

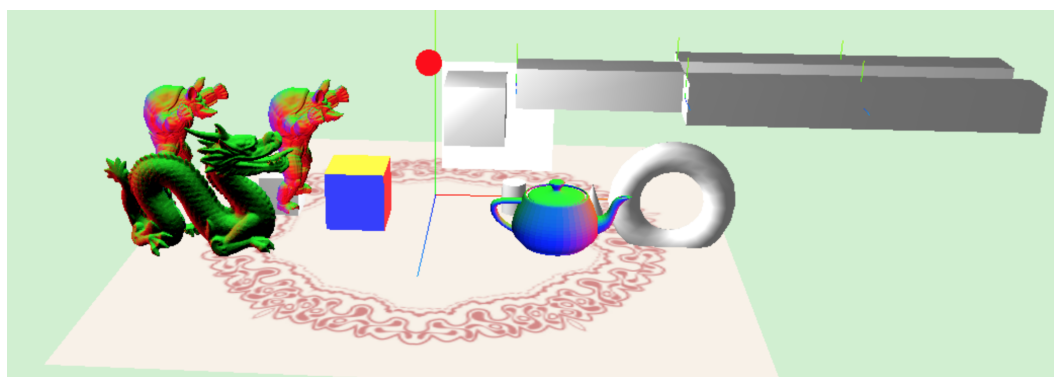
CPSC 314 Assignment 3

Due 23:59:59 on Mon October 7, 2019

1. (23 points) Animating with Transformations

Your task will be to create and animate a scene with flying dragons. 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 as creative and ambitious as you like. The best work will be posted on the course web site in the Hall of Fame and will be shown in class.



Important Rules:

As illustrated in the template code, you are free to 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(object)`. However, for this assignment, all objects will have the world coordinate frame as their parent, as implemented by `scene.add(object)`. Instead, you will be directly building the 4×4 modeling transformation associated with each object, i.e., `obj.matrix`, where 'obj' is the object in question. This matrix should take points to the world coordinate frame, known as the `scene` object in `three.js`. Look at the example hand animation in the template code, and as shown in class.

Requirements:

- (a) (5 points) First, be able to demonstrate an understanding of the fundamentals of the code to your TA. There are no specific deliverables for this aspect, aside from leaving these changes (commented out if you wish) in the final submitted code.
 - (i) Ensure that the scene loads and plays the basic animation. Remember to enable local file access.
 - (ii) Understand how objects are loaded from OBJ files e.g., see `initFileObjects()` and are then instantiated, i.e., see `onResourcesLoaded()`. Add an extra dragon 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). You may need to make adjustments to the finger transformations as well.
- (b) (5 points) Develop a flying dragon character that you add to your scene. It should have two wings, two arms, two legs, a body, and a head, and consist of at least 12 links in total. At this point, you could use fixed rotations for the joint angles. You can build it from threejs primitives of your own choosing. Custom geometry is fine as well, although document how it was created. You do not need to add visible coordinate frames for each link, as was done for the hand. Creating a sketch on paper will be helpful for keeping track of the dimensions and joint locations. Be proud of your creation :-)
- (c) (3 points) Create a dance cycle for your dragon. It can be in the air or on the ground. Do animate the overall position of your alien.
- (d) (2 points) Create at least one more dance style, and add keystrokes commands that can switch the dragon between different dance cycles.
- (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, but we will value overall novelty, aesthetic appeal, interaction, scene complexity, and motion complexity.

To hand in:

- Hand in a `README.txt` file. This should include your name, student number, and user name, as well as any information you wish to pass along to the marker. 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 in your `README` file.
- To hand in, create a `cs314` subdirectory under your home directory, and create an `a3` subdirectory underneath that. Copy all the files that you wish to hand in to this directory. The assignment can be handed in using `handin cs314 a3`.
- Demonstrate your code to a TA in the lab, or at other times that may be announced.

We look forward to showing off the creative results from this assignment in class!