

CompositorNodeColorSpill(CompositorNode)

base classes — `bpy_struct`, `Node`, `NodeInternal`, `CompositorNode`

class `bpy.types.CompositorNodeColorSpill(CompositorNode)`

Remove colors from a blue or green screen, by reducing one RGB channel compared to the others

channel

- `R` `R` – Red spill suppression.
- `G` `G` – Green spill suppression.
- `B` `B` – Blue spill suppression.

TYPE:

enum in ['R', 'G', 'B'], default 'R'

limit_channel

- `R` `R` – Limit by red.
- `G` `G` – Limit by green.
- `B` `B` – Limit by blue.

TYPE:

enum in ['R', 'G', 'B'], default 'R'

limit_method

- `SIMPLE` `Simple` – Simple limit algorithm.
- `AVERAGE` `Average` – Average limit algorithm.

TYPE:

enum in ['SIMPLE', 'AVERAGE'], default 'SIMPLE'

ratio

Scale limit by value

TYPE:

float in [0.5, 1.5], default 0.0

unspill_blue

Blue spillmap scale

TYPE:

float in [0, 1.5], default 0.0

unspill_green

Green spillmap scale

TYPE:

float in [0, 1.5], default 0.0

unspill_red

Red spillmap scale

TYPE:

float in [0, 1.5], default 0.0

use_unspill

Compensate all channels (differently) by hand

TYPE:

boolean, default False

classmethod is_registered_node_type()

True if a registered node type

RETURNS:

Result

RETURN TYPE:

boolean

classmethod input_template(index)

Input socket template

PARAMETERS:

index (*int in $[0, inf]$*) – Index

RETURNS:

result

RETURN TYPE:

`NodeInternalSocketTemplate`

classmethod output_template(index)

Output socket template

PARAMETERS:

index (*int in $[0, inf]$*) – Index

RETURNS:

result

RETURN TYPE:

`NodeInternalSocketTemplate`

update()

classmethod bl_ma_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_ma_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`
- `Node.type`
- `Node.location`
- `Node.location_absolute`
- `Node.width`
- `Node.height`
- `Node.dimensions`
- `Node.name`
- `Node.label`
- `Node.inputs`
- `Node.outputs`
- `Node.internal_links`
- `Node.parent`
- `Node.warning_propagation`
- `Node.use_custom_color`
- `Node.color`
- `Node.color_tag`
- `Node.select`
- `Node.show_options`
- `Node.show_preview`
- `Node.hide`
- `Node.mute`
- `Node.show_texture`
- `Node.bl_idname`
- `Node.bl_label`
- `Node.bl_description`
- `Node.bl_icon`
- `Node.bl_static_type`
- `Node.bl_width_default`
- `Node.bl_width_min`
- `Node.bl_width_max`
- `Node.bl_height_default`
- `Node.bl_height_min`
- `Node.bl_height_max`

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`
- `Node.socket_value_update`
- `Node.is_registered_node_type`
- `Node.poll`
- `Node.update`
- `Node.insert_link`
- `Node.init`
- `Node.copy`
- `Node.free`
- `Node.draw_buttons`
- `Node.draw_buttons_ext`
- `Node.draw_label`
- `Node.debug_zone_body_lazy_function_graph`
- `Node.debug_zone_lazy_function_graph`
- `Node.poll`
- `Node.bl_rna_get_subclass`
- `Node.bl_rna_get_subclass_py`
- `NodeInternal.poll`
- `NodeInternal.poll_instance`
- `NodeInternal.update`
- `NodeInternal.draw_buttons`
- `NodeInternal.draw_buttons_ext`
- `NodeInternal.bl_rna_get_subclass`
- `NodeInternal.bl_rna_get_subclass_py`
- `CompositorNode.tag_need_exec`
- `CompositorNode.poll`
- `CompositorNode.update`
- `CompositorNode.bl_rna_get_subclass`
- `CompositorNode.bl_rna_get_subclass_py`

- `Node.poll_instance`

— — — —

[Previous](#)
[CompositorNodeColorMatte\(CompositorNode\)](#)
[Report issue on this page](#)

Copyright © Blender Authors
Made with [Furo](#)

[CompositorNodeCombHSVA\(CompositorNode\)](#) [Node](#)