

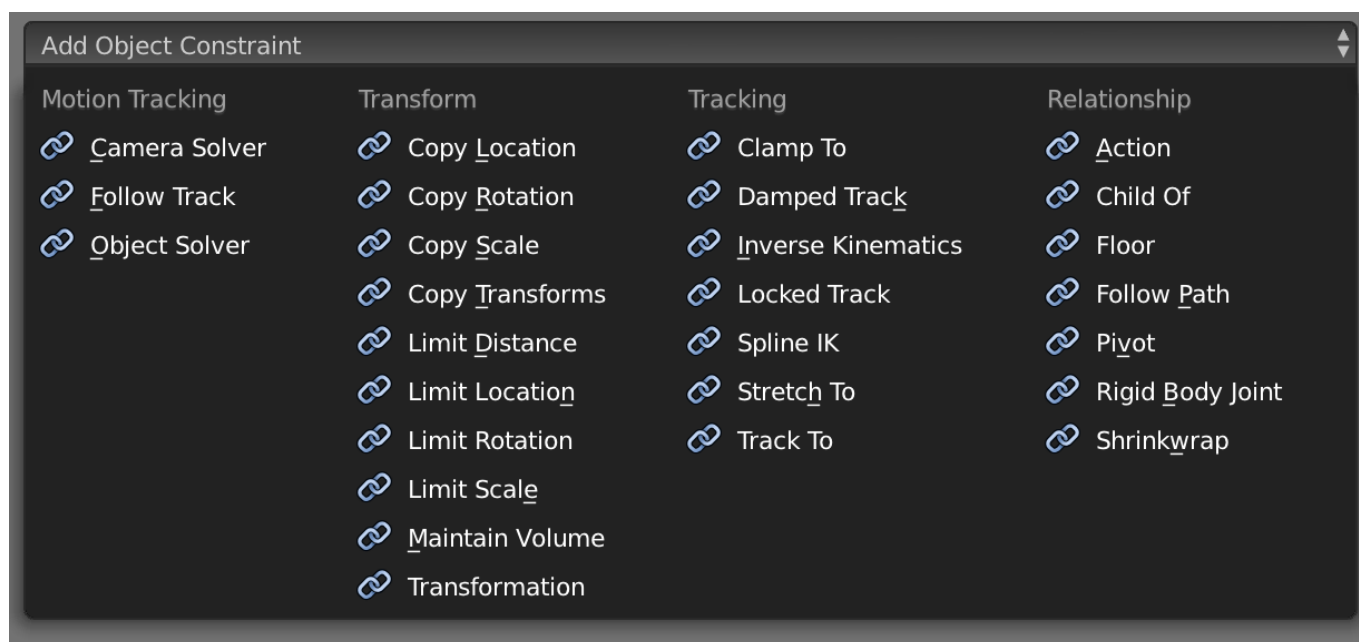
7.2.2 Rigging - Constraints - Interface

Adding/Removing a Constraint.....	1
Header.....	2
Target.....	3
Space.....	4
Target Space & Owner Space.....	5
Space Types.....	5
Influence.....	6
The Constraints Stack.....	6

Adding/Removing a Constraint

To add a constraint in the *Constraints Panel*:

Click on the “Add Constraint” drop-down box.



To add a constraint in the 3D view:

Select the object you would like to constrain. Press **Ctrl-Shift-C** and choose a constraint from the pop-up menu. If the chosen constraint needs a target, Blender will add an empty automatically as the target and position it at the center of the constrained object.

To add a constraint in the 3D view and simultaneously give it a target:

Select the target first and then shift-select the object you would like to constrain. Press **Ctrl-Shift-C** and choose a constraint from the pop-up menu.

To remove a constraint:

Click on the “X” button in the *header*.

To remove all constraints from all selected object(s):

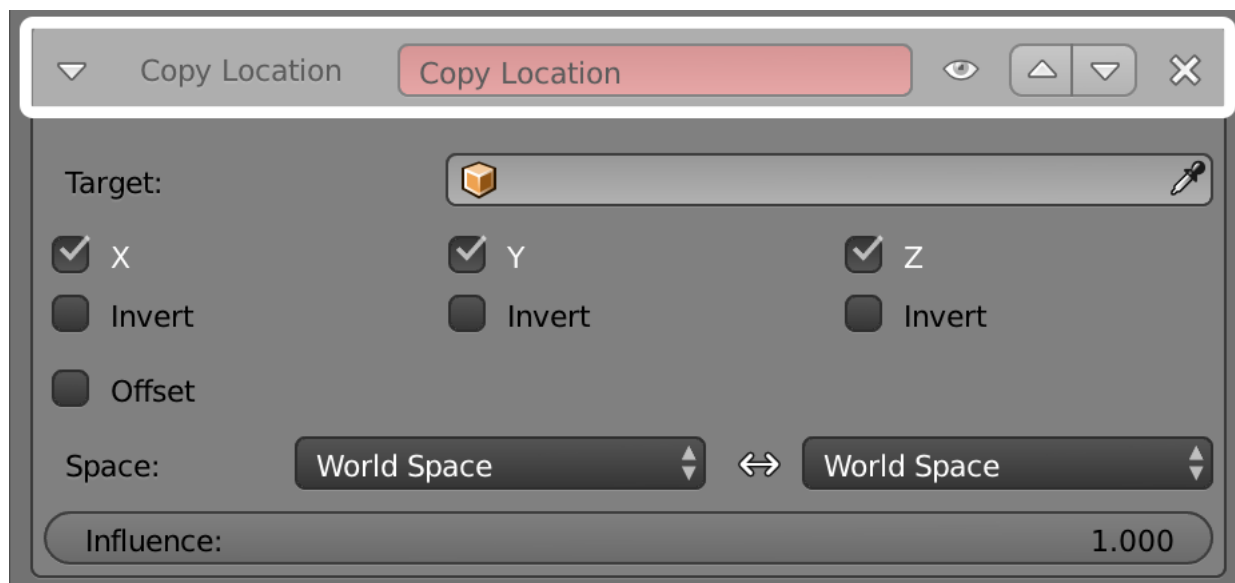
Click **Object** ▸ **Constraints** ▸ **Clear Object Constraints** in the 3D View Header.

or **Pose** ▸ **Constraints** ▸ **Clear Pose Constraints** (for bone constraints).

or press **Ctrl-Alt-C**.

Header

Every constraint has a header. The interface elements of the header are explained below using a Copy Location constraint as an example.



A Header sits at the top of every constraint.

Expansion Arrow (pointing down or right)

Show or Hide the settings of the constraint. Tidy up the *constraint stack* by hiding constraints that don't currently need attention. Constraints will continue to affect the scene even when hidden.

“Copy Location” (first occurrence)

The type of constraint. This is determined at the time the constraint is created.

“Copy Location” (second occurrence)

Give the constraint a meaningful name in this field, something that describes its intent. Meaningful names help you and your team members understand what each constraint is supposed to do.

The **red** background is a warning that the constraint is not yet functional. The background will turn **gray** when the constraint is functioning. When this Copy Location constraint has a valid target in the “Target Field” it will turn gray and begin to function.

Eyeball (open or closed)

Enable or Disable (Mute/Unmute) the constraint. Disabling a constraint will stop its affect on the scene.

Disabling a constraint is useful for turning off a constraint without losing all of its settings. Disabling means you can enable the constraint at a later time with the settings intact. Disabling is similar to setting the *influence* slider to 0.0.

Up/Down Arrows

Move a constraint up or down in the *constraint stack*. Since the stack is evaluated from top to bottom,

moving a constraint in the stack can significantly affect the final outcome of the stack.

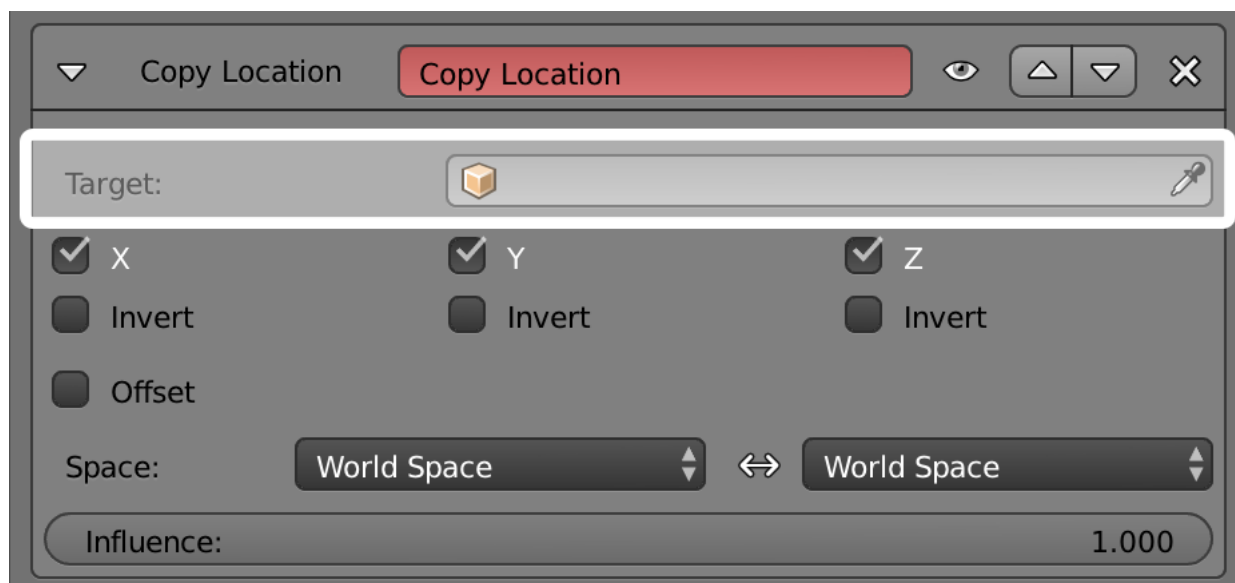
If there is only one constraint in the stack, the arrows will not be drawn. If the constraint is at the top of the stack, only the down arrow will be drawn. If the constraint is at the bottom of the stack, only the up arrow will be drawn.

X

Delete the constraint from the stack. The settings will be lost. The constraint will no longer affect the final outcome of the stack.

Target

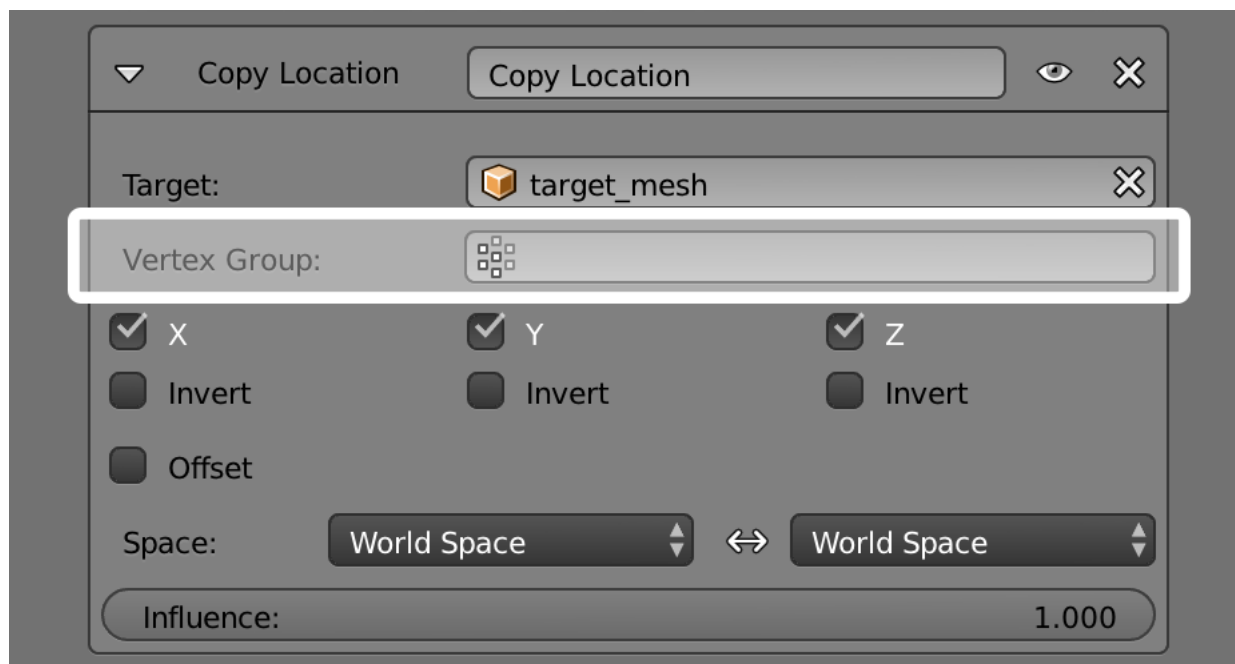
The Target field lets you link the constraint to a Target object of your choosing. This link provides data to the constraint so that it can begin to function. For example, the Copy Location Constraint needs location data to function. Fill in the Target field, and the Copy Location constraint will begin to use location data from the Target object.



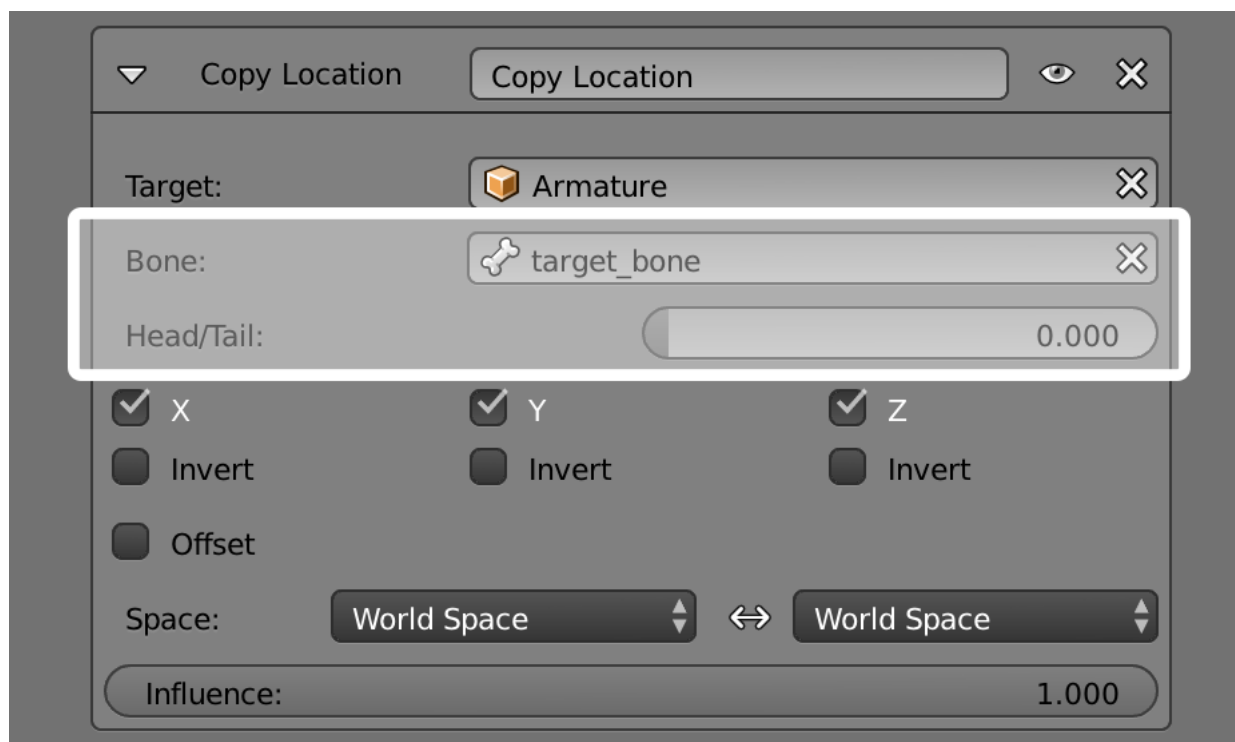
The Target field must be filled in for the constraint to function.

By default, the Target will use the Object Center as the target point.

If the Target field links to a Mesh or Lattice object, a Vertex Group field will appear. Enter the name of a vertex group and the constraint will target the median point of this vertex group instead of the object center.



If the Target field links to an Armature, a Bone field will appear along with a Head/Tail slider. Enter the name of a bone and the constraint will target the bone instead of the entire armature object center. Slide the slider and the constraint will target the head, the tail or somewhere inbetween.

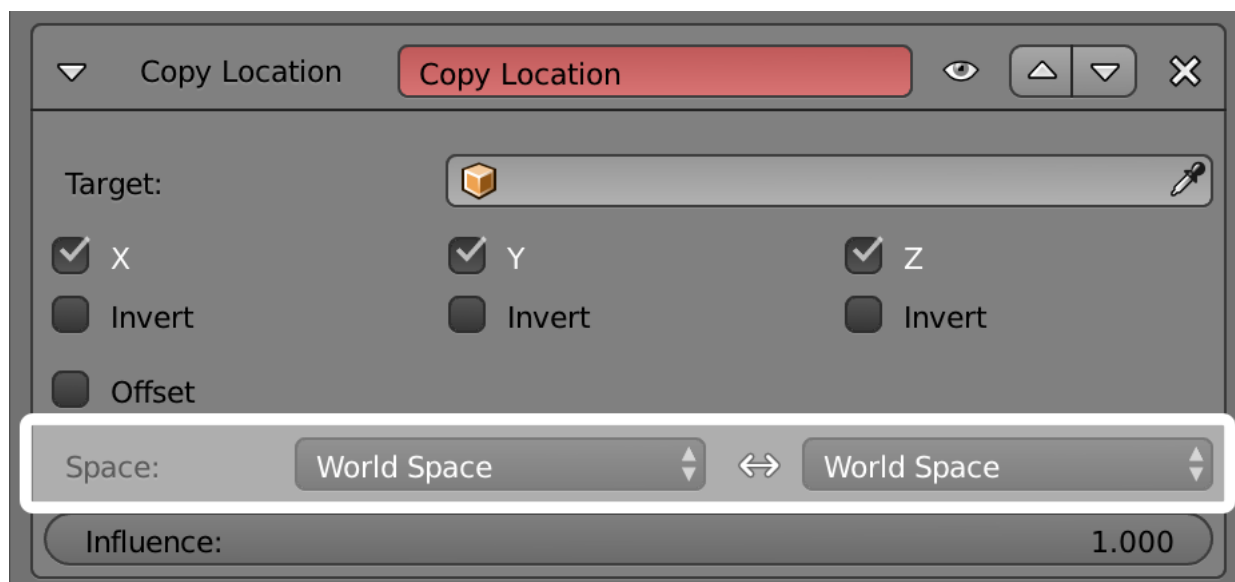


Space

Constraints need a frame of reference in order to function. This frame of reference is called the “space” of the constraint. Choosing one space vs. another will change this frame of reference and substantially alter the

behavior of a constraint.

To understand how changing the space will change the behavior of the constraint, consider experimenting with two empties. Make sure they display as arrows so that you can see the local axes for each empty. Make sure to size one empty a little larger than the other so that they are both always visible even if directly on top of each other. Then add a constraint to one empty that targets the other and experiment thoroughly by moving, rotating and scaling the target in many different ways.



This constraint is set to use World Space as the frame of reference for both its Target Space and its Owner Space.

Target Space & Owner Space

The space used to evaluate the target of the constraint is called the Target Space. The space used to evaluate the constrained object (the object that owns the constraint) is called the owner space. Hover over the space drop-down box (or boxes) to learn whether it affects the space of the target or the space of the owner.

Some constraints don't use Target or Owner space, so there won't be a drop-down box. Some constraints use only Target or only Owner space, so there will only be one drop-down box. Some constraints (like the Copy Location constraint above) use both Target AND Owner space, so there will be two drop-down boxes.

When a constraint uses both Target and Owner space, the Target and Owner can be any combination of space types.

Space Types

World Space

In this space type the world is the frame of reference for the object (or bone). Location is relative to the world origin. Rotation and Scale are oriented to the world axes. Transformations to the object, the object's parent and any other constraints higher up in the constraint stack are all taken into account.

Local Space

In this space type the parent of the object (or bone) is the frame of reference. Location is relative to the parent object origin. Rotation and Scale are oriented to the parent object axes. Only transformations to the object itself are taken into account. Transformations to the object's parent and any other constraints higher up in the constraint stack are NOT taken into account.

Local With Parent (bones only)

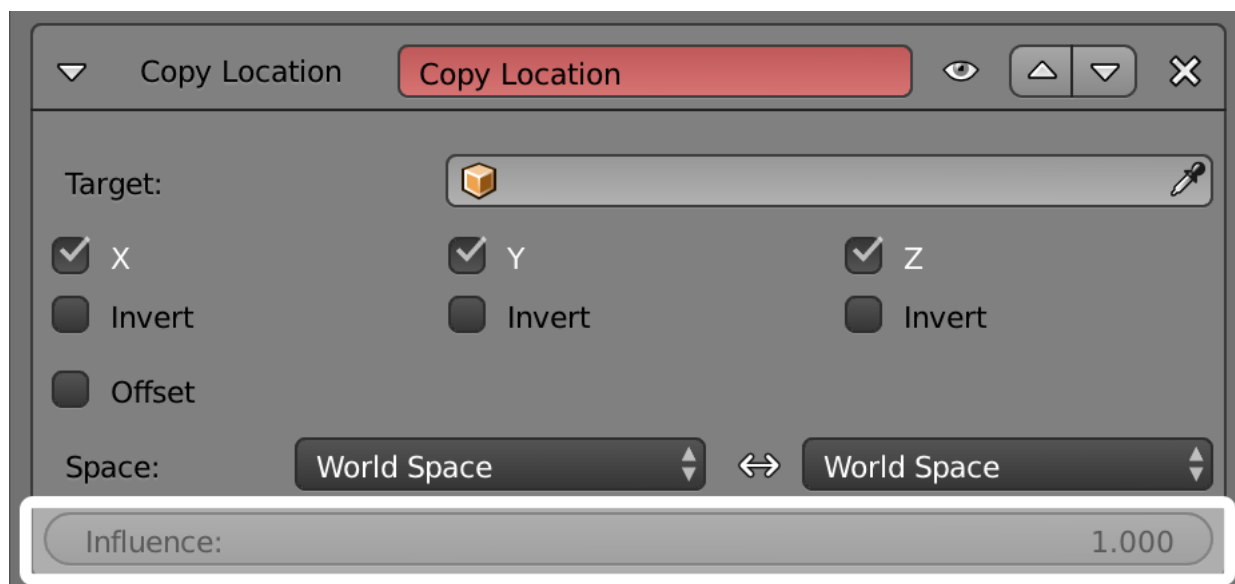
The bone properties are evaluated in its own local space, *including* the transformations due to a possible parent relationship (i.e. due to the chain’s transformations above the bone).

Pose Space (bones only)

The bone properties are evaluated in the armature object local space (i.e. independently from the armature transformations in *Object* mode). Hence, if the armature object has null transformations, *Pose Space* will have the same effect as *World Space*.

Influence

The influence slider determines how much the constraint will affect the constrained object.



An influence of **0.0** will have no effect. An influence of **1.0** will have the full effect.

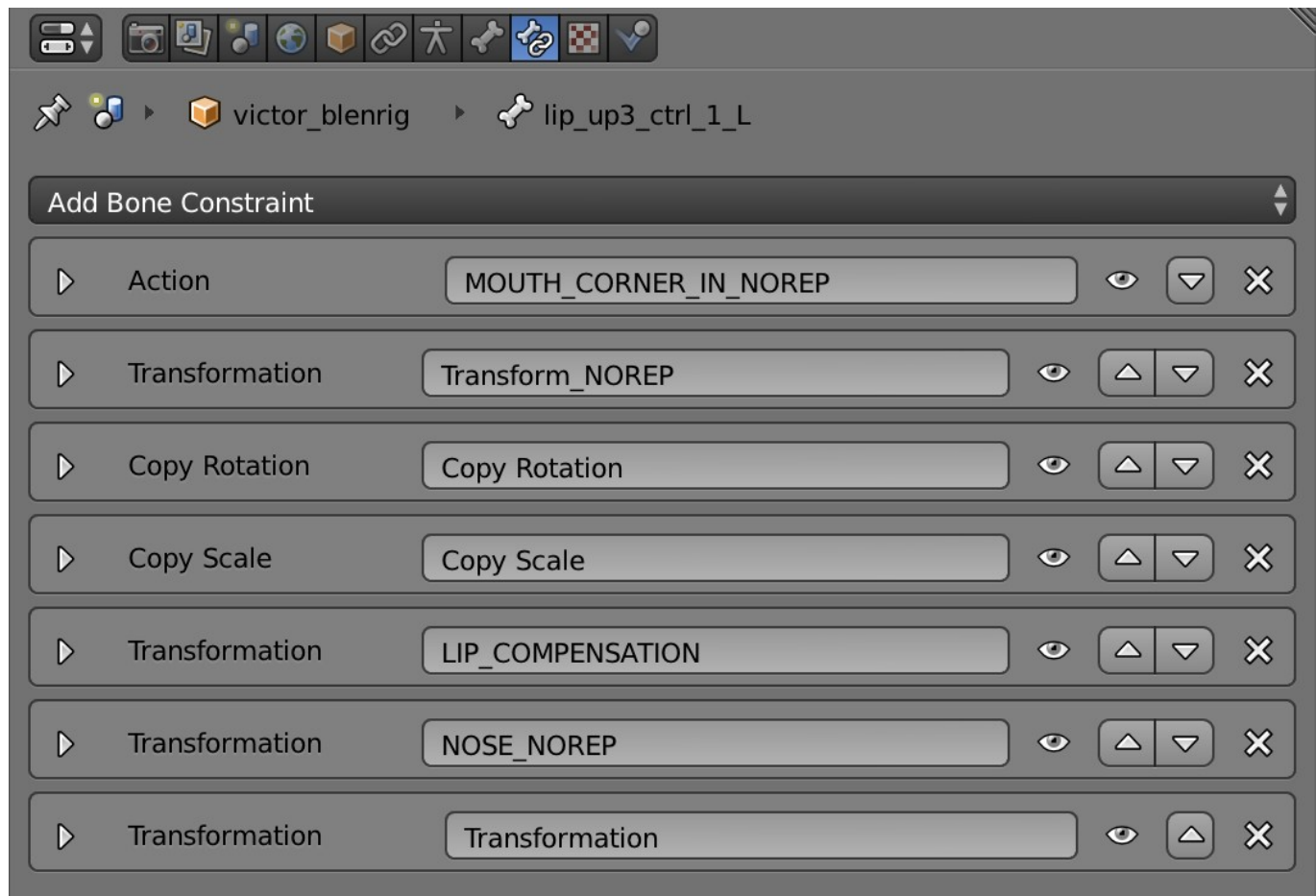
Values between 0.0 and 1.0, will have a partial effect, but be careful. These partial effects can be difficult to control, especially as the *constraint stack* grows in complexity.

The influence value is animatable, allowing constraints to be turned off, or or partially on as needed. (see

The Constraints Stack

The combination of all the constraints that affect an object are called the Constraints Stack. The Stack is in the Constraints panel, below the “Add Constraint” drop-down box.

Constraints in the stack are evaluated from top to bottom. The order of each constraint has a substantial impact on the final outcome of the stack. Changing the order of the constraints can change the behavior of the entire stack.



The 7 constraints in this example stack are evaluated from top to bottom starting with the “Action” constraint and ending with the final “Transformation” constraint.

To change the order of a constraint use the up/down arrows in the *header*.