

# Dealing with inconsistencies between MakeHuman's armature and Vicon™ BodyBuilder's ACCLAIM motion file skeleton

*Note: This tutorial assumes that the reader has basic working knowledge of Blender (v2.76) and MakeHuman (v1.0.2).*

## Approach 1: Using “MakeWalk” add-on

Once the MHX (MakeHuman eXchange) file containing the 3D character is imported into Blender, the MakeWalk add-on can be used to retarget<sup>1</sup> the armature of the character with a BVH motion file.

However, in our case, because of different skeleton definitions across the two software packages, the source armature cannot be automatically identified, causing an error (see Figure 1).



Figure 1: Blender add-on MakeWalk cannot automatically identify the source armature

<sup>1</sup> Retargeting is the procedure in which the movement from the motion capture of a human volunteer is “transferred” onto the armature inside a 3D character.

## Approach 2: Using “Mocap tools” add-on

A different approach is to use the “Mocap tools” add-on. This approach requires a hierarchy mapping between the 3D character’s armature and the skeleton of the motion file (see Figure 2, right part of image). Once the hierarchy is set up, retargeting will take place once the “Advanced retargeting” option is ticked (see Figure 2, bottom right of the image).

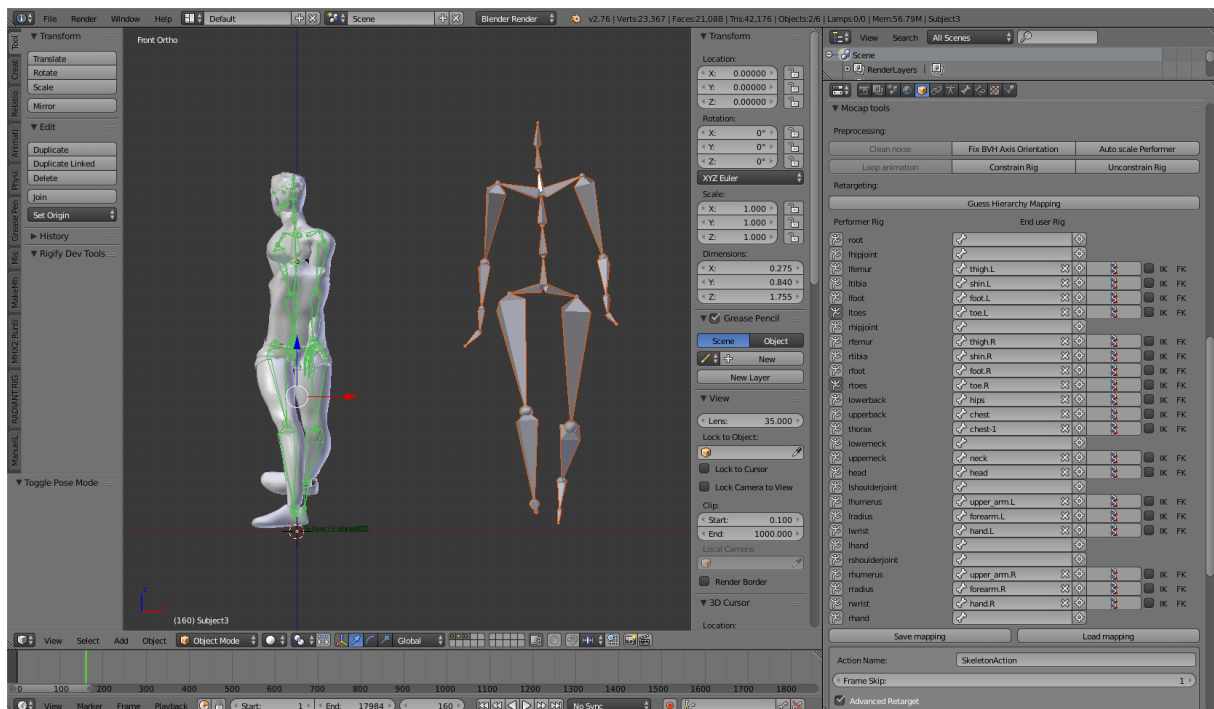


Figure 2: Retargeting using the Mocap tools add-on. Initial attempt fails because of inconsistencies between the two skeleton definitions.

However, because of inconsistencies between the two skeleton definitions, we found that the motion was improperly transferred onto the target rig. If we take a deeper look into the two skeletons, we are able to see why this happens.

Figure 3(a) shows the 3D orientation of the right thigh bone of the 3D character’s rig, in rest position (note that this is presented in “Edit mode”). When compared with the 3D orientation of the thigh bone of the skeleton in the ACCLAIM motion file (Figure 3(b)), it is readily apparent that, given a right-handed coordinate system, there is an offset of  $-90^\circ$  (around the local Y-axis) between the bones of the two skeletons.

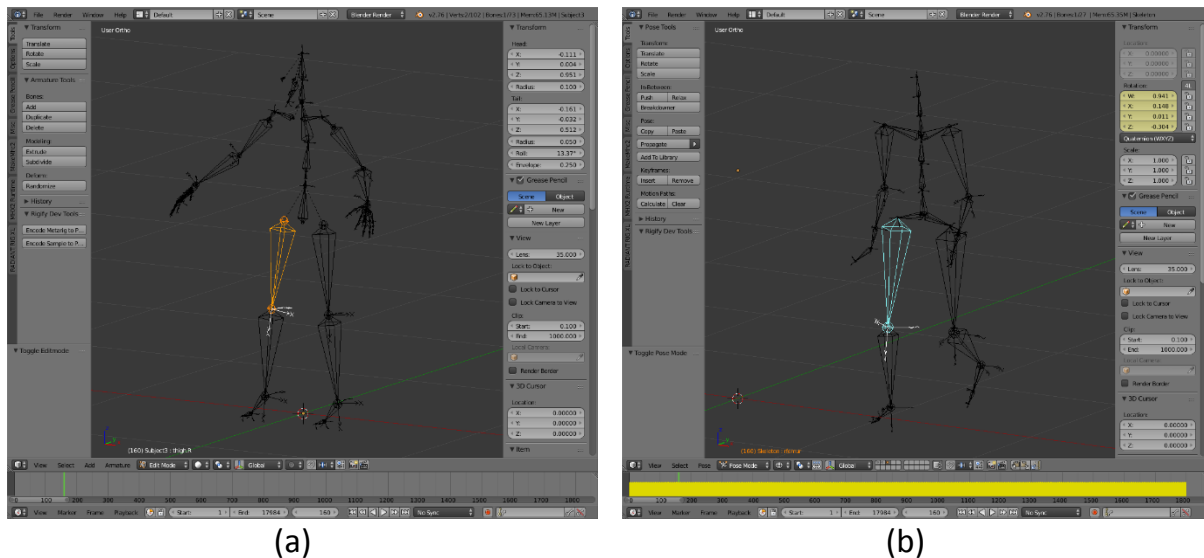


Figure 3: The 3D orientation of the right thigh bone in (a) the 3D character’s armature, in rest position (Edit mode) and (b) the ACCLAIM motion file’s skeleton, in pose position (Pose mode).

The way to tackle this is to compensate for the inconsistencies between the two skeleton orientation definitions by adding roll angle offsets to every bone that shows such inconsistencies. This can be done by editing the value of the field “Roll” in the “Transform” panel (see Figure 3(a) top right part of image).

Motion files, such as the ACCLAIM file format used in this study, describe the orientations of the joints as 3D Euler angles (here the order is XYZ), relative to the skeleton’s rest position. Therefore, the 3D orientations of the appropriate bones in the rest position of the 3D character need to be modified. In Blender, this is done in “Edit mode”. However, because the 3D mesh of the character is already associated to its armature, we first need to dissociate the two and then parent<sup>2</sup> them again, after the changes take place.

Figure 4 shows how to parent the 3D mesh of the character with its modified armature. One has to first select the 3D mesh object and then (holding down the Shift button) select its modified armature. With the two objects selected, by pressing the key combination CTRL+P, the “Set Parent To” drop down menu appears, in which we choose “With Automatic Weights”. The two objects are now parented. This means that any motion retargeted on the modified armature, will make the 3D character move by appropriately deforming the mesh object.

<sup>2</sup> Parenting is the procedure in which an object (here the 3D mesh of the character) is associated to another object (here the skeleton of the motion file).

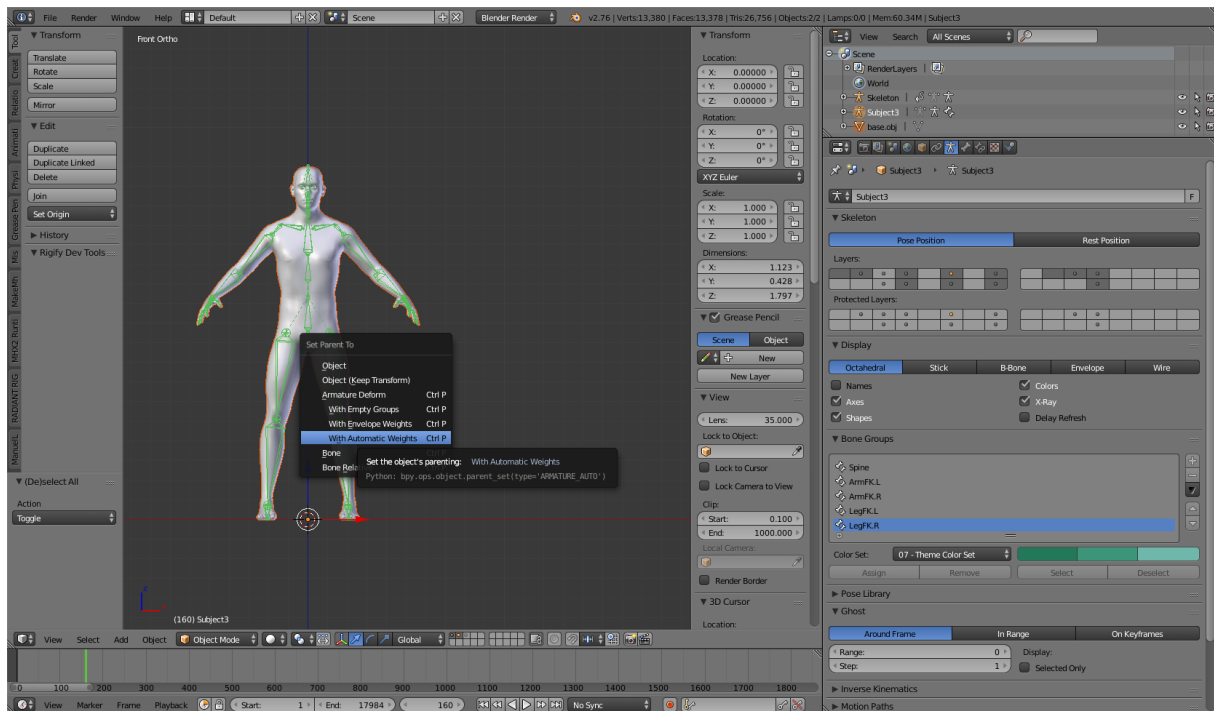


Figure 4: Parenting the 3D character's mesh object to the armature object using automatic weights.

Figure 5 shows the same example with the right thigh bone, seen in Figure 3, after its 3D orientation has been modified. It is readily apparent that by adding a roll angle offset of  $+90^\circ$  to the thigh bone in the 3D character's armature, the right thigh bone has the same 3D orientation across the two skeletons.

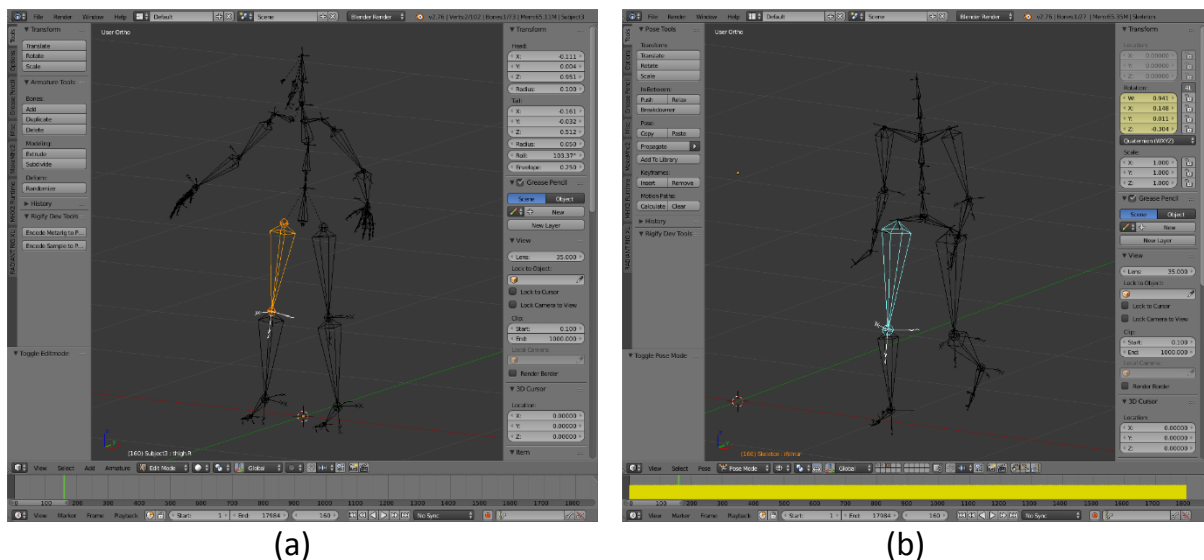


Figure 5: The 3D orientation of the right thigh bone in (a) the 3D character's **modified** armature, in rest position (Edit mode) and (b) the ACCLAIM motion file's skeleton, in pose position (Pose mode) - same as in Figure 3(b), for easy comparison.

Once the above procedure is applied to all the bones that show inconsistencies between the definitions of the two skeletons, the 3D character's armature can now be retargeted with the motion from the ACCLAIM motion file. Figure 6 shows the 3D character being deformed (properly now) according to the motion capture data.

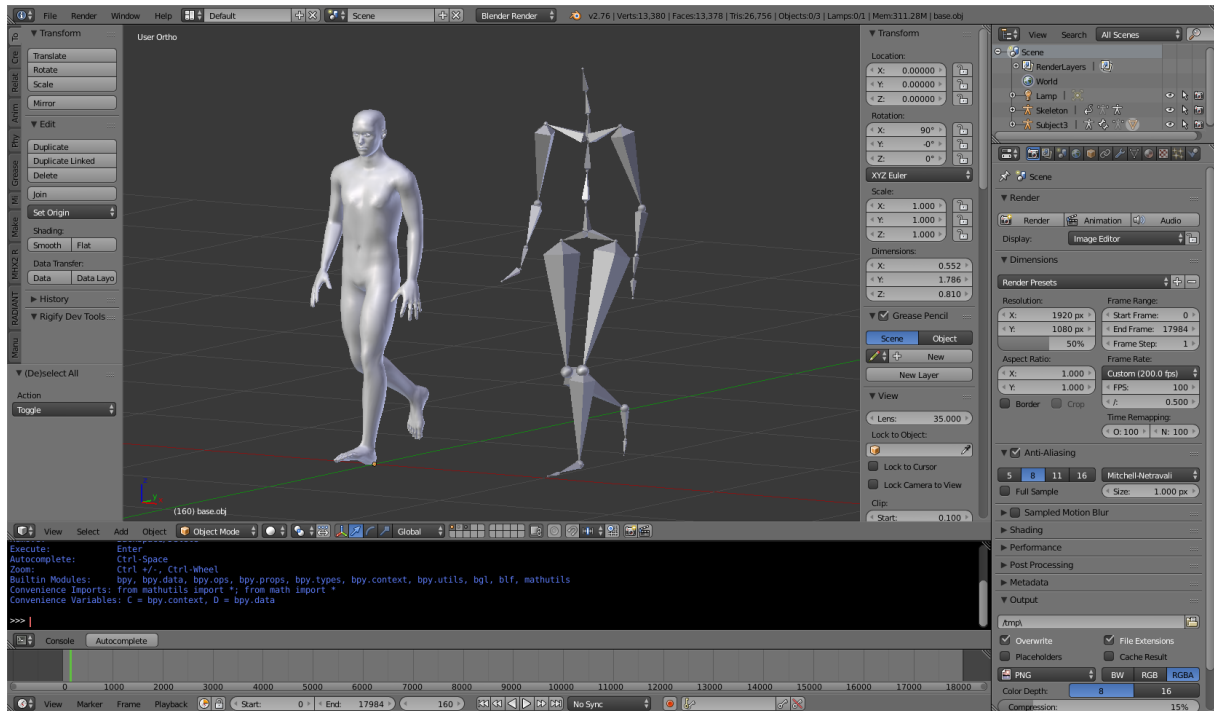


Figure 6: The 3D character properly deformed, after its armature is modified to deal with all the possible orientation inconsistencies that may appear.