** – An equirectangular texture containing the sky colors.

- **Resolution Limit** – Sets an upper bound on the resolution of *Base Texture*.
- **Color Bit Depth** – Each color channel is constrained to this many possible values. Low values may darken your image because a floor function is applied.
- **Color Bit Depth Offset** – Applies a slight offset to the colors to avoid the darkening issue.
- **Use Point Filtering** – Choose whether to use nearest neighbor or bilinear filtering.

## RETRO SKYBOX [CUBEMAP]

This skybox shader reads from a cubemap texture and applies it to the sky.

- **Base Color** - The albedo color of the sky.
- **Base Texture** – A cubemap texture containing the sky colors.
- **Resolution Limit** – Sets an upper bound on the resolution of *Base Texture*.
- **Color Bit Depth** – Each color channel is constrained to this many possible values. Low values may darken your image because a floor function is applied.
- **Color Bit Depth Offset** – Applies a slight offset to the colors to avoid the darkening issue.
- **Use Point Filtering** – Choose whether to use nearest neighbor or bilinear filtering.

## RETRO SKYBOX [PROCEDURAL]

This skybox shader generates noise clouds which scroll above the scene, alongside a color gradient.

- **Ground Color** – Skybox color close to the horizon.
- **Sky Color** – Skybox color at the very top of the sky.
- **Color Mix Power** – Lets you configure which of **Ground Color** or **Sky Color** are more strongly mixed in the sky gradient.
- **Cloud Height Threshold** – Controls how far the clouds extend. The first value determines a cutoff point for 0% opacity, and the second value determines at what point the clouds use 100% opacity.
- **Cloud Sizes** – Values used for the noise generator while creating the cloud shapes.
- **Cloud Density Threshold** – Controls the amount of cloud that appears. The first value thresholds the generated noise values. The second value controls where the clouds reach 100% opacity.
- **Cloud Color** – Tint applied to the clouds.
- **Cloud Velocity** – How fast the clouds scroll across the sky.
- **Resolution Limit** – Pixelates the noise texture.
- **Combine Mode** – Choose how to combine the two cloud maps.
  - Cloud value = Cloud Size 1 <operation> Cloud Size 2.

## CRT

The CRT post processing effect covers the entire screen to make it appear like an old-school CRT TV, complete with distortion effects and RGB subpixels.

## RESOLUTION & FIDELITY

- **Pixel Size** – An integer value representing how pixelated the image becomes.

- **Scale in Screen Space** – Choose whether to scale the size of other parameters relative to a reference resolution. Enable this when you want pixelation/RGB properties to appear the “same size” on different resolution screens.
- **Reference Resolution (Vertical)** – Set this to the vertical screen resolution you are designing the game around. When Scale Parameters is enabled, each parameter remains unchanged when the screen is at this resolution.

## BARREL DISTORTION

- **Distortion Strength** – Controls how strongly the edges of the screen warp inwards to form the shape of a CRT glass screen.
- **Background Color** – Color of the areas outside the distorted CRT screen shape.

## RGB SUBPIXELS & SCANLINES

- **Subpixel Texture** – Texture to use for the RGB subpixel effect. All pixels on the screen are multiplied by this texture such that the red, green, and blue screen colors appear separate to each other.
  - An example of a texture to use for this is contained in “Retro Shaders Pro/Resources/Textures/RGBTexture.png”.
- **Subpixel Strength** – How strongly the RGB subpixel effect is applied.
- **Scanline Texture** – Texture to use for the scanline effect. All pixels on the screen are multiplied by this texture such that scanlines appear scrolling over the image.
  - An example of a texture to use for this is contained in “Retro Shaders Pro/Resources/Textures/ScanlineTexture.png”.
- **Scanline Strength** – How strongly the scanline effect is applied.
- **Scanline/RGB Size** – Larger values make the scanlines (and RGB subpixels) appear larger on-screen.
- **Scanline Scroll Speed** – How quickly the scanline texture scrolls over the screen.

## VHS ARTIFACTS

- **Random Wear** – Adds small, noisy UV distortions horizontally to simulate the fuzzy look common with old VHS tapes.
- **Aberration Strength** – How strongly chromatic aberration (color channel separation) is applied at the screen edges.
- **Use Tracking** – Toggle the VHS tracking artifacts.
- **Tracking Texture** – A control texture for tracking artifacts.
  - This should be an x-by-1 image.
  - The red channel controls the strength of the tracking UV offsets.
  - The green channel controls the presence of tracking lines overlaid onto the screen.
  - An example of a texture to use for this is contained in “Retro Shaders Pro/Resources/Textures/TrackingRamp.png”.
- **Tracking Size** – How ‘zoomed in’ the tracking ramp is when scrolling over the screen.
- **Tracking Strength** – How strongly the tracking ramp red channel offsets the UVs of the screen horizontally.

- **Tracking Speed** – How quickly the tracking ramp scrolls across the screen. Use negative values to scroll upwards.
- **Tracking Jitter** – A random offset applied to the scrolling to make it appear jittery.
- **Tracking Color Damage** – Cycle the chrominance of the image slightly to look like the tape is damaged. The screen is converted to YIQ color space, and the offset is applied to the I and Q channels.
- **Tracking Lines Threshold** – A threshold for tracking lines to appear on screen. Higher values mean fewer lines.
- **Tracking Lines Color** – The color of the tracking lines, where the alpha channel acts as a global multiplier on the tracking line strength.

## COLOR ADJUSTMENTS

- **Tint Color** – A global tint applied to the entire CRT screen effect.
- **Brightness** – Global multiplier for the image colors before some effects are applied. A value of 1 preserves the image as-is.
- **Contrast** – Forces differences in colors to become more obvious. A value of 1 preserves the image as-is.

## CRT (MESH)

A version of the CRT effect which can be applied to a regular mesh in your scene. This effect supports most of the parameters of the post-process version, besides the variables relating to screen resolution.

## SPECIAL THANKS

Many thanks to:

- Contributors to [this thread about texel lighting](#), especially user [GreatestBear](#)
- [ambientCG](#) for many of the CC0 licensed textures used in the demo
- [freestylized.com](#) for the toon cubemap skybox
- [Polyhaven](#) for the equirectangular skybox texture
- [OpenGameArt](#) for some CC0 licensed textures used in the demo
- [Timothy Ahene on Sketchfab](#) for the CRT model used in the promotional art
- [Sergej Majboroda on Polyhaven](#) for the sky HDRI used in the promotional art