SEGI beta v0.8

Real-Time Global Illumination For Unity

User Guide v1.0

1. Introduction

Thank you for being an early adopter of SEGI! Your experience and feedback with this asset in its beta phase will help SEGI become the best that it can be!

1.1 Summary

SEGI is a voxel-based Global Illumination effect that aspires to provide 100% interactive Global Illumination to Unity games and applications. Since it requires no precomputation, SEGI can bring GI to certain situations where precomputed solutions like Enlighten cannot!

SEGI provides indirect lighting and glossy reflections from a single directional light, the sky, and any emissive materials in the scene. SEGI calculates indirect light visibility for soft indirect shadows, and also calculates soft sky light shadows. It can render either a single bounce or infinite bounces of indirect light. In the future, SEGI will support indirect lighting from point and spot lights.

1.2 Compatibility

As this is the first beta version of SEGI, compatibility is limited. SEGI has only been thoroughly tested on a Windows PC; behavior on other platforms is unknown. Improving compatibility is of high priority. SEGI is not compatible with mobile devices, and probably never will be.

2. Usage Guide

This section will help you to set up SEGI in your Unity scenes and will provide tips for implementation so that you can make the most out of this effect.

2.1 Adding SEGI to Your Scene

Step 1: Set Color Space to Linear

Calculating lighting in gamma-space is probably one of the most crucial mistakes visually a game can make. Make sure that your game's lighting isn't in gamma-space!

1. From the Menu Bar, go to Edit > Project Settings > Player

2. In the Inspector, under Other Settings, select the Color Space dropdown and select Linear

Step 2: Set the Rendering Path to Deferred

As of now, SEGI only works for the deferred rendering path. This can either be set per-camera or project-wide.

1. Select your main camera and set the Rendering Path property in the Inspector to Deferred

2. To set all cameras to render deferred by default, go to Edit > Project Settings > Player, and under Other Settings, set Rendering Path to Deferred

Step 3: Disable Unity's Ambient Lighting and GI

Since SEGI renders its own ambient lighting and GI, you'll want to disable them for a proper result.

1. Open the Lighting window by going to Window > Lighting in the Menu Bar

2. Ensure that Ambient Intensity is set to 0

3. Uncheck Precomputed Realtime GI and Baked GI

Step 4: Add SEGI to Your Main Camera

1. Select your main camera in the Hierarchy, and from the inspector, click the Add Component button.

2. In the drop-down menu, select Image Effects > Sonic Ether > SEGI. Alternatively, you can type SEGI and hit Enter

3. Ensure that SEGI is above any other effects on the camera. This includes SSAO effects, fog effects, or any other post-processing effects. SEGI must be applied to the image before any other image processing happens to ensure a correct result

Step 5: Assign the Sun Property to Your Main Directional Light

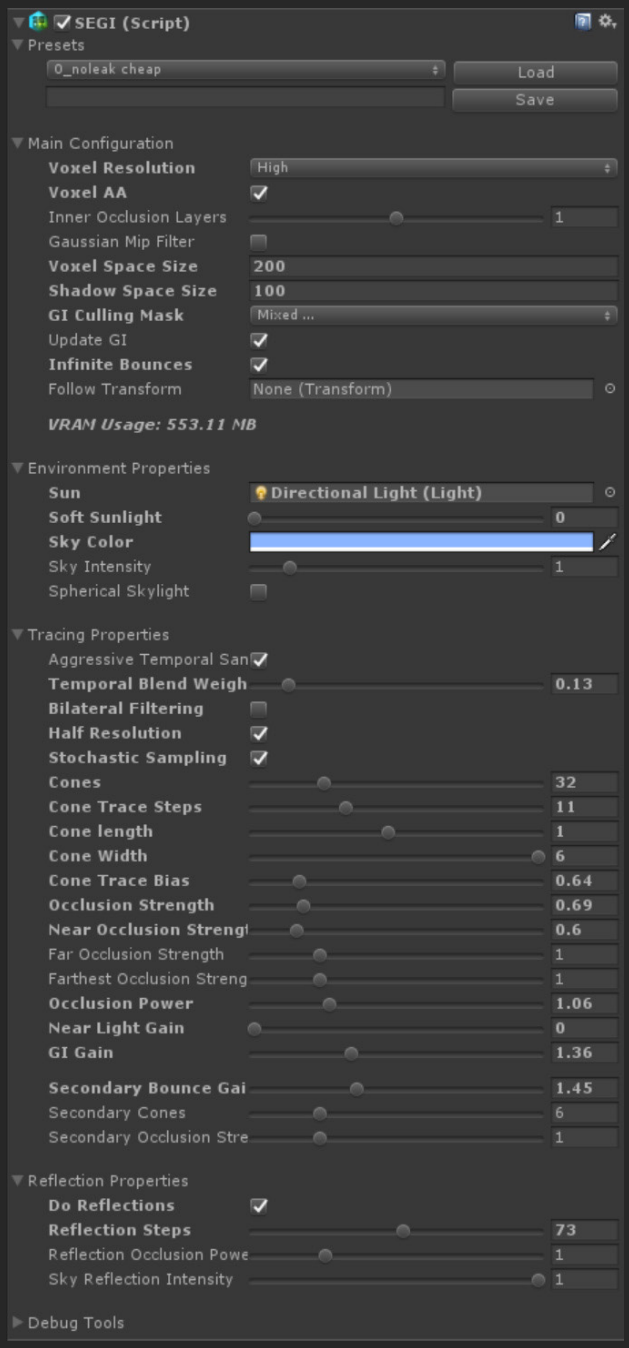
If your scene uses a key directional light as sunlight, you'll need to assign it to the Sun property so SEGI can calculate indirect lighting from it. If your scene does not, you can leave this property unassigned.

1. Drag and drop your key directional light from the Hierarchy to the Sun property under Environment Properties in the Inspector UI for SEGI

3. SEGI Parameter Info

SEGI has a lot of parameters, and they will all be explained in this section.

The parameters for SEGI are broken down into 5 groups: Main Configuration, Environment Properties, Tracing Properties, Reflection Properties, and Debug Tools.



Main Configuration is where you'll control the technical aspects of SEGI such as Voxel Resolution and Voxel Space Size (extents and scale of GI).

Environment Properties is where you'll assign your key directional light and adjust environmental lighting settings.

Tracing Properties is where you'll control various parameters for the diffuse indirect lighting tracing.

Reflection Properties is where you'll control options for reflection tracing.

Debug Tools provides some visual debugging tools for making sure everything is working properly.

3.1 Main Configuration

Voxel Resolution: The resolution of the voxel volume in which scene data is stored to render GI. This parameter has a significant performance impact and should be one of the first things to consider reducing if performance is too heavy.

Voxel AA: When enabled, this parameter enables 8x supersampling during the voxelization pass. This can improve the impact moving or small objects have on indirect lighting. The performance cost of this property is dependent on the complexity of the scene that is being voxelized.

Inner Occlusion Layers: Represents how many black opaque voxel layers are written on the backside of voxelized geometry. This can significantly improve light leaking artifacts, but can cause issues with thin or small objects. In the future, this parameter will be controlled per-object instead of globally to help with these problems.

Gaussian Mip Filter: When enabled, a gaussian filter will be used during the mipmapping step (instead of a basic box filter). This can improve the smoothness and consistency of indirect lighting and reflections. This has a significant performance impact at high voxel resolution, and a minor one at low voxel resolution.

Voxel Space Size: The extents of the voxel volume in world-space units. The higher the setting, the more far away objects will contribute to GI, and consequently the larger the voxels will be. The extents of the voxel volume can be visualized while the camera with SEGI is selected in the scene view. This visualization is turned off when the inspector UI for SEGI is collapsed in the inspector.

Shadow Space Size: The extents of the shadow map used during direct light injection during voxelization. This value should be around the same that you've chosen for Voxel Space Size, and can be a bit smaller to provide increased sunlight indirect light detail near the camera.

GI Culling Mask: Just like the Culling Mask property on a camera, this property allows you to select which layers are voxelized and which aren't. This can be extremely useful for having objects that don't render on-screen but affect GI, or for optimization by utilizing low-detail proxy objects for voxelization instead of the high-detail objects that are being rendered on-screen. It can also be useful for skipping the voxelization of small objects/characters to save on rendering time.

Update GI: If enabled, GI voxel data is updated interactively. If this is disabled during run-time, GI voxel data will stop updating and the indirect illumination in the scene will simply be calculated with cached data from the last time this was enabled. In the future, GI data updating will be smarter and do less redundant work, but for now, if there are moments in which you know not much will change in your scene to affect lighting, this can be disabled to run on cached data.

Infinite Bounces: If enabled, SEGI will utilize a secondary tracing step to render unlimited indirect lighting bounces. This works with a feedback loop, so each secondary bounce is based on lighting data from the previous frame and added back during the light injection phase. Secondary bounces can therefore "lag behind" lighting updates. The performance impact that this has is dependent on the number of secondary cones (in Tracing Properties) and the complexity of the voxelized scene.

Follow Transform: If a transform is assigned here, the voxel GI origin will be centered on this transform instead of on the camera it is attached to. This can be useful for situations where the camera is far away from any objects in the scene (a distant top-down view).

VRAM Usage: An estimate of how much graphics memory SEGI is using. This is pretty much entirely dependent on Voxel Resolution.

3.2 Environment Properties

Sun: The directional light that will be considered the key directional light in the scene to render indirect light from. If nothing is assigned here, SEGI will skip directional light injection. If your scene has sunlight and the directional light is not assigned here, it will not contribute to GI.

Soft Sunlight: This adds cone-traced soft sunlight to the scene. This can be useful for partially cloudy environment lighting or just to add a soft look to the outdoor lighting. It is particularly effective for sunset lighting.

Sky Color: The color of sky light that will be added to the scene. This should be set to the average color of the sky in your scene. In the future, you'll be able to select a cubemap here and have sky light match the assigned cubemap.

Sky Intensity: The brightness of sky light in the scene.

Spherical Skylight: If enabled, sky light will come from all directions (up and down). If disabled, sky light will only come from above (the top hemisphere of the sky).

3.3 Tracing Properties

Aggressive Temporal Sampling: If enabled, based on the Temporal Blend Weight, the diffuse GI result will be an exponential moving average based on the result from previous frames. This means that previous frames contribute to the current frame, and when used in conjunction with Stochastic Sampling, this can reduce the graininess of the diffuse GI. If disabled, diffuse GI will simply be an average between the current frame and the previous frame. If you want to completely stop previous frames from contributing to the current frame's GI, enable this and set Temporal Blend Weight to 1. Temporal sampling is still in an experimental phase and will cause some artifacts with moving objects, this will be improved in the future.

Temporal Blend Weight: If Aggressive Temporal Sampling is enabled, this will control how much previous frames of GI will contribute to the current frame. The lower the value, the more previous frames will contribute, and the more "lazily" diffuse GI will appear to update. Lower values used in conjunction with Stochastic Sampling will provide smoother, less grainy and more temporally coherent results. A value of 1 means previous frames will not contribute at all.

Bilateral Filtering: If enabled, diffuse GI will be blurred in an edge-aware manner to smooth out graininess. This is meant to be used in conjunction with Stochastic Sampling, and isn't really needed if Stochastic Sampling is disabled.

Half Resolution: If enabled, diffuse GI will be calculated at half the render resolution. This is highly recommended to keep diffuse GI tracing fast. Unless your scene is very complex geometrically, you can usually get away with this being enabled with minimally noticeable drawbacks.

Stochastic Sampling: If enabled, the diffuse cones that are traced to calculate indirect lighting will be randomly rotated per-pixel. This essentially trades banding for noise. If Aggressive Temporal Sampling is enabled, the randomization will vary for each frame. This parameter has a very slight performance impact when enabled.

Cones: The number of diffuse cones to trace for diffuse indirect lighting. Higher values result in less noise/banding, but require a higher performance cost. Lower values can be used here if Gaussian Mip Filter is enabled.

Cone Trace Steps: The number of steps that will be taken during each cone trace to gather GI data for diffuse indirect lighting. Higher values result in less radial banding or stepping artifacts, but require a higher performance cost. Higher values also cause indirect shadows to darken, so you may wish to compensate by adjusting Occlusion Strength. Lower values can be used here if Gaussian Mip Filter is enabled.

Cone Length: The length of each cone that is traced for diffuse indirect lighting. This is essentially an adjustment for the "GI radius", meaning, this will directly influence how distant objects can be to contribute to GI. If you need a higher "GI radius", you may want to simply increase Voxel Space Size instead of this parameter.

Cone Width: The width of each cone that is traced for diffuse indirect lighting. Wider cones result in more spatially coherent results (less noise/banding) but can cause over-occlusion and self-bleeding artifacts. Lower values can have a negative impact on performance.

Cone Trace Bias: The amount of offset away from any surface that cone tracing begins. If you are encountering self-occlusion or self-bleeding artifacts (voxel acne), increase this value. Values too high can cause light leaking.

Occlusion Strength: A global modifier for how much light should be blocked by occupied voxels. This uniformly affects the strength of near and far indirect shadows.

Near Occlusion Strength: The strength of shadowing nearby solid objects will cause. Only affects the strength of very close blockers. Reducing this can improve close-proximity shadowing artifacts.

Far Occlusion Strength: How much light far solid objects block. This value gives additional light blocking proportional to the distance of the current cone trace step.

Farthest Occlusion Strength: How much light the farthest occluders block. This value gives additional light blocking proportional to the squared distance of the current cone trace step.

Occlusion Power: This affects the strength of shadowing in a different way from the other parameters. Higher values cause indirect shadows to "fill-out".

Near Light Gain: Affects the attenuation of indirect light. Higher values allow for more close-proximity indirect light. Lower values reduce close-proximity indirect light, sometimes resulting in a cleaner result.

GI Gain: The overall brightness of indirect light.

Secondary Bounce Gain: Affects the strength of secondary/infinite bounces. Be careful, values above 1 can cause runaway light bouncing and flood areas with extremely bright light!

Secondary Cones: The number of cones that will be traced for calculating secondary bounces. Increasing this value improves the accuracy of secondary bounces at the cost of performance. The performance impact this causes is proportional to voxelized scene complexity.

Secondary Occlusion Strength: The strength of light blocking during secondary bounce tracing. Be careful, a value too low can cause runaway light bouncing and flood areas with extremely bright light!

3.4 Reflection Properties

Do Reflections: If enabled, cone-traced reflections will be rendered.

Reflection Steps: The number of steps taken while tracing reflections. Higher values reduce tracing artifacts but cause a higher performance cost.

Reflection Occlusion Power: Affects light blocking during reflection tracing step. Similar to Occlusion Power property. Higher values can reduce reflection light leaking.

Sky Reflection Intensity: The brightness of sky reflections applied during reflection tracing.

3.5 Debug Tools

Visualize Sun Depth Texture: Displays the depth texture used to inject sunlight with shadows into GI voxel data. This can be useful for checking the accuracy of injected sunlight.

Visualize GI: Displays only the indirect lighting result.

Visualize Voxels: Displays a direct traced view of the voxel data as seen from the camera's position. This can be very helpful in checking that the right objects are getting voxelized properly.

4. Effectively Utilizing SEGI

This section includes things that you should know that will help you make the most of SEGI. Fully real-time global illumination solutions are forced to cut corners to achieve real-time performance, and SEGI is no different. Therefore, sometimes more care is needed than putting SEGI on the main camera and forgetting about it!

4.1 Saving and Loading Presets

SEGI comes with several presets to get you started, but you can also save and load your own presets.

Loading Presets

1. Select the preset you would like to load in the drop-down

2. Click the Load button to load the selected preset

Saving Presets

1. Type the name of an existing preset you want to overwrite, or a new preset you would like to save in the text input field

2. Click the Save button to save the preset

Managing or Deleting Presets

All presets are stored in the folder SEGI / Resources / Presets. SEGI Presets are .asset files with the special type SEGIPreset. Any presets placed here can be loaded from within SEGI's main UI. This means you can delete, duplicate, or share and exchange SEGI presets.

4.2 Using "GI Only" Objects

There are many situations where you might want to have objects in the scene contribute to GI but not to be rendered on the screen directly, such as using low-detail proxy objects as contributors to GI instead of the high-detail objects that are rendered directly on-screen. You can set this up with Layers and Culling Masks.

1. Create a unique layer for all of your "GI Only" objects

2. Assign the layer to any object you want to influence GI but not be rendered directly

3. On your main camera, make sure that this layer is unchecked in the Culling Mask

4. Ensure that this layer is included in the GI Culling Mask property inside SEGI so that anything in this layer gets voxelized and contributes to GI

4.3 Using GI Blockers

Included in this asset is the GIBlocker shader. This shader allows you to use any object to provide additional light blocking.

1. Create a new material and assign the GIBlocker shader (SEGI > GIBlocker) to it

2. Assign the material to any object you want to act as an additional indirect light blocker

3. Follow the steps in Section 4.2 above to make sure your GI blockers are not rendered to the screen directly

The Blocker Value property determines how much additional light blocking power this object will have. Higher values mean that this object will cast stronger indirect shadows

5. Support

If you need help with SEGI or would like to report an issue, you can either visit the thread for this asset on the Unity Forums, or you can email me at cody@sonicether.com

If you are reporting an issue, please provide a detailed description of the problem you are experiencing, as well as system information (graphics card, operating system, Unity version, etc.).