Interactive Graphics

Point (1).

I Created a hierarchical model of a (simplified) Horse, And All components are cubes, So The torso is the root node of the figure, To represent a complex figure, like a humanoid or a Horse, it is possible to figure it with a hierarchical tree of rectangles as shown in the following figure, I have taken from Chapter 9 of book Interactive Computer Graphics: A Top-Down Approach with WebGL by Edward Angel and Dave Shreiner.

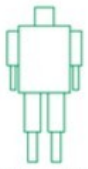

Figure 1: A humanoid figure

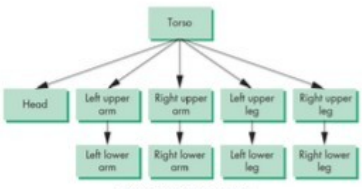

Figure 2: Tree representation

I created to the horse's one tail which has two parts I introduce variables and increase the number of nodes at 15. Theta must have same dimension of nodes number. Modifying width and height variables it is possible to make proper horse's dimension. We have to add the tail to initiNodes function: a case of the switch can represent the tail and another appropriate function tail() can place the tail in the horse.

Finally, with Theta angles it is possible to rotate the rectangles to make them look like a horse and with translate in initNodes function it is possible to translate rectangles to attach them to the body.

A-In Html File:
I only have removed the sliders.

B- In Js File:
I have Removed document.getElementById for the components.

Point (2).

I Added a procedural texture to the body(only on the torso) of the horse. The texture should be a checkerboard pattern but with a linear decrease of intensity from the front to the back of the body. I Use, as a reference, textureCube4 of Chapter 7 of the examples of the textbook. And I applied a linear decrease of the intensity on the torso with the texture.

A-In Html File:
Added many features which can be used for the model such In the vertex-shader script, I have to add varying vec4 fColor and varying vec2 fTextCoord and, in the main, fTextCoord = vTextCoord, so it is possible to program the next part in fragment-shader, where I have to add a variable varying vec2 fTexCoord, two uniform sampler2D (Text0 and Text1) and, in the main, I have to replace (or multiply) the fragment color vector with two texture2D functions that takes two parameters: sempler2D Text0/Text1 and varying vec2 fTexCoord.

B-In Js File:

I have a procedural texture applied to the torso of the horse and exclude the others, I have to add variables to manage texture coordinates and texture images and I used the colorCube(color) function with configureTexture(); into torso(); and for exclude the other parts either I can use colorCube(color) function inside the components functions Or I could use another method which is delete.Texture(textures) this method to excludes the other parts from the texture, now has a linear decrease. So I wrote the code of configureTexture function to configure as a texture the just added texture images variables. and finally I pass to fragment shader which means that just created the texture on the torso and declined the others.

Point (3).

I Added a simple model for jumping obstacle for horse and the obstacle has consists of three bars (2 vertical,1 horizontal).so I just created 3 nodes for the obstacle. Then created 3 cases for those 3 nodes in switch. After I created those nodes functions which is identical to other torso nodes functions. And call traverse() function in render().

Point (4).

I Added a button that starts an animation of the horse(walk and jump above the obstacle) so that, starting from an initial position where it is standing and positioned along the x axis, it walks to the right by moving (alternatively back and forth) the legs and according the jump which has two steps (first-we do the linear interpolation and bending the legs, second-to jump and must be avoid the obstacle)

A-In Html File:

I only have added an button to animate.

B-In Js File:
I added a document.getElementById and a variable both linked to the Html button to start the animation on the button click.

```
document.getElementById("startAnimation").onclick = function(event) { animate = !
animate;
```

So when it click again on the button all the variables come to a default value, making the horse return to the initial position. Now it is needed to move the initNodes function call inside the render function. Finally, inside the animation function, and if else-if the model will does jumping by its own steps. And for legs animation it taks this formula *(resulting values: b = 0, a = - (LengthUpperLeg - LengthUpperLeg*cos(180-140) + LengthLowerLeg - LengthLowerLeg*cos(60)) / 21).*

Basically when animate button is clicked it is needed to restart all the variables to reset the initial situation. And also before jumping the horse should works on bending his legs in order to execute the jump.

Briefly conclusion:

I-noticed that problem regards the use of the resources in our model because at each render(); function all the objects transform a cube into their cubes, so there are some matrix multiplication done for each render(); function inside the application to build the model view matrix. and this is done for each object. All these operations involve a high use of the CPU and could be optimized by using a single buffer for all the objects having the same color and it will be in one resulte.

II-The problem that mentioed above could be solved by using all the transformation matrices inside the vertex shader and multiplying them in other words(to use the writing by transform and cmpute them)there with the vertex,so the render(); function of each object it will returns a number which specifies which matrix we have to use .