Railgun Shaders

Introduction

To achieve a non-photorealistic, stylized look, that is also highly customizable and very flexible, we decided to roll out our own lighting model.

To keep this easy to use for modders we decided to support both node based workflows and fully coded based workflows.
To keep everything coherent, we supply both code libraries that you can import in your code and nodes that use the same functions in the code libraries. This way, no matter how you structure the shaders, be it code or nodes, the underlying code that will run will be functionally the same. (However, a fully code based approach would be preferable in terms of performance and control)

Our NPR Workflow

What is NPR?

NPR stands for Non Photorealistic Rendering. This is opposed to **PBR** which stands for Physically Based Rendering.
NPR is all sorts of stylized rendering that doesn't want to imitate real life. Such as "anime-style" rendering. Which is, kind of, our goal here.

How is it structured

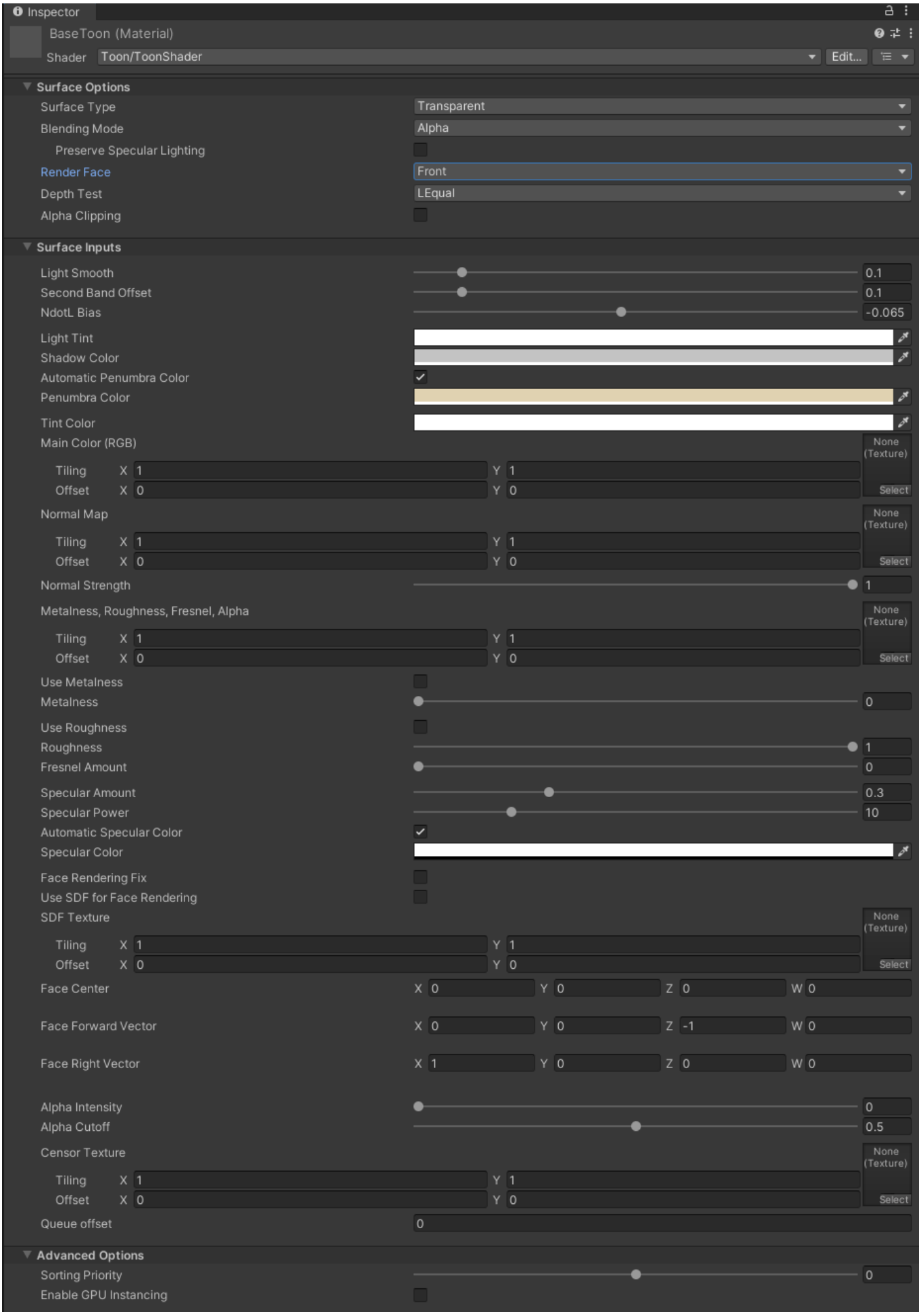
We currently provide a single "uber-shader" that basically gives you access to all the features available from our library. This allows for a faster rendering as the gpu doesn't have to switch shader program as often, saving some time. This also allows the gpu to render multiple objects as they were one, further improving the frame draw time.

> ⓘ **Info**
>
> This might change in the future, as there are already plans for some more specialized versions of the shaders.

However, internally, every feature of the shader is modular. If a modder wants to write a custom shader that does just a couple of things differently than how we do it, it will be easy to reuse the code.

The Material

From the unity editor, a material appears like this:


Inspector
BaseToon (Material)
Shader
Toon/ToonShader
Edit...
Surface Options
Surface Type
Transparent
Blending Mode
Alpha
Preserve Specular Lighting
Render Face
Front
Depth Test
LEqual
Alpha Clipping
Surface Inputs
Light Smooth
0.1
Second Band Offset
0.1
NdotL Bias
-0.065
Light Tint
Shadow Color
Automatic Penumbra Color
Penumbra Color
Tint Color
Main Color (RGB)
None (Texture)
Tiling X 1 Y 1
Offset X 0 Y 0
Select
Normal Map
None (Texture)
Tiling X 1 Y 1
Offset X 0 Y 0
Select
Normal Strength
1
Metalness, Roughness, Fresnel, Alpha
None (Texture)
Tiling X 1 Y 1
Offset X 0 Y 0
Select
Use Metalness
Metalness
0
Use Roughness
Roughness
1
Fresnel Amount
0
Specular Amount
0.3
Specular Power
10
Automatic Specular Color
Specular Color
Face Rendering Fix
Use SDF for Face Rendering
SDF Texture
None (Texture)
Tiling X 1 Y 1
Offset X 0 Y 0
Select
Face Center X 0 Y 0 Z 0 W 0
Face Forward Vector X 0 Y 0 Z -1 W 0
Face Right Vector X 1 Y 0 Z 0 W 0
Alpha Intensity
0
Alpha Cutoff
0.5
Censor Texture
None (Texture)
Tiling X 1 Y 1
Offset X 0 Y 0
Select
Queue offset
0
Advanced Options
Sorting Priority
0
Enable GPU Instancing

The first block, *Surface Options*, allows the user to customize some general settings about how the material should be treated at render time.

Option	Values	Description
Surface Type		Selects how this material should be treated.
	Opaque	Renders only solid/fully transparent (i.e. with "holes") surfaces.
	Transparent	Allows for semi-transparent surfaces (glass, see-trough fabric etc...)
Blending Mode		*(available only if surface tye is transparent)* Selects how the transparent surface is going to blend with itself and the background.
	Alpha	Uses the surface alpha as mix blending amount.
	Pre-Multiply	Multiplies the surface color by it's own alpha value then blends it with the underlying color.
	Additive	Adds it's own color (multiplied by it's own alpha) to the underlying color.
	Multiply	Multiplies it's own color by it's own alpha, then by the underlying color.
Preserve Specular Lighting	True/False	Does nothing due to our own custom lighting.
Render Face		Selects which face side are considered for rendering.
	Front	*(default)* Uses the side that is considered as exterior.
	Back	Uses the opposite side, the one considered interior. Useful if your model appears invisible from the correct side.
	Both	Evaluates both sides for rendering. This is the preferred way to handle clothing.
Depth Test		Selects how to determine if as pixel will be drawn or not depending on the depth. In most cases, `LEqual` is fine.
Alpha Clipping	True/False	Enables the use of an alpha map to cut holes on an object. This works both in Opaque and Transparent. (it is reccommended to use it only in opaque to avoid depth sorting errors)
Threshold	0f-1f	Selects the cutoff point at which the alpha will be considered transparent and not opaque.

The second block, *Surface Inputs*, allows the user to configure the appearance of the actual material.

Option	Values	Description
Light Smooth	0f-1f	Controls how sharp the shading is. 0 will provide a fully sharp shadow appearance, 1 a smoothed shading.
Second band offset	0f-1f	Controls how more smoothed is the second shadow band. This doesn't need to be larger than Light Smooth as it uses `Light Smooth + Offset` to perform a new additive light smooth. Large values might cause a "light ring" to appear on shadows, adjust the `Penumbra Color` to a darker color if it happens or use `Automatic Penumbra Color`
NdotL Bias	-1f - 1f	Biases the light/shadow calculation in order to reduce or increase the shadowed areas. Extreme values can cause weird shading in complex lighting situations.
Light Tint	Color	All of the lighting will be tinted by this color. (`LightColor * LightTint`) This is not an HUE only coloring, rather a straight multiplication. Using a shade of grey here will effectively reduce the overall light intensity on the surface.
Shadow Color	Color	All of the shadowed parts of the object will be tinted (multiplied) by this color.
Automatic Penumbra Color	True/False	Selects if you want an automatically calculated penumbra color. If true, the penumbra color will be based of the current shadow color, using a lighter shade.
Penumbra Color	Color	The color to use for the second band shading. This will do nothing if `Automatic Penumbra Color` is active.
Tint Color	Color	Tint to apply to the `Main Color` texture. Or overall color of the object if no texture is assigned.
Main Color (RGB)	Texture2D	Color texture applied to the object.
Normal Map	Texture2D	Tangent space Normal map to use on the object.
Normal Strenght	0f-1f	Intensity of the normal map texture over the surface.
Metalness, Roughness, Fresnel, Alpha	Texture2D	From here on referred as "MRFA", multichannel map encoding variable values for the entire surface, allowing the user to have all 4 of them to vary across. The sliders below will still influence the value stored on the texture, so you can leave the channels you aren't interested in as full (255 or 1f).
Use Metalness	True/False	Selects if you want the stylized metalness calculation to be performed or not.
Metalness	0f - 1f	How much the metalness effect will be applied to the surface.
Use Roughness	True/False	Selects if you want the surface to receive realtime and baked enviroment reflections, including diffused ones.

Option	Values	Description
Roughness	0f - 1f	How rough the simulated micro surface is supposed to be. 0 being perfectly smooth, 1 extremely rough.
Fresnel amount	0f - 1f	Makes the reflections more transparent when the view direction matches the surface angle. Higher values will push reflection visibility towards the edges. This option requires `Use Roughness` enabled.
Specular Amount	0f - 1f	How much visible the specular highlights will be on the surface.
Specular Power	0.01f - 100f	How sharp the specular highlight will be. Lower values will make it more faint and diffused, higher values will make it smaller and more intense.
Automatic Specular Color	True/False	Selects if the Specular Color should be determined by the surface color or set manually.
Specular Color	Color	Color to be used if the specular color is set manually. The Specular highlight will be tinted by this color.
Face Rendering Fix	True/False	Enables/Disables the lighting adjustments for face shadowing. Enable this **only** if your material is used on a character face.
Use SDF for Face Rendering	True/False	Enables/Disables the use of a Signed Distance Field-encoded Shadowmap for face shadowing.
SDF Texture	Texture2D	Red/Green channel encoded shadowmap. R channel for right hemisphere, Green for left (w/ Head facing you).
Face Center	Vector 4D	*Ignore these as they will be set by the face helper component*
Face Forward Vector	Vector 4D	*Ignore these as they will be set by the face helper component*
Face Right Vector	Vector 4D	*Ignore these as they will be set by the face helper component*
Alpha Intensity	0f - 1f	*Ignore as currently not used*
Alpha Cutoff	0f - 1f	*Controls the Alpha clipping Threshold, it's the same property*
Censor Texture	Texture 2D	Used to make mosaic censorship appear on censored versions of the game.
Queue Offset	Integer Value	*Changes when the material will render. This can be problematic, so avoid changing it.*
Hair Highlight Toggle	True/False	Should hair highlights be used on this material.
Hair Highlight Color	Color	Color of the hair highlight.
Hair Highlight Strength	0f - 1f	Controls the strength of the hair highlight.

Option	Values	Description
Hair Highlight Exponent	0f - 20f	Controls how soft or hard the highlight looks, better used in combination with noise to obtain a harsher or softer light effect.
Hair Highlight Length	0f - 1.5f	Controls the maximum length of te highlight. The effective length may vary based on exponent and amount of noise applied.
Hair Highlight Noise Strength	0.001f - 3f	How strong the noise effect on the highlight should be.
Hair Highlight Nosie Stretch	0.001f - 50f	Frequency of the noise. Lower gives more coarse low frequency noise, higher gives finer high frequency noise.
Hair Highlight Noise Seed	0 - 9999999	Seed used for noise sampling, changing this will change noise patterns.
Hair Highlight Fresnel Power	0.001f - 100f	Controls how narrow the fresnel effect on individual hair strands is.
Hair Highlight Fresnel Bias	0.7f - 1f	Controls how narrow the overall fresnel effect should be on the whole hair.

Advanced options documentation can be retrieved from unity docs themselves.