SceneEEVEE(bpy_struct)

base class — bpy_struct

class bpy.types.**SceneEEVEE(bpy_struct)**

Scene display settings for 3D viewport

bokeh_max_size

Max size of the bokeh shape for the depth of field (lower is faster)

TYPE:

float in [0, 2000], default 100.0

bokeh_neighbor_max

Maximum brightness to consider when rejecting bokeh sprites based on neighborhood (lower is faster)

TYPE:

float in [0, 100000], default 10.0

bokeh_overblur

Apply blur to each jittered sample to reduce under-sampling artifacts

TYPE:

float in [0, 100], default 5.0

bokeh_threshold

Brightness threshold for using sprite base depth of field

TYPE:

float in [0, 100000], default 1.0

clamp_surface_direct

If non-zero, the maximum value for lights contribution on a surface. Higher values will be scaled down to avoid too much noise and slow convergence at the cost of accuracy. Used by light objects.

TYPE:

float in [0, inf], default 0.0

clamp_surface_indirect

If non-zero, the maximum value for indirect lighting on surface. Higher values will be scaled down to avoid too much noise and slow convergence at the cost of accuracy. Used by ray-tracing and light-probes.

TYPE:

float in [0, inf], default 10.0

clamp_volume_direct

If non-zero, the maximum value for lights contribution in volumes. Higher values will be scaled down to avoid too much noise and slow convergence at the cost of accuracy. Used by light objects.

TYPE:

float in [0, inf], default 0.0

clamp_volume_indirect

If non-zero, the maximum value for indirect lighting in volumes. Higher values will be scaled down to avoid too much noise and slow convergence at the cost of accuracy. Used by light-probes.

— —

TYPE:

float in [0, inf], default 0.0

fast_gi_bias

Bias the shading normal to reduce self intersection artifacts

TYPE:

float in [0, 1], default 0.05

fast_gi_distance

If non-zero, the maximum distance at which other surfaces will contribute to the fast GI approximation

TYPE:

float in [0, 100000], default 0.0

fast_gi_method

Fast GI approximation method

- `AMBIENT_OCCLUSION_ONLY` Ambient Occlusion – Use ambient occlusion instead of full global illumination.
- `GLOBAL_ILLUMINATION` Global Illumination – Compute global illumination taking into account light bouncing off surrounding objec

TYPE:

enum in [‘AMBIENT_OCCLUSION_ONLY’, ‘GLOBAL_ILLUMINATION’], default ‘GLOBAL_ILLUMINATION’

fast_gi_quality

Precision of the fast GI ray marching

TYPE:

float in [0, 1], default 0.25

fast_gi_ray_count

Amount of GI ray to trace for each pixel

TYPE:

int in [1, 16], default 2

fast_gi_resolution

Control the quality of the fast GI lighting. Higher resolution uses more memory.

- `1` 1:1 – Full resolution.
- `2` 1:2 – Render this effect at 50% render resolution.
- `4` 1:4 – Render this effect at 25% render resolution.
- `8` 1:8 – Render this effect at 12.5% render resolution.
- `16` 1:16 – Render this effect at 6.25% render resolution.

TYPE:

enum in [‘1’, ‘2’, ‘4’, ‘8’, ‘16’], default ‘2’

fast_gi_step_count

Amount of screen sample per GI ray

TYPE:

int in [1, 64], default 8

fast_gi_thickness_far

Angular thickness of the surfaces when computing fast GI and ambient occlusion. Reduces energy loss and missing occlusion of far geometry.

TYPE:

float in [0.0174533, 3.14159], default 0.785398

fast_gi_thickness_near

Geometric thickness of the surfaces when computing fast GI and ambient occlusion. Reduces light leaking and missing contact occlusion.

TYPE:

float in [0, 100000], default 0.25

gi_cubemap_resolution

Size of every cubemaps

TYPE:

enum in ['128', '256', '512', '1024', '2048', '4096'], default '512'

gi_diffuse_bounces

Number of times the light is reinjected inside light grids, 0 disable indirect diffuse light

TYPE:

int in [0, inf], default 3

gi_glossy_clamp

Clamp pixel intensity to reduce noise inside glossy reflections from reflection cubemaps (0 to disable)

TYPE:

float in [0, inf], default 0.0

gi_irradiance_pool_size

Size of the irradiance pool, a bigger pool size allows for more irradiance grid in the scene but might not fit into GPU memory and decrease performance

TYPE:

enum in ['16', '32', '64', '128', '256', '512', '1024'], default '16'

gi_visibility_resolution

Size of the shadow map applied to each irradiance sample

TYPE:

enum in ['8', '16', '32', '64'], default '32'

gtao_distance

Distance of object that contribute to the ambient occlusion effect

TYPE:

float in [0, 100000], default 0.2

gtao_quality

Precision of the horizon search

TYPE:

float in [0, 1], default 0.25

light_threshold

Minimum light intensity for a light to contribute to the lighting

TYPE:

float in [0, inf], default 0.01

motion_blur_depth_scale

Lower values will reduce background bleeding onto foreground elements

TYPE:

float in [0, inf], default 100.0

motion_blur_max

Maximum blur distance a pixel can spread over

TYPE:

int in [0, 2048], default 32

motion_blur_steps

Controls accuracy of motion blur, more steps means longer render time

TYPE:

int in [1, inf], default 1

overscan_size

Percentage of render size to add as overscan to the internal render buffers

TYPE:

float in [0, 50], default 3.0

ray_tracing_method

Select the tracing method used to find scene-ray intersections

- `PROBE` Light Probe – Use light probes to find scene intersection.
- `SCREEN` Screen-Trace – Raytrace against the depth buffer. Fallback to light probes for invalid rays..

TYPE:

enum in ['PROBE', 'SCREEN'], default 'SCREEN'

ray_tracing_options

EEVEE settings for tracing reflections

TYPE:

`RaytraceEEVEE`, (readonly)

shadow_pool_size

Size of the shadow pool, a bigger pool size allows for more shadows in the scene but might not fit into GPU memory

TYPE:

enum in ['16', '32', '64', '128', '256', '512', '1024'], default '512'

shadow_ray_count

Amount of shadow ray to trace for each light

TYPE:

int in [1, 4], default 1

shadow_resolution_scale

Resolution percentage of shadow maps

TYPE:

float in [0, 1], default 1.0

shadow_step_count

Amount of shadow map sample per shadow ray

TYPE:

int in [1, 16], default 6

taa_render_samples

Number of samples per pixel for rendering

TYPE:

int in [1, inf], default 64

taa_samples

Number of samples, unlimited if 0

TYPE:

int in [0, inf], default 16

use_bokeh_jittered

Jitter camera position to create accurate blurring using render samples (only for final render)

TYPE:

boolean, default False

use_fast_gi

Use faster global illumination technique for high roughness surfaces

TYPE:

boolean, default False

use_gtao

Enable ambient occlusion to simulate medium scale indirect shadowing

TYPE:

boolean, default False

use_overscan

Internally render past the image border to avoid screen-space effects disappearing

TYPE:

boolean, default False

use_raytracing

Enable the ray-tracing module

TYPE:

boolean, default False

use_shadow_jitter_viewport

Enable jittered shadows on the viewport. (Jittered shadows are always enabled for final renders).

TYPE:

boolean, default False

use_shadows

Enable shadow casting from lights

TYPE:

boolean, default False

use_taa_reprojection

Denoise image using temporal reprojection (can leave some ghosting)

TYPE:

boolean, default True

use_volume_custom_range

Enable custom start and end clip distances for volume computation

TYPE:

boolean, default False

use_volumetric_shadows

Cast shadows from volumetric materials onto volumetric materials (Very expensive)

TYPE:

boolean, default False

volumetric_end

End distance of the volumetric effect

TYPE:

float in [1e-06, inf], default 100.0

volumetric_light_clamp

Maximum light contribution, reducing noise

TYPE:

float in [0, inf], default 0.0

volumetric_ray_depth

Maximum surface intersection count used by the accurate volume intersection method. Will create artifact if it is exceeded. Higher count increases VRAM usage.

TYPE:

int in [1, 16], default 16

volumetric_sample_distribution

Distribute more samples closer to the camera

TYPE:

float in [0, 1], default 0.8

volumetric_samples

Number of steps to compute volumetric effects. Higher step count increase VRAM usage and quality.

TYPE:

int in [1, 256], default 64

volumetric_shadow_samples

Number of samples to compute volumetric shadowing

TYPE:

int in [1, 128], default 16

volumetric_start

Start distance of the volumetric effect

TYPE:

float in [1e-06, inf], default 0.1

volumetric_tile_size

Control the quality of the volumetric effects. Higher resolution uses more memory.

- `1` 1:1 – Full resolution.
- `2` 1:2 – Render this effect at 50% render resolution.
- `4` 1:4 – Render this effect at 25% render resolution.
- `8` 1:8 – Render this effect at 12.5% render resolution.
- `16` 1:16 – Render this effect at 6.25% render resolution.

TYPE:

enum in ['1', '2', '4', '8', '16'], default '8'

classmethod bl_rna_get_subclass(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The RNA type or default when not found.

RETURN TYPE:

`bpy.types.Struct` subclass

classmethod bl_rna_get_subclass_py(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The class or default when not found.

RETURN TYPE:

type

Inherited Properties

- `bpy_struct.id_data`

Inherited Functions

- `bpy_struct.as_pointer`
- `bpy_struct.driver_add`
- `bpy_struct.driver_remove`
- `bpy_struct.get`
- `bpy_struct.id_properties_clear`
- `bpy_struct.id_properties_ensure`
- `bpy_struct.id_properties_ui`
- `bpy_struct.is_property_hidden`
- `bpy_struct.is_property_overridable_library`
- `bpy_struct.is_property_readonly`
- `bpy_struct.is_property_set`
- `bpy_struct.items`
- `bpy_struct.keyframe_delete`
- `bpy_struct.keyframe_insert`
- `bpy_struct.keys`
- `bpy_struct.path_from_id`
- `bpy_struct.path_resolve`
- `bpy_struct.pop`
- `bpy_struct.property_overridable_library_set`
- `bpy_struct.property_unset`
- `bpy_struct.type_recast`
- `bpy_struct.values`

References

- `Scene.eevee`