Skip to content

Image Operators

bpy.ops.image.**add_render_slot()**

Add a new render slot

bpy.ops.image.**change_frame(*, frame=0)**

Interactively change the current frame number

PARAMETERS:

frame (*int in [-1048574, 1048574], (optional)*) – Frame

bpy.ops.image.**clear_render_border()**

Clear the boundaries of the render region and disable render region

bpy.ops.image.**clear_render_slot()**

Clear the currently selected render slot

bpy.ops.image.**clipboard_copy()**

Copy the image to the clipboard

bpy.ops.image.**clipboard_paste()**

Paste new image from the clipboard

bpy.ops.image.**convert_to_mesh_plane(*, interpolation='Linear', extension='CLIP', alpha_mode='STRAIGHT', use_auto_refresh=True, relative=True, shader='PRINCIPLED', emit_strength=1.0, use_transparency=True, render_method='DITHERED', use_backface_culling=False, show_transparent_back=True, overwrite_material=True, name_from='OBJECT', delete_ref=True)**

Convert selected reference images to textured mesh plane

PARAMETERS:

- **interpolation** (*enum in ['Linear', 'Closest', 'Cubic', 'Smart'], (optional)*) –
  Interpolation, Texture interpolation
  - `Linear` Linear – Linear interpolation.
  - `Closest` Closest – No interpolation (sample closest texel).
  - `Cubic` Cubic – Cubic interpolation.
  - `Smart` Smart – Bicubic when magnifying, else bilinear (OSL only).
- **extension** (*enum in ['CLIP', 'EXTEND', 'REPEAT'], (optional)*) –
  Extension, How the image is extrapolated past its original bounds
  - `CLIP` Clip – Clip to image size and set exterior pixels as transparent.
  - `EXTEND` Extend – Extend by repeating edge pixels of the image.
  - `REPEAT` Repeat – Cause the image to repeat horizontally and vertically.
- **alpha_mode** (*enum in ['STRAIGHT', 'PREMUL', 'CHANNEL_PACKED', 'NONE'], (optional)*) –
  Alpha Mode, Representation of alpha in the image file, to convert to and from when saving and loading the image
  - `STRAIGHT` Straight – Store RGB and alpha channels separately with alpha acting as a mask, also known as unassociated alpha. Commonly used by image editing applications and file formats like PNG..
  - `PREMUL` Premultiplied – Store RGB channels with alpha multiplied in, also known as associated alpha. The natural format for renders ar used by file formats like OpenEXR..
  - `CHANNEL_PACKED` Channel Packed – Different images are packed in the RGB and alpha channels, and they should not affect each other. Channel packing is commonly used by game engines to save memory..
  - `NONE` None – Ignore alpha channel from the file and make image fully opaque.
- **use_auto_refresh** (*boolean, (optional)*) – Auto Refresh, Always refresh image on frame changes

- **use_auto_refresh** (*boolean, (optional)*) – Auto Refresh, Always refresh image on frame changes
- **relative** (*boolean, (optional)*) – Relative Paths, Use relative file paths
- **shader** (*enum in ['PRINCIPLED', 'SHADELESS', 'EMISSION'], (optional)*) –

  Shader, Node shader to use

  - `PRINCIPLED` Principled – Principled shader.
  - `SHADELESS` Shadeless – Only visible to camera and reflections.
  - `EMISSION` Emission – Emission shader.
- **emit_strength** (*float in [0, inf], (optional)*) – Emission Strength, Strength of emission
- **use_transparency** (*boolean, (optional)*) – Use Alpha, Use alpha channel for transparency
- **render_method** (*enum in ['DITHERED', 'BLENDED'], (optional)*) –

  Render Method

  - `DITHERED` Dithered – Allows for grayscale hashed transparency, and compatible with render passes and ray-tracing. Also known as deferred rendering..
  - `BLENDED` Blended – Allows for colored transparency, but incompatible with render passes and ray-tracing. Also known as forward rendering..
- **use_backface_culling** (*boolean, (optional)*) – Backface Culling, Use backface culling to hide the back side of faces
- **show_transparent_back** (*boolean, (optional)*) – Show Backface, Render multiple transparent layers (may introduce transparency sorting problems)
- **overwrite_material** (*boolean, (optional)*) – Overwrite Material, Overwrite existing material with the same name
- **name_from** (*enum in ['OBJECT', 'IMAGE'], (optional)*) –

  Name After, Name for new mesh object and material

  - `OBJECT` Source Object – Name after object source with a suffix.
  - `IMAGE` Source Image – Name from loaded image.
- **delete_ref** (*boolean, (optional)*) – Delete Reference Object, Delete empty image object once mesh plane is created

FILE:

startup/bl_operators/image_as_planes.py:1145

bpy.ops.image.**curves_point_set(*, point='BLACK_POINT', size=1)**

Set black point or white point for curves

PARAMETERS:

- **point** (*enum in ['BLACK_POINT', 'WHITE_POINT'], (optional)*) – Point, Set black point or white point for curves
- **size** (*int in [1, 128], (optional)*) – Sample Size

bpy.ops.image.**cycle_render_slot(*, reverse=False)**

Cycle through all non-void render slots

PARAMETERS:

reverse (*boolean, (optional)*) – Cycle in Reverse

bpy.ops.image.**external_edit(*, filepath='')**

Edit image in an external application

PARAMETERS:

filepath (*string, (optional, never None)*) – filepath

FILE:

startup/bl_operators/image.py:54

bpy.ops.image.**file_browse(*, filepath='', hide_props_region=True, check_existing=False, filter_blender=False, filter_backup=False, filter_image=True, filter_movie=True, filter_python=False, filter_font=False, filter_sound=False, filter_text=False, filter_archive=False, filter_btx=False, filter_collada=False, filter_alembic=False, filter_usd=False, filter_obj=False,**

filter_volume=False, filter_folder=True, filter_blenlib=False, filemode=9, relative_path=True, show_multiview=False, use_multiview=False, display_type='DEFAULT', sort_method='')

Open an image file browser, hold Shift to open the file, Alt to browse containing directory

PARAMETERS:

- **filepath** (*string, (optional, never None)*) – File Path, Path to file
- **hide_props_region** (*boolean, (optional)*) – Hide Operator Properties, Collapse the region displaying the operator settings
- **check_existing** (*boolean, (optional)*) – Check Existing, Check and warn on overwriting existing files
- **filter_blender** (*boolean, (optional)*) – Filter .blend files
- **filter_backup** (*boolean, (optional)*) – Filter .blend files
- **filter_image** (*boolean, (optional)*) – Filter image files
- **filter_movie** (*boolean, (optional)*) – Filter movie files
- **filter_python** (*boolean, (optional)*) – Filter Python files
- **filter_font** (*boolean, (optional)*) – Filter font files
- **filter_sound** (*boolean, (optional)*) – Filter sound files
- **filter_text** (*boolean, (optional)*) – Filter text files
- **filter_archive** (*boolean, (optional)*) – Filter archive files
- **filter_btx** (*boolean, (optional)*) – Filter btx files
- **filter_collada** (*boolean, (optional)*) – Filter COLLADA files
- **filter_alembic** (*boolean, (optional)*) – Filter Alembic files
- **filter_usd** (*boolean, (optional)*) – Filter USD files
- **filter_obj** (*boolean, (optional)*) – Filter OBJ files
- **filter_volume** (*boolean, (optional)*) – Filter OpenVDB volume files
- **filter_folder** (*boolean, (optional)*) – Filter folders
- **filter_blenlib** (*boolean, (optional)*) – Filter Blender IDs
- **filemode** (*int in [1, 9], (optional)*) – File Browser Mode, The setting for the file browser mode to load a .blend file, a library or a special fil
- **relative_path** (*boolean, (optional)*) – Relative Path, Select the file relative to the blend file
- **show_multiview** (*boolean, (optional)*) – Enable Multi-View
- **use_multiview** (*boolean, (optional)*) – Use Multi-View
- **display_type** (*enum in ['DEFAULT', 'LIST_VERTICAL', 'LIST_HORIZONTAL', 'THUMBNAIL'], (optional)*) – Display Type
  - `DEFAULT` Default – Automatically determine display type for files.
  - `LIST_VERTICAL` Short List – Display files as short list.
  - `LIST_HORIZONTAL` Long List – Display files as a detailed list.
  - `THUMBNAIL` Thumbnails – Display files as thumbnails.
- **sort_method** (*enum in [], (optional)*) – File sorting mode

bpy.ops.image.**flip(*, use_flip_x=False, use_flip_y=False)**

Flip the image

PARAMETERS:

- **use_flip_x** (*boolean, (optional)*) – Horizontal, Flip the image horizontally
- **use_flip_y** (*boolean, (optional)*) – Vertical, Flip the image vertically

bpy.ops.image.**import_as_mesh_planes(*, interpolation='Linear', extension='CLIP', alpha_mode='STRAIGHT', use_auto_refresh=True, relative=True, shader='PRINCIPLED', emit_strength=1.0, use_transparency=True, render_method='DITHERED', use_backface_culling=False, show_transparent_back=True, overwrite_material=True, filepath='', align='WORLD', location=(0.0, 0.0, 0.0), rotation=(0.0, 0.0, 0.0), files=None, directory='', filter_image=True, filter_movie=True, filter_folder=True, force_reload=False, image_sequence=False, offset=True, offset_axis='+X', offset_amount=0.1, align_axis='CAM_AX', prev_align_axis='NONE', align_track=False, size_mode='ABSOLUTE', fill_mode='FILL', height=1.0, factor=600.0)**

Create mesh plane(s) from image files with the appropriate aspect ratio

Create mesh plane(s) from image files with the appropriate aspect ratio

PARAMETERS:

- **interpolation** (*enum in ['Linear', 'Closest', 'Cubic', 'Smart'], (optional)*) – Interpolation, Texture interpolation
  - `Linear` Linear – Linear interpolation.
  - `Closest` Closest – No interpolation (sample closest texel).
  - `Cubic` Cubic – Cubic interpolation.
  - `Smart` Smart – Bicubic when magnifying, else bilinear (OSL only).
- **extension** (*enum in ['CLIP', 'EXTEND', 'REPEAT'], (optional)*) – Extension, How the image is extrapolated past its original bounds
  - `CLIP` Clip – Clip to image size and set exterior pixels as transparent.
  - `EXTEND` Extend – Extend by repeating edge pixels of the image.
  - `REPEAT` Repeat – Cause the image to repeat horizontally and vertically.
- **alpha_mode** (*enum in ['STRAIGHT', 'PREMUL', 'CHANNEL_PACKED', 'NONE'], (optional)*) – Alpha Mode, Representation of alpha in the image file, to convert to and from when saving and loading the image
  - `STRAIGHT` Straight – Store RGB and alpha channels separately with alpha acting as a mask, also known as unassociated alpha. Commonly used by image editing applications and file formats like PNG..
  - `PREMUL` Premultiplied – Store RGB channels with alpha multiplied in, also known as associated alpha. The natural format for renders an used by file formats like OpenEXR..
  - `CHANNEL_PACKED` Channel Packed – Different images are packed in the RGB and alpha channels, and they should not affect each other. Channel packing is commonly used by game engines to save memory..
  - `NONE` None – Ignore alpha channel from the file and make image fully opaque.
- **use_auto_refresh** (*boolean, (optional)*) – Auto Refresh, Always refresh image on frame changes
- **relative** (*boolean, (optional)*) – Relative Paths, Use relative file paths
- **shader** (*enum in ['PRINCIPLED', 'SHADELESS', 'EMISSION'], (optional)*) – Shader, Node shader to use
  - `PRINCIPLED` Principled – Principled shader.
  - `SHADELESS` Shadeless – Only visible to camera and reflections.
  - `EMISSION` Emission – Emission shader.
- **emit_strength** (*float in [0, inf], (optional)*) – Emission Strength, Strength of emission
- **use_transparency** (*boolean, (optional)*) – Use Alpha, Use alpha channel for transparency
- **render_method** (*enum in ['DITHERED', 'BLENDED'], (optional)*) – Render Method
  - `DITHERED` Dithered – Allows for grayscale hashed transparency, and compatible with render passes and ray-tracing. Also known as deferred rendering..
  - `BLENDED` Blended – Allows for colored transparency, but incompatible with render passes and ray-tracing. Also known as forward rendering..
- **use_backface_culling** (*boolean, (optional)*) – Backface Culling, Use backface culling to hide the back side of faces
- **show_transparent_back** (*boolean, (optional)*) – Show Backface, Render multiple transparent layers (may introduce transparency sorting problems)
- **overwrite_material** (*boolean, (optional)*) – Overwrite Material, Overwrite existing material with the same name
- **filepath** (*string, (optional, never None)*) – File Path, Filepath used for importing the file
- **align** (*enum in ['WORLD', 'VIEW', 'CURSOR'], (optional)*) – Align
  - `WORLD` World – Align the new object to the world.
  - `VIEW` View – Align the new object to the view.

  - `CURSOR` 3D Cursor – Use the 3D cursor orientation for the new object.

- **location** (`mathutils.Vector` of 3 items in [-inf, inf], (optional)) – Location
- **rotation** (`mathutils.Euler` rotation of 3 items in [-inf, inf], (optional)) – Rotation
- **files** (`bpy_prop_collection` of `OperatorFileListElement`, (optional)) – files
- **directory** (*string, (optional, never None)*) – directory
- **filter_image** (*boolean, (optional)*) – filter_image
- **filter_movie** (*boolean, (optional)*) – filter_movie
- **filter_folder** (*boolean, (optional)*) – filter_folder
- **force_reload** (*boolean, (optional)*) – Force Reload, Force reload the image if it is already opened elsewhere in Blender
- **image_sequence** (*boolean, (optional)*) – Detect Image Sequences, Import sequentially numbered images as an animated image sequence instead of separate planes
- **offset** (*boolean, (optional)*) – Offset Planes, Offset planes from each other. If disabled, multiple planes will be created at the same location
- **offset_axis** (*enum in ['+X', '+Y', '+Z', '-X', '-Y', '-Z'], (optional)*) –

  Offset Direction, How planes are oriented relative to each others' local axis

  - `+X` +X – Side by Side to the Left.
  - `+Y` +Y – Side by Side, Downward.
  - `+Z` +Z – Stacked Above.
  - `-X` -X – Side by Side to the Right.
  - `-Y` -Y – Side by Side, Upward.
  - `-Z` -Z – Stacked Below.

- **offset_amount** (*float in [-inf, inf], (optional)*) – Offset Distance, Set distance between each plane
- **align_axis** (*enum in ['+X', '+Y', '+Z', '-X', '-Y', '-Z', 'CAM', 'CAM_AX'], (optional)*) –

  Align, How to align the planes

  - `+X` +X – Facing positive X.
  - `+Y` +Y – Facing positive Y.
  - `+Z` +Z – Facing positive Z.
  - `-X` -X – Facing negative X.
  - `-Y` -Y – Facing negative Y.
  - `-Z` -Z – Facing negative Z.
  - `CAM` Face Camera – Facing camera.
  - `CAM_AX` Camera's Main Axis – Facing the camera's dominant axis.

- **prev_align_axis** (*enum in ['+X', '+Y', '+Z', '-X', '-Y', '-Z', 'CAM', 'CAM_AX', 'NONE'], (optional)*) –

  prev_align_axis

  - `+X` +X – Facing positive X.
  - `+Y` +Y – Facing positive Y.
  - `+Z` +Z – Facing positive Z.
  - `-X` -X – Facing negative X.
  - `-Y` -Y – Facing negative Y.
  - `-Z` -Z – Facing negative Z.
  - `CAM` Face Camera – Facing camera.
  - `CAM_AX` Camera's Main Axis – Facing the camera's dominant axis.
  - `NONE` Undocumented.

- **align_track** (*boolean, (optional)*) – Track Camera, Add a constraint to make the planes track the camera
- **size_mode** (*enum in ['ABSOLUTE', 'CAMERA', 'DPI', 'DPBU'], (optional)*) –

  Size Mode, Method for computing the plane size

  - `ABSOLUTE` Absolute – Use absolute size.
  - `CAMERA` Scale to Camera Frame – Scale to fit or fill the camera frame.

  - `CAMERA` Scale to Camera Frame – Scale to fit or fill the camera frame.
  - `DPI` Pixels per Inch – Scale based on pixels per inch.
  - `DPBU` Pixels per Blender Unit – Scale based on pixels per Blender Unit.

- **fill_mode** (*enum in ['FILL', 'FIT'], (optional)*) –

  Scale, Method to scale the plane with the camera frame

  - `FILL` Fill – Fill camera frame, spilling outside the frame.
  - `FIT` Fit – Fit entire image within the camera frame.

- **height** (*float in [0.001, inf], (optional)*) – Height, Height of the created plane
- **factor** (*float in [1, inf], (optional)*) – Definition, Number of pixels per inch or Blender Unit

FILE:

startup/bl_operators/image_as_planes.py:867

bpy.ops.image.**invert(*, invert_r=False, invert_g=False, invert_b=False, invert_a=False)**

Invert image’s channels

PARAMETERS:

- **invert_r** (*boolean, (optional)*) – Red, Invert red channel
- **invert_g** (*boolean, (optional)*) – Green, Invert green channel
- **invert_b** (*boolean, (optional)*) – Blue, Invert blue channel
- **invert_a** (*boolean, (optional)*) – Alpha, Invert alpha channel

bpy.ops.image.**match_movie_length()**

Set image’s user’s length to the one of this video

bpy.ops.image.**new(*, name='Untitled', width=1024, height=1024, color=(0.0, 0.0, 0.0, 1.0), alpha=True, generated_type='BLANK', float=False, use_stereo_3d=False, tiled=False)**

Create a new image

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Image data-block name
- **width** (*int in [1, inf], (optional)*) – Width, Image width
- **height** (*int in [1, inf], (optional)*) – Height, Image height
- **color** (*float array of 4 items in [0, inf], (optional)*) – Color, Default fill color
- **alpha** (*boolean, (optional)*) – Alpha, Create an image with an alpha channel
- **generated_type** (enum in Image Generated Type Items, (optional)) – Generated Type, Fill the image with a grid for UV map testing
- **float** (*boolean, (optional)*) – 32-bit Float, Create image with 32-bit floating-point bit depth
- **use_stereo_3d** (*boolean, (optional)*) – Stereo 3D, Create an image with left and right views
- **tiled** (*boolean, (optional)*) – Tiled, Create a tiled image

bpy.ops.image.**open(*, allow_path_tokens=True, filepath='', directory='', files=None, hide_props_region=True, check_existing=False, filter_blender=False, filter_backup=False, filter_image=True, filter_movie=True, filter_python=False, filter_font=False, filter_sound=False, filter_text=False, filter_archive=False, filter_btx=False, filter_collada=False, filter_alembic=False, filter_usd=False, filter_obj=False, filter_volume=False, filter_folder=True, filter_blenlib=False, filemode=9, relative_path=True, show_multiview=False, use_multiview=False, display_type='DEFAULT', sort_method='', use_sequence_detection=True, use_udim_detecting=True)**

Open image

PARAMETERS:

- **allow_path_tokens** (*boolean, (optional)*) – Allow the path to contain substitution tokens
- **filepath** (*string, (optional, never None)*) – File Path, Path to file
- **directory** (*string, (optional, never None)*) – Directory, Directory of the file
- **files** (`bpy_prop_collection` of `OperatorFileListElement`, (optional)) – Files

- **hide_props_region** (*boolean, (optional)*) – Hide Operator Properties, Collapse the region displaying the operator settings
- **check_existing** (*boolean, (optional)*) – Check Existing, Check and warn on overwriting existing files
- **filter_blender** (*boolean, (optional)*) – Filter .blend files
- **filter_backup** (*boolean, (optional)*) – Filter .blend files
- **filter_image** (*boolean, (optional)*) – Filter image files
- **filter_movie** (*boolean, (optional)*) – Filter movie files
- **filter_python** (*boolean, (optional)*) – Filter Python files
- **filter_font** (*boolean, (optional)*) – Filter font files
- **filter_sound** (*boolean, (optional)*) – Filter sound files
- **filter_text** (*boolean, (optional)*) – Filter text files
- **filter_archive** (*boolean, (optional)*) – Filter archive files
- **filter_btx** (*boolean, (optional)*) – Filter btx files
- **filter_collada** (*boolean, (optional)*) – Filter COLLADA files
- **filter_alembic** (*boolean, (optional)*) – Filter Alembic files
- **filter_usd** (*boolean, (optional)*) – Filter USD files
- **filter_obj** (*boolean, (optional)*) – Filter OBJ files
- **filter_volume** (*boolean, (optional)*) – Filter OpenVDB volume files
- **filter_folder** (*boolean, (optional)*) – Filter folders
- **filter_blenlib** (*boolean, (optional)*) – Filter Blender IDs
- **filemode** (*int in [1, 9], (optional)*) – File Browser Mode, The setting for the file browser mode to load a .blend file, a library or a special fil
- **relative_path** (*boolean, (optional)*) – Relative Path, Select the file relative to the blend file
- **show_multiview** (*boolean, (optional)*) – Enable Multi-View
- **use_multiview** (*boolean, (optional)*) – Use Multi-View
- **display_type** (*enum in ['DEFAULT', 'LIST_VERTICAL', 'LIST_HORIZONTAL', 'THUMBNAIL'], (optional)*) – Display Type

  - `DEFAULT` Default – Automatically determine display type for files.
  - `LIST_VERTICAL` Short List – Display files as short list.
  - `LIST_HORIZONTAL` Long List – Display files as a detailed list.
  - `THUMBNAIL` Thumbnails – Display files as thumbnails.

- **sort_method** (*enum in [], (optional)*) – File sorting mode
- **use_sequence_detection** (*boolean, (optional)*) – Detect Sequences, Automatically detect animated sequences in selected images (based o file names)
- **use_udim_detecting** (*boolean, (optional)*) – Detect UDIMs, Detect selected UDIM files and load all matching tiles

bpy.ops.image.**open_images(*, directory='', files=None, relative_path=True, use_sequence_detection=True, use_udim_detection=True)**

Undocumented, consider contributing.

PARAMETERS:

- **directory** (*string, (optional, never None)*) – directory
- **files** (`bpy_prop_collection` of `OperatorFileListElement`, (optional)) – files
- **relative_path** (*boolean, (optional)*) – Use relative path
- **use_sequence_detection** (*boolean, (optional)*) – Use sequence detection
- **use_udim_detection** (*boolean, (optional)*) – Use UDIM detection

FILE:

startup/bl_operators/image.py:238

bpy.ops.image.**pack()**

Pack an image as embedded data into the .blend file bpy.ops.image.**project_apply()**

Project edited image back onto the object

FILE:

startup/bl_operators/image.py:192

bpy.ops.image.**project_edit()**

Edit a snapshot of the 3D Viewport in an external image editor

FILE:

startup/bl_operators/image.py:122

bpy.ops.image.**read_viewlayers()**

Read all the current scene's view layers from cache, as needed

bpy.ops.image.**reload()**

Reload current image from disk

bpy.ops.image.**remove_render_slot()**

Remove the current render slot

bpy.ops.image.**render_border(*, xmin=0, xmax=0, ymin=0, ymax=0, wait_for_input=True)**

Set the boundaries of the render region and enable render region

PARAMETERS:

- **xmin** (*int in [-inf, inf], (optional)*) – X Min
- **xmax** (*int in [-inf, inf], (optional)*) – X Max
- **ymin** (*int in [-inf, inf], (optional)*) – Y Min
- **ymax** (*int in [-inf, inf], (optional)*) – Y Max
- **wait_for_input** (*boolean, (optional)*) – Wait for Input

bpy.ops.image.**replace(*, filepath='', hide_props_region=True, check_existing=False, filter_blender=False, filter_backup=False, filter_image=True, filter_movie=True, filter_python=False, filter_font=False, filter_sound=False, filter_text=False, filter_archive=False, filter_btx=False, filter_collada=False, filter_alembic=False, filter_usd=False, filter_obj=False, filter_volume=False, filter_folder=True, filter_blenlib=False, filemode=9, relative_path=True, show_multiview=False, use_multiview=False, display_type='DEFAULT', sort_method='')**

Replace current image by another one from disk

PARAMETERS:

- **filepath** (*string, (optional, never None)*) – File Path, Path to file
- **hide_props_region** (*boolean, (optional)*) – Hide Operator Properties, Collapse the region displaying the operator settings
- **check_existing** (*boolean, (optional)*) – Check Existing, Check and warn on overwriting existing files
- **filter_blender** (*boolean, (optional)*) – Filter .blend files
- **filter_backup** (*boolean, (optional)*) – Filter .blend files
- **filter_image** (*boolean, (optional)*) – Filter image files
- **filter_movie** (*boolean, (optional)*) – Filter movie files
- **filter_python** (*boolean, (optional)*) – Filter Python files
- **filter_font** (*boolean, (optional)*) – Filter font files
- **filter_sound** (*boolean, (optional)*) – Filter sound files
- **filter_text** (*boolean, (optional)*) – Filter text files
- **filter_archive** (*boolean, (optional)*) – Filter archive files
- **filter_btx** (*boolean, (optional)*) – Filter btx files
- **filter_collada** (*boolean, (optional)*) – Filter COLLADA files
- **filter_alembic** (*boolean, (optional)*) – Filter Alembic files

- **filter_usd** (*boolean, (optional)*) – Filter USD files
- **filter_obj** (*boolean, (optional)*) – Filter OBJ files
- **filter_volume** (*boolean, (optional)*) – Filter OpenVDB volume files
- **filter_folder** (*boolean, (optional)*) – Filter folders
- **filter_blenlib** (*boolean, (optional)*) – Filter Blender IDs
- **filemode** (*int in [1, 9], (optional)*) – File Browser Mode, The setting for the file browser mode to load a .blend file, a library or a special fil
- **relative_path** (*boolean, (optional)*) – Relative Path, Select the file relative to the blend file
- **show_multiview** (*boolean, (optional)*) – Enable Multi-View
- **use_multiview** (*boolean, (optional)*) – Use Multi-View
- **display_type** (*enum in ['DEFAULT', 'LIST_VERTICAL', 'LIST_HORIZONTAL', 'THUMBNAIL'], (optional)*) – Display Type
  - `DEFAULT` Default – Automatically determine display type for files.
  - `LIST_VERTICAL` Short List – Display files as short list.
  - `LIST_HORIZONTAL` Long List – Display files as a detailed list.
  - `THUMBNAIL` Thumbnails – Display files as thumbnails.
- **sort_method** (*enum in [], (optional)*) – File sorting mode

bpy.ops.image.**resize(*, size=(0, 0), all_udims=False)**

Resize the image

PARAMETERS:

- **size** (*int array of 2 items in [1, inf], (optional)*) – Size
- **all_udims** (*boolean, (optional)*) – All UDIM Tiles, Scale all the image’s UDIM tiles

bpy.ops.image.**rotate_orthogonal(*, degrees='90')**

Rotate the image

PARAMETERS:

degrees (*enum in ['90', '180', '270'], (optional)*) –

Degrees, Amount of rotation in degrees (90, 180, 270)

- `90` 90 Degrees – Rotate 90 degrees clockwise.
- `180` 180 Degrees – Rotate 180 degrees clockwise.
- `270` 270 Degrees – Rotate 270 degrees clockwise.

bpy.ops.image.**sample(*, size=1)**

Use mouse to sample a color in current image

PARAMETERS:

size (*int in [1, 128], (optional)*) – Sample Size

bpy.ops.image.**sample_line(*, xstart=0, xend=0, ystart=0, yend=0, flip=False, cursor=5)**

Sample a line and show it in Scope panels

PARAMETERS:

- **xstart** (*int in [-inf, inf], (optional)*) – X Start
- **xend** (*int in [-inf, inf], (optional)*) – X End
- **ystart** (*int in [-inf, inf], (optional)*) – Y Start
- **yend** (*int in [-inf, inf], (optional)*) – Y End
- **flip** (*boolean, (optional)*) – Flip
- **cursor** (*int in [0, inf], (optional)*) – Cursor, Mouse cursor style to use during the modal operator

bpy.ops.image.**save()**

Save the image with current name and settings

bpy.ops.image.**save_all_modified()**

Save all modified images

bpy.ops.image.**save_as(*, save_as_render=False, copy=False, allow_path_tokens=True, filepath='', check_existing=True, filter_blender=False, filter_backup=False, filter_image=True, filter_movie=True, filter_python=False, filter_font=False, filter_sound=False, filter_text=False, filter_archive=False, filter_btx=False, filter_collada=False, filter_alembic=False, filter_usd=False, filter_obj=False, filter_volume=False, filter_folder=True, filter_blenlib=False, filemode=9, relative_path=True, show_multiview=False, use_multiview=False, display_type='DEFAULT', sort_method='')**

Save the image with another name and/or settings

PARAMETERS:

- **save_as_render** (*boolean, (optional)*) – Save As Render, Save image with render color management.For display image formats like PNG, apply view and display transform.For intermediate image formats like OpenEXR, use the default render output color space
- **copy** (*boolean, (optional)*) – Copy, Create a new image file without modifying the current image in Blender
- **allow_path_tokens** (*boolean, (optional)*) – Allow the path to contain substitution tokens
- **filepath** (*string, (optional, never None)*) – File Path, Path to file
- **check_existing** (*boolean, (optional)*) – Check Existing, Check and warn on overwriting existing files
- **filter_blender** (*boolean, (optional)*) – Filter .blend files
- **filter_backup** (*boolean, (optional)*) – Filter .blend files
- **filter_image** (*boolean, (optional)*) – Filter image files
- **filter_movie** (*boolean, (optional)*) – Filter movie files
- **filter_python** (*boolean, (optional)*) – Filter Python files
- **filter_font** (*boolean, (optional)*) – Filter font files
- **filter_sound** (*boolean, (optional)*) – Filter sound files
- **filter_text** (*boolean, (optional)*) – Filter text files
- **filter_archive** (*boolean, (optional)*) – Filter archive files
- **filter_btx** (*boolean, (optional)*) – Filter btx files
- **filter_collada** (*boolean, (optional)*) – Filter COLLADA files
- **filter_alembic** (*boolean, (optional)*) – Filter Alembic files
- **filter_usd** (*boolean, (optional)*) – Filter USD files
- **filter_obj** (*boolean, (optional)*) – Filter OBJ files
- **filter_volume** (*boolean, (optional)*) – Filter OpenVDB volume files
- **filter_folder** (*boolean, (optional)*) – Filter folders
- **filter_blenlib** (*boolean, (optional)*) – Filter Blender IDs
- **filemode** (*int in [1, 9], (optional)*) – File Browser Mode, The setting for the file browser mode to load a .blend file, a library or a special fil
- **relative_path** (*boolean, (optional)*) – Relative Path, Select the file relative to the blend file
- **show_multiview** (*boolean, (optional)*) – Enable Multi-View
- **use_multiview** (*boolean, (optional)*) – Use Multi-View
- **display_type** (*enum in ['DEFAULT', 'LIST_VERTICAL', 'LIST_HORIZONTAL', 'THUMBNAIL'], (optional)*) – Display Type
  - `DEFAULT` Default – Automatically determine display type for files.
  - `LIST_VERTICAL` Short List – Display files as short list.
  - `LIST_HORIZONTAL` Long List – Display files as a detailed list.
  - `THUMBNAIL` Thumbnails – Display files as thumbnails.
- **sort_method** (*enum in [], (optional)*) – File sorting mode

bpy.ops.image.**save_sequence()**

Save a sequence of images

bpy.ops.image.**tile_add(*, number=1002, count=1, label='', fill=True, color=(0.0, 0.0, 0.0, 1.0), generated_type='BLANK', width=1024, height=1024, float=False, alpha=True)**

Adds a tile to the image

PARAMETERS:

- **number** (*int in [1001, 2000], (optional)*) – Number, UDIM number of the tile
- **count** (*int in [1, inf], (optional)*) – Count, How many tiles to add
- **label** (*string, (optional, never None)*) – Label, Optional tile label
- **fill** (*boolean, (optional)*) – Fill, Fill new tile with a generated image
- **color** (*float array of 4 items in [0, inf], (optional)*) – Color, Default fill color
- **generated_type** (enum in Image Generated Type Items, (optional)) – Generated Type, Fill the image with a grid for UV map testing
- **width** (*int in [1, inf], (optional)*) – Width, Image width
- **height** (*int in [1, inf], (optional)*) – Height, Image height
- **float** (*boolean, (optional)*) – 32-bit Float, Create image with 32-bit floating-point bit depth
- **alpha** (*boolean, (optional)*) – Alpha, Create an image with an alpha channel

bpy.ops.image.**tile_fill(*, color=(0.0, 0.0, 0.0, 1.0), generated_type='BLANK', width=1024, height=1024, float=False, alpha=True)**

Fill the current tile with a generated image

PARAMETERS:

- **color** (*float array of 4 items in [0, inf], (optional)*) – Color, Default fill color
- **generated_type** (enum in Image Generated Type Items, (optional)) – Generated Type, Fill the image with a grid for UV map testing
- **width** (*int in [1, inf], (optional)*) – Width, Image width
- **height** (*int in [1, inf], (optional)*) – Height, Image height
- **float** (*boolean, (optional)*) – 32-bit Float, Create image with 32-bit floating-point bit depth
- **alpha** (*boolean, (optional)*) – Alpha, Create an image with an alpha channel

bpy.ops.image.**tile_remove()**

Removes a tile from the image

bpy.ops.image.**unpack(*, method='USE_LOCAL', id='')**

Save an image packed in the .blend file to disk

PARAMETERS:

- **method** (enum in Unpack Method Items, (optional)) – Method, How to unpack
- **id** (*string, (optional, never None)*) – Image Name, Image data-block name to unpack

bpy.ops.image.**view_all(*, fit_view=False)**

View the entire image

PARAMETERS:

fit_view (*boolean, (optional)*) – Fit View, Fit frame to the viewport

bpy.ops.image.**view_center_cursor()**

Center the view so that the cursor is in the middle of the view

bpy.ops.image.**view_cursor_center(*, fit_view=False)**

Set 2D Cursor To Center View location

PARAMETERS:

fit_view (*boolean, (optional)*) – Fit View, Fit frame to the viewport

bpy.ops.image.**view_ndof()**

Use a 3D mouse device to pan/zoom the view

bpy.ops.image.**view_pan(*, offset=(0.0, 0.0))**

Pan the view

PARAMETERS:

offset (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Offset, Offset in floating-point units, 1.0 is the width and height of the image

bpy.ops.image.**view_selected()**

View all selected UVs

bpy.ops.image.**view_zoom(*, factor=0.0, use_cursor_init=True)**

Zoom in/out the image

PARAMETERS:

- **factor** (*float in [-inf, inf], (optional)*) – Factor, Zoom factor, values higher than 1.0 zoom in, lower values zoom out
- **use_cursor_init** (*boolean, (optional)*) – Use Mouse Position, Allow the initial mouse position to be used

bpy.ops.image.**view_zoom_border(*, xmin=0, xmax=0, ymin=0, ymax=0, wait_for_input=True, zoom_out=False)**

Zoom in the view to the nearest item contained in the border

PARAMETERS:

- **xmin** (*int in [-inf, inf], (optional)*) – X Min
- **xmax** (*int in [-inf, inf], (optional)*) – X Max
- **ymin** (*int in [-inf, inf], (optional)*) – Y Min
- **ymax** (*int in [-inf, inf], (optional)*) – Y Max
- **wait_for_input** (*boolean, (optional)*) – Wait for Input
- **zoom_out** (*boolean, (optional)*) – Zoom Out

bpy.ops.image.**view_zoom_in(*, location=(0.0, 0.0))**

Zoom in the image (centered around 2D cursor)

PARAMETERS:

location (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Cursor location in screen coordinates

bpy.ops.image.**view_zoom_out(*, location=(0.0, 0.0))**

Zoom out the image (centered around 2D cursor)

PARAMETERS:

location (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Cursor location in screen coordinates

bpy.ops.image.**view_zoom_ratio(*, ratio=0.0)**

Set zoom ratio of the view

PARAMETERS:

ratio (*float in [-inf, inf], (optional)*) – Ratio, Zoom ratio, 1.0 is 1:1, higher is zoomed in, lower is zoomed out