View3DShading(bpy_struct)

base class — `bpy_struct`

class bpy.types.**View3DShading(bpy_struct)**

> Settings for shading in the 3D viewport
>
> **aov_name**
>> Name of the active Shader AOV
>>
>> **TYPE:**
>>> string, default "", (never None)
>
> **background_color**
>> Color for custom background color
>>
>> **TYPE:**
>>> `mathutils.Color` of 3 items in [0, 1], default (0.05, 0.05, 0.05)
>
> **background_type**
>> Way to display the background
>>
>> - `THEME` Theme – Use the theme for background color.
>> - `WORLD` World – Use the world for background color.
>> - `VIEWPORT` Viewport – Use a custom color limited to this viewport only.
>>
>> **TYPE:**
>>> enum in ['THEME', 'WORLD', 'VIEWPORT'], default 'THEME'
>
> **cavity_ridge_factor**
>> Factor for the cavity ridges
>>
>> **TYPE:**
>>> float in [0, 250], default 1.0
>
> **cavity_type**
>> Way to display the cavity shading
>>
>> - `WORLD` World – Cavity shading computed in world space, useful for larger-scale occlusion.
>> - `SCREEN` Screen – Curvature-based shading, useful for making fine details more visible.
>> - `BOTH` Both – Use both effects simultaneously.
>>
>> **TYPE:**
>>> enum in ['WORLD', 'SCREEN', 'BOTH'], default 'SCREEN'
>
> **cavity_valley_factor**
>> Factor for the cavity valleys
>>
>> **TYPE:**
>>> float in [0, 250], default 1.0
>
> **color_type**
>> Color Type
>>
>> - `MATERIAL` Material – Show material color.
>> - `SINGLE` Single – Show scene in a single color.

- `OBJECT` Object – Show object color.
- `RANDOM` Random – Show random object color.
- `VERTEX` Attribute – Show active color attribute.
- `TEXTURE` Texture – Show the texture from the active image texture node using the active UV map coordinates.

TYPE:

enum in ['MATERIAL', 'SINGLE', 'OBJECT', 'RANDOM', 'VERTEX', 'TEXTURE'], default 'MATERIAL'

curvature_ridge_factor

Factor for the curvature ridges

TYPE:

float in [0, 2], default 1.0

curvature_valley_factor

Factor for the curvature valleys

TYPE:

float in [0, 2], default 1.0

cycles

TYPE:

`CyclesView3DShadingSettings`, (readonly)

light

Lighting Method for Solid/Texture Viewport Shading

- `STUDIO` Studio – Display using studio lighting.
- `MATCAP` MatCap – Display using matcap material and lighting.
- `FLAT` Flat – Display using flat lighting.

TYPE:

enum in ['STUDIO', 'MATCAP', 'FLAT'], default 'STUDIO'

object_outline_color

Color for object outline

TYPE:

`mathutils.Color` of 3 items in [0, 1], default (0.0, 0.0, 0.0)

render_pass

Render Pass to show in the viewport

TYPE:

enum in ['COMBINED', 'EMISSION', 'ENVIRONMENT', 'AO', 'SHADOW', 'TRANSPARENT', 'DIFFUSE_LIGHT', 'DIFFUSE_COLOR', 'SPECULAR_LIGHT', 'SPECULAR_COLOR', 'VOLUME_LIGHT', 'POSITION', 'NORMAL', 'MIST', 'CryptoObject', 'CryptoAsset', 'CryptoMaterial', 'AOV'], default 'COMBINED'

selected_studio_light

Selected StudioLight

TYPE:

`StudioLight`, (readonly)

shadow_intensity

Darkness of shadows

TYPE:

float in [0, 1], default 0.5

show_backface_culling

Use back face culling to hide the back side of faces

TYPE:

boolean, default False

show_cavity

Show Cavity

TYPE:

boolean, default False

show_object_outline

Show Object Outline

TYPE:

boolean, default False

show_shadows

Show Shadow

TYPE:

boolean, default False

show_specular_highlight

Render specular highlights

TYPE:

boolean, default True

show_xray

Show whole scene transparent

TYPE:

boolean, default False

show_xray_wireframe

Show whole scene transparent

TYPE:

boolean, default True

single_color

Color for single color mode

TYPE:

mathutils.Color of 3 items in [0, 1], default (0.8, 0.8, 0.8)

studio_light

Studio lighting setup

TYPE:

enum in ['DEFAULT'], default 'DEFAULT'

studiolight_background_alpha

Show the studiolight in the background

TYPE:

float in [0, 1], default 0.0

studiolight_background_blur

Blur the studiolight in the background

TYPE:

float in [0, 1], default 0.5

studiolight_intensity

Strength of the studiolight

TYPE:

float in [0, inf], default 1.0

studiolight_rotate_z

Rotation of the studiolight around the Z-Axis

TYPE:

float in [-3.14159, 3.14159], default 0.0

type

Method to display/shade objects in the 3D View

TYPE:

enum in Shading Type Items, default 'SOLID'

use_compositor

When to preview the compositor output inside the viewport

- DISABLED Disabled – The compositor is disabled.
- CAMERA Camera – The compositor is enabled only in camera view.
- ALWAYS Always – The compositor is always enabled regardless of the view.

TYPE:

enum in ['DISABLED', 'CAMERA', 'ALWAYS'], default 'DISABLED'

use_dof

Use depth of field on viewport using the values from the active camera

TYPE:

boolean, default False

use_scene_lights

Render lights and light probes of the scene

TYPE:

boolean, default False

use_scene_lights_render

Render lights and light probes of the scene

TYPE:

boolean, default True

use_scene_world

Use scene world for lighting

TYPE:
> boolean, default False

use_scene_world_render

Use scene world for lighting

TYPE:
> boolean, default True

use_studiolight_view_rotation

Make the HDR rotation fixed and not follow the camera

TYPE:
> boolean, default True

use_world_space_lighting

Make the lighting fixed and not follow the camera

TYPE:
> boolean, default False

wireframe_color_type

Wire Color Type

- `THEME` Theme – Show scene wireframes with the theme's wire color.
- `OBJECT` Object – Show object color on wireframe.
- `RANDOM` Random – Show random object color on wireframe.

TYPE:
> enum in ['THEME', 'OBJECT', 'RANDOM'], default 'THEME'

xray_alpha

Amount of opacity to use

TYPE:
> float in [0, 1], default 0.5

xray_alpha_wireframe

Amount of opacity to use

TYPE:
> float in [0, 1], default 0.5

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

- SceneDisplay.shading
- SpaceView3D.shading
- XrSessionSettings.shading