

Assignment 2:

Hierarchical Modeling: Modeling and Animating Quadruped Animals

Due Date: November 29, Friday, 2019

Demo will be in class B-Z05 during the class hour (09:00 - 10:30)

Requirements

You are supposed to implement a hierarchical model for modeling and animating a quadruped animal (could be a dog, horse, donkey, cow, ship or your favourite). You will use the matrix stack data type and the WebGL procedures that are used to manipulate the matrix stack that we have discussed in class. You can use the schematic humanoid code in Edward Angel's source codes associated with the textbook as the starting code (EdwardAngel\7th_Edition\CODE\09\figure.html and EdwardAngel\7th_Edition\CODE\09\figure.js). It just generates static humanoid postures. However, in this code, the articulated humanoid figure is hard coded and the hierarchy cannot be changed. You must use a generalized tree structure as well as a generalized traversal strategy as we discussed in the lectures.

I want your articulated body of your quadruped animal to be more realistic than the one provided there (not just rectangles as body parts; more realistic body parts such as ellipsoids, cylinders, and so on.). The speed and quality of the animations, as well as the realism of the quadruped animal will affect your grading.

Essentially, you will write a sequence of function calls with appropriate transformations and transformation parameters and when you execute that segment, it should produce a hierarchical model of the 3D quadruped animal. When we change the model parameters, it should behave accordingly. You will also animate your models by changing the model parameters gradually. Your primitives will be 3D rectangular prisms, cylinders, spheres, and ellipsoids (or even polygonal models) for appropriate parts.

Build a user interface to display and animate your model. Also define some buttons/sliders for giving transformation parameters for joint angles, etc. In other words, I should easily be able to give a posture to your quadruped animal using your interface. You should be able to give the posture to the animal model and save these postures as keyframes. Then, you will animate the figure by interpolating the joint angles and other transformation parameters between the keyframes.

You should demonstrate a sample animation of some behavior, such as walking, running, or a more complex behavior (like the animal jumping over a hurdle). Please prepare this animation in advance before coming to the demonstration. This will have a significant effect on grading.

You should be able to save your animations and load and play the them back. You can save the animations in the form of text files containing the keyframe numbers and joint parameters (both translation and rotation parameters) for each frame. Intermediate frames between the keyframes must be filled using a simple interpolation techniques, such as linear interpolation.)

You must use a projection transformation for 3D to 2D conversion (e.g., a perspective projection transformation such as `glPerspective`, as described in `MV.js`).

Grading will be based on the essential functionality, and speed and quality of the animations, as well as the realism of the quadruped animal.

Optional: After you implement the basic functionality, you can consider the following optional parts:

- A direct manipulation interface using mouse to give the animal model the desired posture. You can do this by directly moving the parts of the model using mouse (direct manipulation).
- Applying texture mapping (you can use WebGL's facilities for this purpose) to increase realism.
- Add or delete limbs to your articulated body to change the articulated structure using an interface and animate that modified articulated quadruped animal.

If you do not implement all the basic functionality, optional part will not be considered for grading.

Look at the documentation about WebGL for implementation details of data types and the functions that are used to manipulate hierarchies.