CPSC 314 Assignment 3

due: Fri Oct 13, 2023; worth 7%

1. (23 points) Animating with Hierarchical Transformations

   Your task will be to create and animate an underwater scene with one-or-more sea creatures, e.g., sea turtle, seal, octopus, sea turtle, star fish, and more. The learning goals are to be able to create and use transformation hierarchies, creating basic character animations using these, and using additional `three.js` functionality, including keystroke-driven interactions.

   Be creative and have some fun! In the end, your personal satisfaction with what you've done and learned is arguably more important than anything else. We'll show some of the most creative results in class, with some prizes.

   Important Rules:
   As illustrated in the template code, you should use the basic matrix support functions provided by `Three.js`. This includes: `Matrix4.multiply(), Matrix4.copy(), Matrix4.makeTranslation(), Matrix4.makeRotationX(), Matrix4.makeRotationY(), Matrix4.makeRotationZ(), Matrix4.makeScale()`.

   `Three.js` provides tools to help create and use hierarchies via the specification of a `parent` frame, for objects, i.e., via `parentObject.add(childObject)`. However, for this assignment, all animated objects will have the world coordinate frame as their parent, as implemented by `scene.add(object)`. You will be directly building their 4×4 object-to-world modeling transformation associated with each object, i.e., object.matrix. Look at the example hand animation in the template code, which was also discussed in class.

   Requirements:

   (a) (5 points) **Basic experimentation:** First, be able to demonstrate an understanding of the fundamentals of the code to your TA. There are no specific deliverables, aside from leaving the basic changes described below in the submitted code.
   (i) Ensure that the scene loads and plays the basic animation. You will need a local http-server or another way to enable local file access, as you did for A1.
   (ii) Understand how objects are loaded form OBJ files e.g., see `initFileObjects()` and are then instanced, i.e., see `onResourcesLoaded()`. Add an extra teapot to your scene at a desired location and in a desired orientation. Test your scene.
   (iii) Understand how the `myboxMotion` and `handMotion` keyframes influence the motion. This can be found in `initMotions()`. Make `mybox` travel higher during its animation cycle. Animate the orientation of link 1 of the hand. Also, when the right finger bends, increase the amount amount by which it bends. (iv) Understand how `handSetMatrices()` makes use of animation variables, `avars`, to build the relevant transformation matrices. Make the palm wider (in the Z-dimension) by making changes to `handSetMatrices()`. Also make the palm longer (in the X-dimension). For this you will need to make adjustments to the finger transformations as well.

   i) must keep the basic animation and explain it.

   ii) try to understand mesh properly

   iv) I would try to understand more on how the matrices work in transformations

   (b) (5 points) **Your underwater scene:** We recommend using separate files and code for your underwater scene and sea creature(s). Follow the steps below.

   First, find an image of a sea creature in order to help you build an multi-articulated model that is approximately anatomically correct in terms of the joint locations and

limb lengths. In your README.txt file, document the URL of the image that you have used as an approximate reference. Take liberties as needed, i.e., introducing joints to approximate the flexible arms of an octopus is fine.

Second, create a crude sketch on paper (or tablet) that illustrates the labelled links and their coordinate frames, and that provides approximate dimensions of each link. The goal here is to encourage you to be methodical in the design of your creature and to have a specific plan in mind. Links can be made using any of the basic primitives, including boxes, spheres, capsules (cylinders capped with spheres), etc. At a minimum, build one sea creature that has at least 10 links. This might include a multi-link body, a neck, head, tail, and multi-link legs or fins. Use what is needed; a starfish, orca, and octopus will necessarily all have very different constructions.

Take an image of your crude sketch, named `sketch.jpg`, and include it in your code directory, to be submitted with your final code. Use whatever image file format you find to be convenient, but try to keep the size to less than 1 Mb.

Third, build your creature in the assignment code. Begin by using fixed rotations for the joint angles. Each link can be built using existing threejs primitives, i.e., box, capsule, sphere, cylinder, plane, cone, tube, etc. Custom geometry is fine as well, although you should document how it was created. There is no need to add visible coordinate frames for each link, as was done for the hand, although it may be useful for debugging.

(c) (3 points) Create an animated motion for your sea creature, which should repeat in an animated loop.

(d) (2 points) Create at least one more motion, and add keystrokes commands that can switch the creature between different motions.

(e) (8 points) Creative component: enhance your scene via improvements to the scene, the motions, and the way in which keystrokes can be used to interact with it. Evaluation of the creative component will necessarily be subjective. We will value overall novelty, aesthetic appeal, interaction, scene complexity, and motion complexity.

To hand in:

- In the README.txt file, include your name, student number, and any information you wish to pass along to the TAs. List any external resources (web pages, books, consultation with others) that you used. This assignment is to be completed individually. Comment on your implemented features and keybindings.
- Submit your assignment by the deadline using web handin:
  `https://www.students.cs.ubc.ca/cgi-bin/protected/handin`
  or: `handin cs-314 a3` on the linux machines in ICCS x005
- Demonstrate your code to a TA in the lab, or at other times that may be announced.

We look forward to seeing your sea creature(s) and their world!