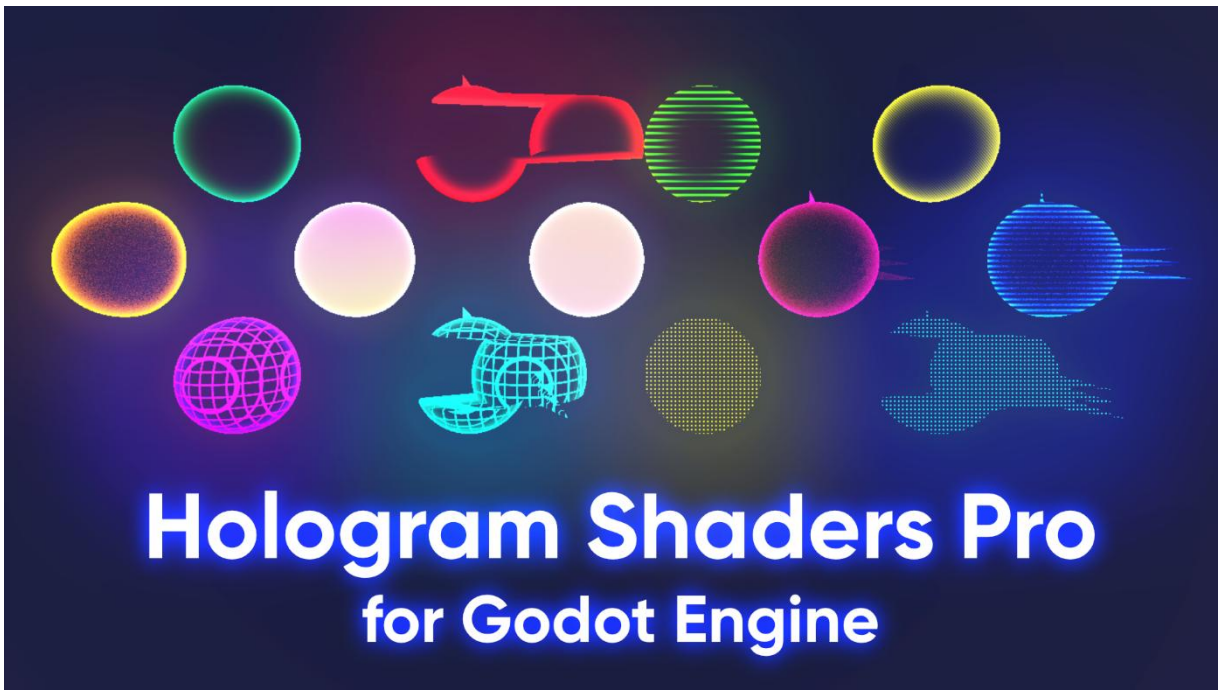

WHAT IS "HOLOGRAM SHADERS PRO FOR GODOT"?

Hologram Shaders Pro for Godot is a collection of **new hologram shaders** for Godot, featuring glitch effects, scanlines, gridlines, dot matrix effects, noise, gradients, and more!

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+hologramshadersgodot@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*.
- A**ttach images or short videos** where necessary to describe your problem.

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

PARAMETERS INCLUDED

The following parameters are available on shaders in the collection:

BASIC PBR PROPERTIES

These properties control the basic interactions between Godot's lighting and the object. Includes the crucial emissive color settings that control the appearance of the holograms.

- **Output Mode** – Controls whether the shader outputs color to Albedo, Emission, or both.
- **Base Color** – Controls the color of the object.
- **Base Texture** – Also controls the color of the object. This is multiplied by Base Color.
- **Normal Texture** – Used to change the normal vectors on the surface, which influences lighting interactions. May not be noticeable due to the usage of bright emissive light.
- **Normal Strength** – A value between 0 and 1 which controls how strongly the Normal Texture impacts the surface normals.
- **Alpha Clip Threshold** – Any pixel with an output alpha value below this value is culled entirely. Useful for Dot Matrix holograms.

SCANLINES

Some of the hologram shaders support a hologram texture scrolling down the object. This scanline controls the output alpha of the object, which in turn influences which parts of the object glow.

- **Scanline Mode** – An enum with three values: *Screen Space* means the Scanline Texture will be applied relative to the position of pixels on the screen; *World Space* means the scanline is applied based on the position of the object in the world; *None* will turn off the scanlines.
- **Scanline Rotation** – Only works in Screen Space mode. Rotates the direction of the scanlines.
- **Scanline Direction** – Only works in World Space mode. Orients the scanlines along this vector in world space.
- **Scanline Texture** – A texture which encodes the scanline pattern. You can increase the scaling of this texture for closely-packed scanlines, and vice versa. In World Space mode, only the vertical information matters.
- **Scanline Velocity** – Controls the speed at which the scanlines scroll across the object.
- **Scanline MinMax Alpha** – Completely black pixels from the Scanline Texture will output the Min value (first component of this vector). Completely white pixels from the Scanline Texture will output the Max value (second component of this vector).

VERTEX GLITCHES

Some of the hologram shaders support glitch effects. This type of glitch effect pulls individual vertices of the object to the side (away from the center of the object).

- **Use Vertex Glitches?** – A Boolean value – when on, vertex glitches will be displayed, and when off, vertex glitches are turned off.

- **Glitch Sensitivity** – How easy it is for a given vertex to fall under the random threshold and start glitching. Higher values mean a lower chance to glitch. This is a value between 0 and 1 that I recommend keeping very high (above 0.99) unless you want a lot of glitches.
- **Glitch Normal Multiplier** - A multiplier weight value applied to each component of the normal vector for calculating the offsets, e.g. (1, 0, 1) will pull vertices only along the X-Z plane.
- **Glitch Strength** – How far glitches vertices extend from their original position along their vertex normal.
- **Glitch Offset** – A time offset – you can use two materials with identical settings except a different offset, and they won't glitch at the same time.
- **Glitch Frequency** – How often glitches occur.

SEGMENT GLITCHES

This type of glitch effect takes a horizontal slice of the object and pulls all the vertices in the slice in a specified direction briefly.

- **Use Slice Glitches?** – A Boolean value – when on, slice glitches will be displayed, and when off, slice glitches are turned off.
- **Slice Width** – The width, in world space, along the y-axis that the glitch takes up.
- **Slice Speed** – How quickly the glitch effect scrolls down the object.
- **Slice Frequency** – How often glitches occur.
- **Slice Jitter** – A slight random offset which makes it appear as if the glitch jitters. Probably don't set it too high or the shader may flash too rapidly (capped to 0.2 by default).
- **Slice Duration** – How long a slice glitch event occurs for.
- **Slice Direction** – The direction in which all vertices in the glitch are pulled in world space.

NOISE

This type of glitch overlays randomly generated noise values to make the hologram appear low-fi.

- **Use Noise** – For the Uber hologram, choose whether to enable noise calculations.
- **Noise Speed** – How fast the shader scrolls through random noise values.
- **Noise Scale** – How fine or coarse the noise values are in screen space.
- **Noise Strength** – How strongly the noise colors are overlaid onto the mesh.
- **Noise Color** – A color tint applied to the noise values.

GRADIENT

Gradient holograms use a second set of color properties with a falloff value to produce a color gradient across the surface.

- **Gradient Space** – Should the gradient be calculated across Object, World, or Screen space?
- **Base Color 2** – Controls the color of the other side of the hologram.
- **Fresnel Color 2** – The color of the Fresnel highlight on the other side of the hologram.

- **Gradient Min-Max Y** – Which height values should be used as the start and end of the 'intermediate' portion of the gradient. Height values below the first (x) value output *Base Color*. Height values above the second (y) value output *Base Color 2*.

GRIDLINE OPTIONS

For grid-based holograms, these properties control the thickness and spacing of the gridlines.

- **Per Axis Strength** – The visibility of the gridlines along each axis. Set any one of these values to 0 to turn off gridlines along that axis.
- **Rotation Axis** – The axis around which world space rotates.
- **Grid Density** – How closely-packed the gridlines are to each other.
- **Line Thickness** – How thick each gridline is in world space units.
- **Line Falloff** – How much falloff there is along the edge of a gridline before the gridline is completely gone.
- **Grid Offset** – An offset of the grid in world space along the three axes.
- **Grid Velocity** – The speed at which the grid scrolls along the three axes. Can be set to 0.

DOT MATRIX PATTERN

For the dot matrix holograms, these settings control the density and spacing of the screen-space dots.

- **Dot Size** – The size, in pixels, of each dot on-screen. All dots are perfectly square.
- **Dot Space** – The size, in pixels, of the space between dots. The spacing is the same on each axis.

COMMON ROTATION PROPERTIES

Both the Grid-based and Dot Matrix-based holograms use this rotation system.

- **Rotation Radians** – The amount of rotation around Rotation Axis, in radians.

FRESNEL

Fresnel adds light to the edge/rim of an object. Most of the hologram shaders support Fresnel lighting.

- **Fresnel Color** – Color of the Fresnel/rim light. Can be different to the Base Emission Color.
- **Fresnel Power** – The size of the Fresnel/rim. Higher values mean the rim is smaller, but this should be a value above 0.
- **Use Scene Intersections** – Should the shader add a highlight to parts of the object that intersect other objects? This functionality depends on the depth buffer to detect intersections.
- **Intersection Power** – How sensitively the shader should detect intersections. Higher values mean a thinner intersection.

UPDATES & PATCH NOTES

PATCH 1.0.0

First release. Added eight types of hologram shader (some with common features). Broadly maintains feature parity with the Unity version of this asset pack.

- Basic
- Glitch
- Scanline
- Grid
- Dot Matrix
- Noise
- Gradient
- Uber