

Interactive Graphics

Homework 2 Report

Ionut Marian Motoi

E-mail: motoi.1892355@studenti.uniroma1.it

Matricola: 1892355

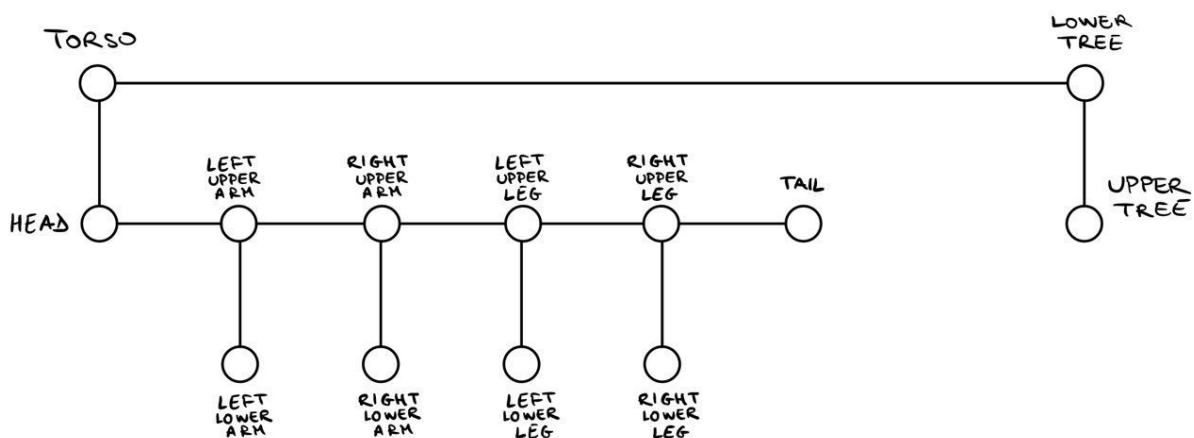
Sapienza Università di Roma

Master course in Artificial Intelligence and Robotics

Homework

The aim of the homework was to create a hierarchical model of a simplified grizzly bear and of a tree. Next different textures had to be added for the bear's body and head and for the tree. Finally the bear had to be animated after pressing a button, making it walk towards the tree, get on his hind legs and start scratching its back against the tree.

Hierarchical model



Both the models of the grizzly bear and of the tree were made using a hierarchical model.

This was implemented through a tree node data structure with a traversal rule of depth first, so first going to the child (if any) and only after to the sibling (if any).

Each node represents a cube in the application and stores a pointer to a sibling (or null), one to a child (or null), one to a function that draws the object (after properly transforming it) and

a homogeneous coordinate matrix to multiply on the right of the current modelview matrix and that represents all the changes made going from the parent node to the current node. The first nodes of the two models, namely the torso of the bear and the trunk of the tree were created as siblings. This choice allowed for a simpler implementation and at the same time it was easier to apply a transformation to both the models, i.e. the rotation of the scene by -45 degrees around y and by -15 degrees around x and z. Using an hierarchical model allows to more easily create complex models composed of different parts. An advantage is that it permits to move and rotate a part of a model with respect to the parent node, thus allowing for a more intuitive modelling and animation.

Texture

A total of four textures were applied: one for the bear's body, one for the face, one for the trunk of the tree and one for the top part of the tree. To apply the textures, only one active texture was used and the appropriate texture was bound when needed during the traversal of the hierarchical model. For example, to draw the face of the bear the correspondent texture was bound only to draw one side of the cube representing the head and then the texture of the body was bound again to draw the rest of the bear.

Animation

At every iteration of the render function the delta time is computed (needed to compute speeds that are independent of the framerate) and it is passed to an animate function. The sequence of actions needed for the animation was managed through a switch case based on a global variable (key). Every key represents a type of action the bear has to do. After reaching certain requirements (e.g. the bear is near the tree) the key increments its value telling the animate function to use the next series of actions. For the more complex actions a separate function was created to maintain readability of the code.

More precisely the key starts with value 0 in which case the bear does not move. When the button is pressed the key value increases to 1 and starts walking by modifying the values of the angles of the limbs between certain limits. When it gets near the tree it starts rotating positioning itself facing in the opposite direction of the tree. Then it gets closer to the tree backwards, stretches its legs and gets on his hind legs. Finally it starts scratching its back against the tree by sliding up and down while the legs move accordingly.

The approach I used with the switch case has the advantage of making the various phases of the animation easier to implement independently. The downside could be that the key number of each case must be modified if there is the necessity to add an action in the middle of the animation. So if it is required to create a longer sequence of actions, a “string” key instead of an integer one would make it easier to modify the order and the number of cases without having to rename each subsequent case.

Preview



Start Animation