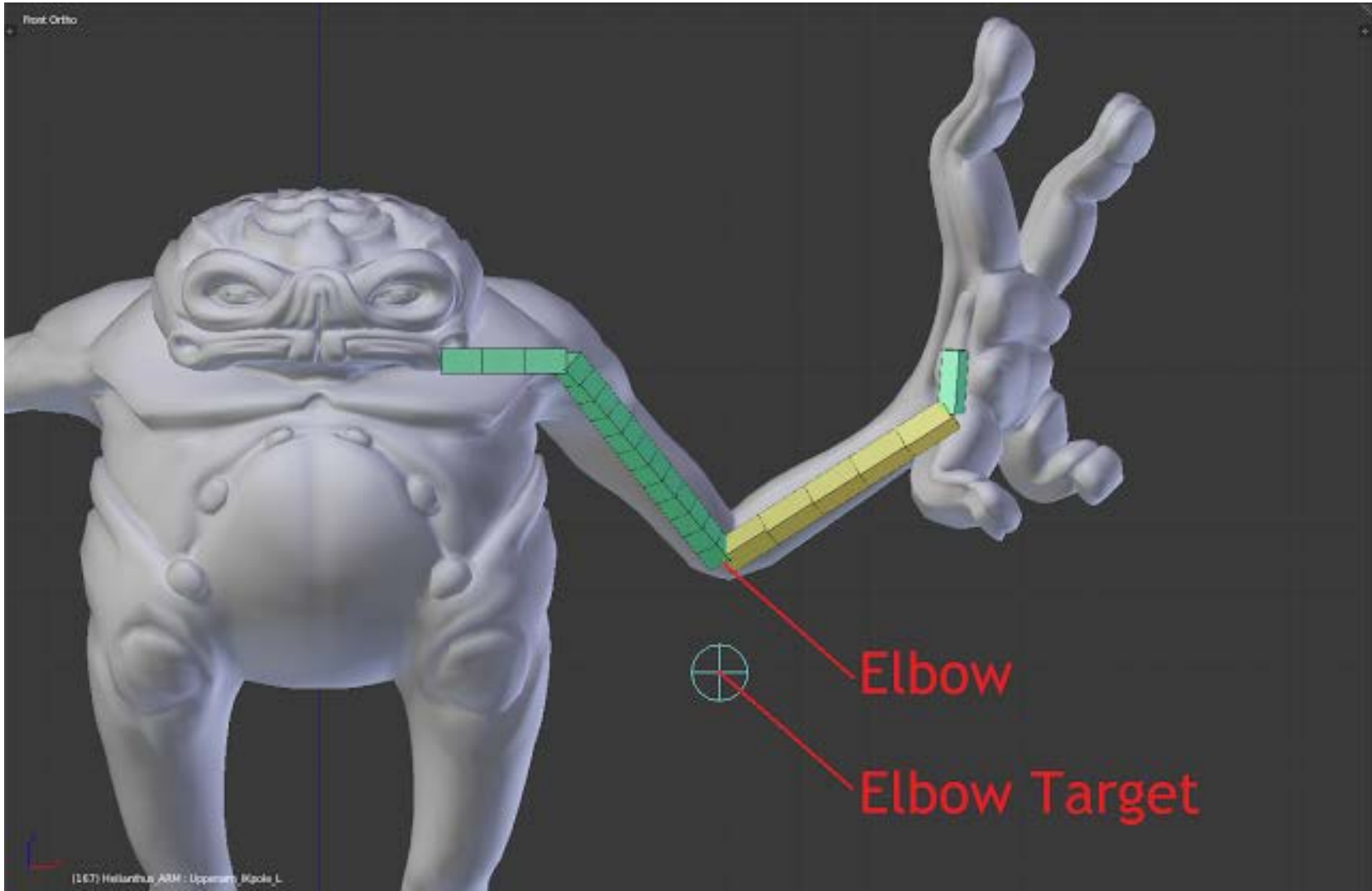
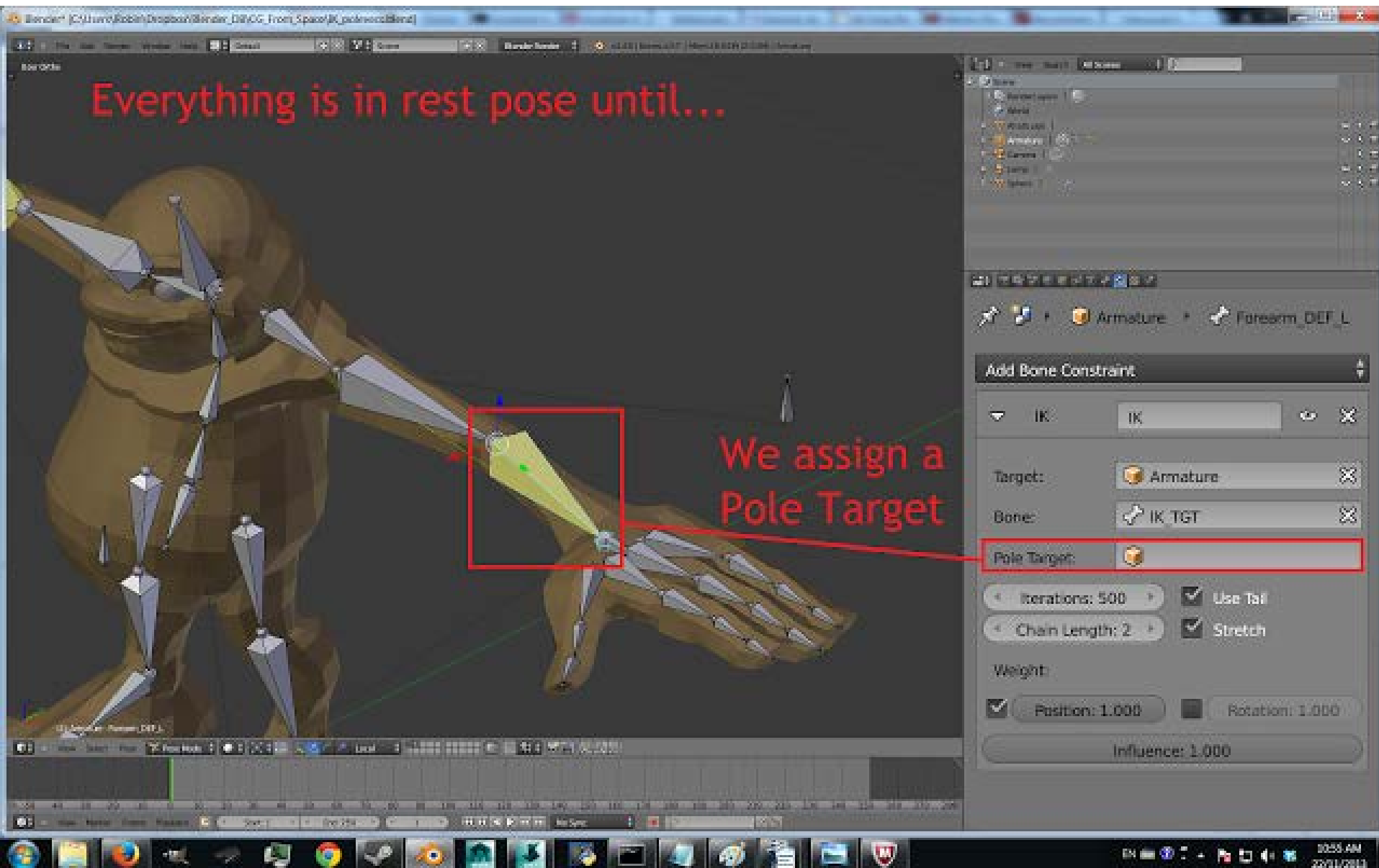


Blender Quick Tip: How to set up IK pole vectors without breaking the bind pose

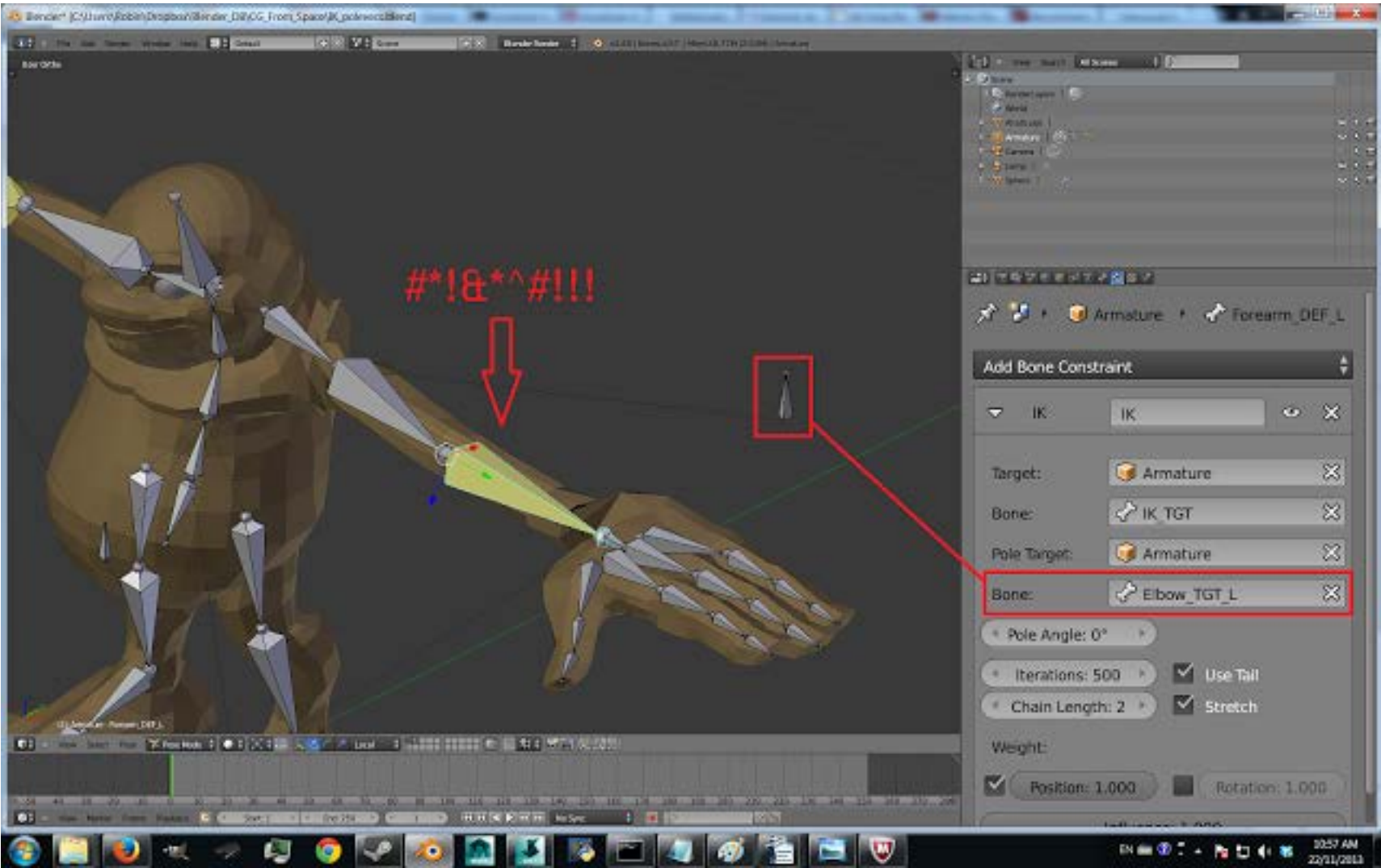


Blender's IK system let's you assign targets (called Pole Vectors) which lets you control the orientation of the bend in the IK chain. But there's a common problem...

Say you're rigging an IK arm.
You need to be able to control the orientation of the elbows, and you do this using IK pole vectors.



But there's a catch, isn't there? When you add the pole vector, your binding pose breaks! That's bad. When you are rigging, Thou shalt honour the bind pose and keep it holy. Preserve it, pure, sanctified and unscathed.



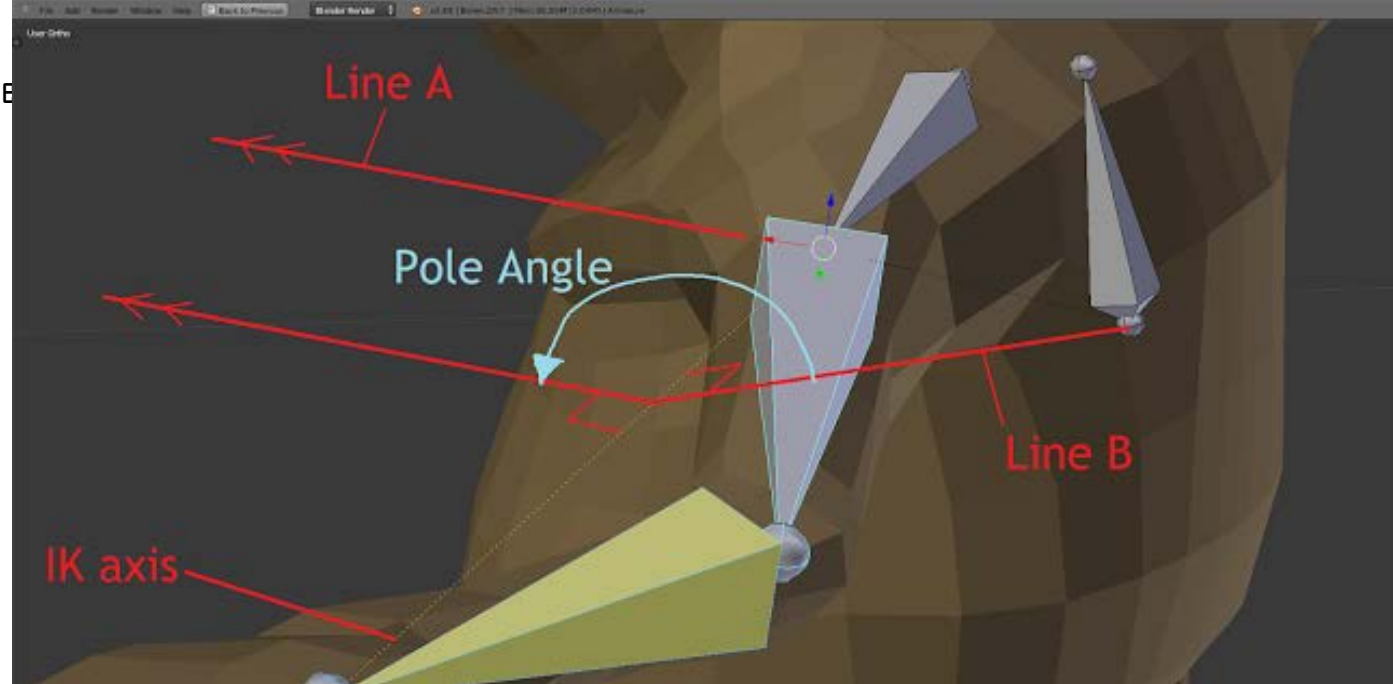
How do we fix this?

Notice that Blender's IK constraint requires that you specify a Pole Angle. By varying this value, you can rotate the elbow around the arm's axis. What value should you put in there? Well, that's the problem. Yes, you can vary it by hand until the elbow is roughly in the binding pose, but if you do it this way, you won't get it exactly right.

To fix this problem, you need to understand exactly what the pole angle is. Check out my diagram.

Line A is the x-axis of the upper arm bone. Line B is drawn from the Pole Target and intersects the IK axis at 90° . The Pole Angle is the angle the IK system will put between lines A and B.

In other words, if the Pole Target is located exactly on the upper arm's positive x-axis, setting the Pole Angle to 0° will put the IK arm into the rest pose.



This is how you can set it up:

- In pose mode, make sure your entire rig is in its rest pose (select everything, hit alt-S, then alt-R, then alt-G)
- In edit mode, duplicate the upper arm bone. This will become your elbow target.
- Keeping the duplicate bone selected, go into pose mode. Move the bone a distance of your choosing along its positive local x axis. Don't get this wrong - it has to be the the local x-axis in the positive direction.
- Go to menu Pose -> Apply -> Apply Pose as Rest Pose
- Add the IK constraint to the forearm bone as normal. Use the new bone as a pole vector. Set the Pole Angle to 0° .

Tada! You can flip between pose mode and edit mode, and the arm won't twitch.

Some other notes:

- You don't have to use a pole angle of 0° . If the Pole Target is on the negative local x-axis, set the Pole Angle to 180° . Put the Pole Target on the positive local z-axis and set the Pole Angle to 90° - whatever it is you need to do.
- What if you have 3 or more bones in the IK chain? Well, Blender uses the x-axis of the basal-most bone when using the Pole Angle. In a 2 bone arm setup, the upper arm is the basal-most bone.
- I'm talking about the local pose x-axis. The x-axis that corresponds to the bone itself. When you try and use the local x-axis in edit mode, Blender thinks you're talking about the local x-axis of the armature, and that's not what you want.
- It *is* however possible to set up custom transform orientations in edit mode. When you're in edit mode, select a bone, look at the bottom of the hotkey "n" panel. There's some "transform orientations" tools which let you use the local axes of the selected bone. Blender doesn't update the local axes if you move the bone though, so if you move it, you need to reload the local axes- that can be kinda annoying.
- The pole angle picture is all wrong. The pole angle is actually angle between line_C and upper_arms x_axis, where line_C is projection of line_B onto XZ plane of upper_arm's bone by vector ik_axis. Also line_A is not an upper_arms x_axis and you don't need it.
- Calculating the IK pole target's pose from an FK pose is tricky because there's little to no docs on how the IK chains work. But! You can do this:
 - * Set up an IK chain
 - * Give it a pole bone (do it so that it doesn't pop).
 - * Duplicate the pole bone
 - * Parent the new bone to the upper FK arm bone.

This new bone serves as a target for the pole bone when you switch IK -> FK. I've yet to try it in Blender (I've tried it in other software packages) but it definitely works!