Skip to content

Clip Operators

bpy.ops.clip.**add_marker(*, location=(0.0, 0.0))**

Place new marker at specified location

PARAMETERS:

location (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Location of marker on frame

bpy.ops.clip.**add_marker_at_click()**

Place new marker at the desired (clicked) position

bpy.ops.clip.**add_marker_move(*, CLIP_OT_add_marker=None, TRANSFORM_OT_translate=None)**

Add new marker and move it on movie

PARAMETERS:

- **CLIP_OT_add_marker** (`CLIP_OT_add_marker` , (optional)) – Add Marker, Place new marker at specified location
- **TRANSFORM_OT_translate** (`TRANSFORM_OT_translate` , (optional)) – Move, Move selected items

bpy.ops.clip.**add_marker_slide(*, CLIP_OT_add_marker=None, TRANSFORM_OT_translate=None)**

Add new marker and slide it with mouse until mouse button release

PARAMETERS:

- **CLIP_OT_add_marker** (`CLIP_OT_add_marker` , (optional)) – Add Marker, Place new marker at specified location
- **TRANSFORM_OT_translate** (`TRANSFORM_OT_translate` , (optional)) – Move, Move selected items

bpy.ops.clip.**apply_solution_scale(*, distance=0.0)**

Apply scale on solution itself to make distance between selected tracks equals to desired

PARAMETERS:

distance (*float in [-inf, inf], (optional)*) – Distance, Distance between selected tracks

bpy.ops.clip.**average_tracks(*, keep_original=True)**

Average selected tracks into active

PARAMETERS:

keep_original (*boolean, (optional)*) – Keep Original, Keep original tracks

bpy.ops.clip.**bundles_to_mesh()**

Create vertex cloud using coordinates of reconstructed tracks

FILE:

startup/bl_operators/clip.py:292

bpy.ops.clip.**camera_preset_add(*, name='', remove_name=False, remove_active=False, use_focal_length=True)**

Add or remove a Tracking Camera Intrinsics Preset

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Name of the preset, used to make the path name
- **remove_name** (*boolean, (optional)*) – remove_name
- **remove_active** (*boolean, (optional)*) – remove_active
- **use_focal_length** (*boolean, (optional)*) – Include Focal Length, Include focal length into the preset

FILE:

startup/bl_operators/presets.py:119

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

Interactively change the current frame number

PARAMETERS:

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

bpy.ops.clip.**clean_tracks(*, frames=0, error=0.0, action='SELECT')**

Clean tracks with high error values or few frames

PARAMETERS:

- **frames** (*int in [0, inf], (optional)*) – Tracked Frames, Affect tracks which are tracked less than the specified number of frames
- **error** (*float in [0, inf], (optional)*) – Reprojection Error, Affect tracks which have a larger reprojection error
- **action** (*enum in ['SELECT', 'DELETE_TRACK', 'DELETE_SEGMENTS'], (optional)*) –

  Action, Cleanup action to execute

  - `SELECT` Select – Select unclean tracks.
  - `DELETE_TRACK` Delete Track – Delete unclean tracks.
  - `DELETE_SEGMENTS` Delete Segments – Delete unclean segments of tracks.

bpy.ops.clip.**clear_solution()**

Clear all calculated data

bpy.ops.clip.**clear_track_path(*, action='REMAINED', clear_active=False)**

Clear tracks after/before current position or clear the whole track

PARAMETERS:

- **action** (*enum in ['UPTO', 'REMAINED', 'ALL'], (optional)*) –

  Action, Clear action to execute

  - `UPTO` Clear Up To – Clear path up to current frame.
  - `REMAINED` Clear Remained – Clear path at remaining frames (after current).
  - `ALL` Clear All – Clear the whole path.

- **clear_active** (*boolean, (optional)*) – Clear Active, Clear active track only instead of all selected tracks

bpy.ops.clip.**constraint_to_fcurve()**

Create F-Curves for object which will copy object's movement caused by this constraint

FILE:

startup/bl_operators/clip.py:530

bpy.ops.clip.**copy_tracks()**

Copy the selected tracks to the internal clipboard

bpy.ops.clip.**create_plane_track()**

Create new plane track out of selected point tracks

bpy.ops.clip.**cursor_set(*, location=(0.0, 0.0))**

Set 2D cursor location

PARAMETERS:

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

bpy.ops.clip.**delete_marker(*, confirm=True)**

Delete marker for current frame from selected tracks

PARAMETERS:

confirm (*boolean, (optional)*) – Confirm, Prompt for confirmation

bpy.ops.clip.**delete_proxy()**

Delete movie clip proxy files from the hard drive

FILE:

startup/bl_operators/clip.py:359

bpy.ops.clip.**delete_track(*, confirm=True)**

Delete selected tracks

PARAMETERS:

confirm (*boolean, (optional)*) – Confirm, Prompt for confirmation

bpy.ops.clip.**detect_features(*, placement='FRAME', margin=16, threshold=0.5, min_distance=120)**

Automatically detect features and place markers to track

PARAMETERS:

- **placement** (*enum in ['FRAME', 'INSIDE_GPENCIL', 'OUTSIDE_GPENCIL'], (optional)*) – Placement, Placement for detected features
  - `FRAME` Whole Frame – Place markers across the whole frame.
  - `INSIDE_GPENCIL` Inside Annotated Area – Place markers only inside areas outlined with the Annotation tool.
  - `OUTSIDE_GPENCIL` Outside Annotated Area – Place markers only outside areas outlined with the Annotation tool.
- **margin** (*int in [0, inf], (optional)*) – Margin, Only features further than margin pixels from the image edges are considered
- **threshold** (*float in [0.0001, inf], (optional)*) – Threshold, Threshold level to consider feature good enough for tracking
- **min_distance** (*int in [0, inf], (optional)*) – Distance, Minimal distance accepted between two features

bpy.ops.clip.**disable_markers(*, action='DISABLE')**

Disable/enable selected markers

PARAMETERS:

action (*enum in ['DISABLE', 'ENABLE', 'TOGGLE'], (optional)*) –

Action, Disable action to execute

- `DISABLE` Disable – Disable selected markers.
- `ENABLE` Enable – Enable selected markers.
- `TOGGLE` Toggle – Toggle disabled flag for selected markers.

bpy.ops.clip.**dopesheet_select_channel(*, location=(0.0, 0.0), extend=False)**

Select movie tracking channel

PARAMETERS:

- **location** (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Mouse location to select channel
- **extend** (*boolean, (optional)*) – Extend, Extend selection rather than clearing the existing selection

bpy.ops.clip.**dopesheet_view_all()**

Reset viewable area to show full keyframe range

bpy.ops.clip.**filter_tracks(*, track_threshold=5.0)**

Filter tracks which has weirdly looking spikes in motion curves

PARAMETERS:

track_threshold (*float in [-inf, inf], (optional)*) – Track Threshold, Filter Threshold to select problematic tracks

FILE:

startup/bl_operators/clip.py:206

startup/bl_operators/clip.py:266

bpy.ops.clip.**frame_jump(*, position='PATHSTART')**

Jump to special frame

PARAMETERS:

position (*enum in ['PATHSTART', 'PATHEND', 'FAILEDPREV', 'FAILNEXT'], (optional)*) –

Position, Position to jump to

- `PATHSTART` Path Start – Jump to start of current path.
- `PATHEND` Path End – Jump to end of current path.
- `FAILEDPREV` Previous Failed – Jump to previous failed frame.
- `FAILNEXT` Next Failed – Jump to next failed frame.

bpy.ops.clip.**graph_center_current_frame()**

Scroll view so current frame would be centered

bpy.ops.clip.**graph_delete_curve(*, confirm=True)**

Delete track corresponding to the selected curve

PARAMETERS:

confirm (*boolean, (optional)*) – Confirm, Prompt for confirmation

bpy.ops.clip.**graph_delete_knot()**

Delete curve knots

bpy.ops.clip.**graph_disable_markers(*, action='DISABLE')**

Disable/enable selected markers

PARAMETERS:

action (*enum in ['DISABLE', 'ENABLE', 'TOGGLE'], (optional)*) –

Action, Disable action to execute

- `DISABLE` Disable – Disable selected markers.
- `ENABLE` Enable – Enable selected markers.
- `TOGGLE` Toggle – Toggle disabled flag for selected markers.

bpy.ops.clip.**graph_select(*, location=(0.0, 0.0), extend=False)**

Select graph curves

PARAMETERS:

- **location** (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Mouse location to select nearest entity
- **extend** (*boolean, (optional)*) – Extend, Extend selection rather than clearing the existing selection

bpy.ops.clip.**graph_select_all_markers(*, action='TOGGLE')**

Change selection of all markers of active track

PARAMETERS:

action (*enum in ['TOGGLE', 'SELECT', 'DESELECT', 'INVERT'], (optional)*) –

Action, Selection action to execute

- `TOGGLE` Toggle – Toggle selection for all elements.
- `SELECT` Select – Select all elements.
- `DESELECT` Deselect – Deselect all elements.
- `INVERT` Invert – Invert selection of all elements.

bpy.ops.clip.**graph_select_box(*, xmin=0, xmax=0, ymin=0, ymax=0, wait_for_input=True, deselect=False, extend=True)**

Select curve points using box selection

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
- **deselect** (*boolean, (optional)*) – Deselect, Deselect rather than select items
- **extend** (*boolean, (optional)*) – Extend, Extend selection instead of deselecting everything first

bpy.ops.clip.**graph_view_all()**

View all curves in editor

bpy.ops.clip.**hide_tracks(*, unselected=False)**

Hide selected tracks

PARAMETERS:

unselected (*boolean, (optional)*) – Unselected, Hide unselected tracks

bpy.ops.clip.**hide_tracks_clear()**

Clear hide selected tracks

bpy.ops.clip.**join_tracks()**

Join selected tracks

bpy.ops.clip.**keyframe_delete()**

Delete a keyframe from selected tracks at current frame

bpy.ops.clip.**keyframe_insert()**

Insert a keyframe to selected tracks at current frame

bpy.ops.clip.**lock_selection_toggle()**

Toggle Lock Selection option of the current clip editor

bpy.ops.clip.**lock_tracks(*, action='LOCK')**

Lock/unlock selected tracks

PARAMETERS:

action (*enum in ['LOCK', 'UNLOCK', 'TOGGLE'], (optional)*) –

Action, Lock action to execute

- `LOCK` Lock – Lock selected tracks.
- `UNLOCK` Unlock – Unlock selected tracks.
- `TOGGLE` Toggle – Toggle locked flag for selected tracks.

bpy.ops.clip.**mode_set(*, mode='TRACKING')**

Set the clip interaction mode

PARAMETERS:

mode (enum in Clip Editor Mode Items, (optional)) – Mode

bpy.ops.clip.**new_image_from_plane_marker()**

Create new image from the content of the plane marker bpy.ops.clip.**open(*, 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='')**

Load a sequence of frames or a movie file

PARAMETERS:

- **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 ['DEFAULT', 'FILE_SORT_ALPHA', 'FILE_SORT_EXTENSION', 'FILE_SORT_TIME', 'FILE_SORT_SIZE', 'ASSET_CATALOG'], (optional)*) –

  File sorting mode

  - `DEFAULT` Default – Automatically determine sort method for files.
  - `FILE_SORT_ALPHA` Name – Sort the file list alphabetically.
  - `FILE_SORT_EXTENSION` Extension – Sort the file list by extension/type.
  - `FILE_SORT_TIME` Modified Date – Sort files by modification time.
  - `FILE_SORT_SIZE` Size – Sort files by size.
  - `ASSET_CATALOG` Asset Catalog – Sort the asset list so that assets in the same catalog are kept together. Within a single catalog, asse

are ordered by name. The catalogs are in order of the flattened catalog hierarchy..

bpy.ops.clip.**paste_tracks()**

Paste tracks from the internal clipboard

bpy.ops.clip.**prefetch()**

Prefetch frames from disk for faster playback/tracking

bpy.ops.clip.**rebuild_proxy()**

Rebuild all selected proxies and timecode indices in the background

bpy.ops.clip.**refine_markers(*, backwards=False)**

Refine selected markers positions by running the tracker from track's reference to current frame

PARAMETERS:

backwards (*boolean, (optional)*) – Backwards, Do backwards tracking

bpy.ops.clip.**reload()**

Reload clip

bpy.ops.clip.**select(*, extend=False, deselect_all=False, location=(0.0, 0.0))**

Select tracking markers

PARAMETERS:

- **extend** (*boolean, (optional)*) – Extend, Extend selection rather than clearing the existing selection
- **deselect_all** (*boolean, (optional)*) – Deselect On Nothing, Deselect all when nothing under the cursor
- **location** (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Mouse location in normalized coordinates, 0.0 to 1.0 is within the image bounds

bpy.ops.clip.**select_all(*, action='TOGGLE')**

Change selection of all tracking markers

PARAMETERS:

action (*enum in ['TOGGLE', 'SELECT', 'DESELECT', 'INVERT'], (optional)*) –

Action, Selection action to execute

- `TOGGLE` Toggle – Toggle selection for all elements.
- `SELECT` Select – Select all elements.
- `DESELECT` Deselect – Deselect all elements.
- `INVERT` Invert – Invert selection of all elements.

bpy.ops.clip.**select_box(*, xmin=0, xmax=0, ymin=0, ymax=0, wait_for_input=True, mode='SET')**

Select markers using box selection

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
- **mode** (*enum in ['SET', 'ADD', 'SUB'], (optional)*) –

  Mode

  - `SET` Set – Set a new selection.
  - `ADD` Extend – Extend existing selection.
  - `SUB` Subtract – Subtract existing selection.

  - SUB Subtract – Subtract existing selection.

bpy.ops.clip.**select_circle(*, x=0, y=0, radius=25, wait_for_input=True, mode='SET')**

Select markers using circle selection

PARAMETERS:

- **x** (*int in [-inf, inf], (optional)*) – X
- **y** (*int in [-inf, inf], (optional)*) – Y
- **radius** (*int in [1, inf], (optional)*) – Radius
- **wait_for_input** (*boolean, (optional)*) – Wait for Input
- **mode** (*enum in ['SET', 'ADD', 'SUB'], (optional)*) –

  Mode

  - `SET` Set – Set a new selection.
  - `ADD` Extend – Extend existing selection.
  - `SUB` Subtract – Subtract existing selection.

bpy.ops.clip.**select_grouped(*, group='ESTIMATED')**

Select all tracks from specified group

PARAMETERS:

group (*enum in ['KEYFRAMED', 'ESTIMATED', 'TRACKED', 'LOCKED', 'DISABLED', 'COLOR', 'FAILED'], (optional)*) –

Action, Clear action to execute

- `KEYFRAMED` Keyframed Tracks – Select all keyframed tracks.
- `ESTIMATED` Estimated Tracks – Select all estimated tracks.
- `TRACKED` Tracked Tracks – Select all tracked tracks.
- `LOCKED` Locked Tracks – Select all locked tracks.
- `DISABLED` Disabled Tracks – Select all disabled tracks.
- `COLOR` Tracks with Same Color – Select all tracks with same color as active track.
- `FAILED` Failed Tracks – Select all tracks which failed to be reconstructed.

bpy.ops.clip.**select_lasso(*, path=None, use_smooth_stroke=False, smooth_stroke_factor=0.75, smooth_stroke_radius=35, mode='SET')**

Select markers using lasso selection

PARAMETERS:

- **path** (`bpy_prop_collection` of `OperatorMousePath`, (optional)) – Path
- **use_smooth_stroke** (*boolean, (optional)*) – Stabilize Stroke, Selection lags behind mouse and follows a smoother path
- **smooth_stroke_factor** (*float in [0.5, 0.99], (optional)*) – Smooth Stroke Factor, Higher values gives a smoother stroke
- **smooth_stroke_radius** (*int in [10, 200], (optional)*) – Smooth Stroke Radius, Minimum distance from last point before selection continues
- **mode** (*enum in ['SET', 'ADD', 'SUB'], (optional)*) –

  Mode

  - `SET` Set – Set a new selection.
  - `ADD` Extend – Extend existing selection.
  - `SUB` Subtract – Subtract existing selection.

bpy.ops.clip.**set_active_clip()**

Undocumented, consider contributing.

FILE:

startup/bl_operators/clip.py:221

bpy.ops.clip.**set_axis(*, axis='X')**

Set the direction of a scene axis by rotating the camera (or its parent if present). This assumes that the selected track lies on a real axis connecting it to the origin

PARAMETERS:

axis (*enum in ['X', 'Y'], (optional)*) –

Axis, Axis to use to align bundle along

- `X` X – Align bundle align X axis.
- `Y` Y – Align bundle align Y axis.

bpy.ops.clip.**set_origin(*, use_median=False)**

Set active marker as origin by moving camera (or its parent if present) in 3D space

PARAMETERS:

use_median (*boolean, (optional)*) – Use Median, Set origin to median point of selected bundles

bpy.ops.clip.**set_plane(*, plane='FLOOR')**

Set plane based on 3 selected bundles by moving camera (or its parent if present) in 3D space

PARAMETERS:

plane (*enum in ['FLOOR', 'WALL'], (optional)*) –

Plane, Plane to be used for orientation

- `FLOOR` Floor – Set floor plane.
- `WALL` Wall – Set wall plane.

bpy.ops.clip.**set_scale(*, distance=0.0)**

Set scale of scene by scaling camera (or its parent if present)

PARAMETERS:

distance (*float in [-inf, inf], (optional)*) – Distance, Distance between selected tracks

bpy.ops.clip.**set_scene_frames()**

Set scene’s start and end frame to match clip’s start frame and length

bpy.ops.clip.**set_solution_scale(*, distance=0.0)**

Set object solution scale using distance between two selected tracks

PARAMETERS:

distance (*float in [-inf, inf], (optional)*) – Distance, Distance between selected tracks

bpy.ops.clip.**set_solver_keyframe(*, keyframe='KEYFRAME_A')**

Set keyframe used by solver

PARAMETERS:

keyframe (*enum in ['KEYFRAME_A', 'KEYFRAME_B'], (optional)*) – Keyframe, Keyframe to set

bpy.ops.clip.**set_viewport_background()**

Set current movie clip as a camera background in 3D Viewport (works only when a 3D Viewport is visible)

FILE:

startup/bl_operators/clip.py:420

bpy.ops.clip.**setup_tracking_scene()**

Prepare scene for compositing 3D objects into this footage

FILE:

startup/bl_operators/clip.py:990

bpy.ops.clip.**slide_marker(*, offset=(0.0, 0.0))**

Slide marker areas

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.clip.**slide_plane_marker()**

Slide plane marker areas

bpy.ops.clip.**solve_camera()**

Solve camera motion from tracks

bpy.ops.clip.**stabilize_2d_add()**

Add selected tracks to 2D translation stabilization

bpy.ops.clip.**stabilize_2d_remove()**

Remove selected track from translation stabilization

bpy.ops.clip.**stabilize_2d_rotation_add()**

Add selected tracks to 2D rotation stabilization

bpy.ops.clip.**stabilize_2d_rotation_remove()**

Remove selected track from rotation stabilization

bpy.ops.clip.**stabilize_2d_rotation_select()**

Select tracks which are used for rotation stabilization

bpy.ops.clip.**stabilize_2d_select()**

Select tracks which are used for translation stabilization

bpy.ops.clip.**track_color_preset_add(*, name='', remove_name=False, remove_active=False)**

Add or remove a Clip Track Color Preset

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Name of the preset, used to make the path name
- **remove_name** (*boolean, (optional)*) – remove_name
- **remove_active** (*boolean, (optional)*) – remove_active

FILE:

startup/bl_operators/presets.py:119

bpy.ops.clip.**track_copy_color()**

Copy color to all selected tracks

bpy.ops.clip.**track_markers(*, backwards=False, sequence=False)**

Track selected markers

PARAMETERS:

- **backwards** (*boolean, (optional)*) – Backwards, Do backwards tracking
- **sequence** (*boolean, (optional)*) – Track Sequence, Track marker during image sequence rather than single image

bpy.ops.clip.**track_settings_as_default()**

Copy tracking settings from active track to default settings

FILE:

FILE:

startup/bl_operators/clip.py:1019

bpy.ops.clip.**track_settings_to_track()**

Copy tracking settings from active track to selected tracks

FILE:

startup/bl_operators/clip.py:1068

bpy.ops.clip.**track_to_empty()**

Create an Empty object which will be copying movement of active track

FILE:

startup/bl_operators/clip.py:268

bpy.ops.clip.**tracking_object_new()**

Add new object for tracking

bpy.ops.clip.**tracking_object_remove()**

Remove object for tracking

bpy.ops.clip.**tracking_settings_preset_add(*, name='', remove_name=False, remove_active=False)**

Add or remove a motion tracking settings preset

PARAMETERS:

- **name** (*string, (optional, never None)*) – Name, Name of the preset, used to make the path name
- **remove_name** (*boolean, (optional)*) – remove_name
- **remove_active** (*boolean, (optional)*) – remove_active

FILE:

startup/bl_operators/presets.py:119

bpy.ops.clip.**update_image_from_plane_marker()**

Update current image used by plane marker from the content of the plane marker

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

View whole image with markers

PARAMETERS:

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

bpy.ops.clip.**view_center_cursor()**

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

bpy.ops.clip.**view_ndof()**

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

bpy.ops.clip.**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.clip.**view_selected()**

View all selected elements

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

Zoom in/out the view

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.clip.**view_zoom_in(*, location=(0.0, 0.0))**

Zoom in the view

PARAMETERS:

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

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

Zoom out the view

PARAMETERS:

location (`mathutils.Vector` of 2 items in [-inf, inf], (optional)) – Location, Cursor location in normalized (0.0 to 1.0) coordinates

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

Set the zoom ratio (based on clip size)

PARAMETERS:

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