Summer 2020
CS 300 | Advanced Computer Graphics I
Assignment 2 | Normal Mapping

Description

In this assignment, the main task to is to implement normal mapping on top of the Phong illumination model of assignment 1.

For this programming assignment you need to the following:

- **Tangent Space**
  For each one of the shapes generated in previous assignments (plane, cube, cylinder, cone and sphere):
  - For each triangle, generate the tangent and bitangent vectors using the vertex positions and texture coordinates. Incorporate lab 3 to your framework.
  - Update the tangent T to make sure that it is orthonormal to the normal to the geometry by implementing the Gram-Schmidt orthonormalization process.
  - Transfer to the GPU as per-vertex data the normal, tangent and bitangent. Again, remember that these two vectors are in model coordinates.

- **Shader programs**

  - **Vertex Shader**

    1. Transform N from model to camera space by multiplying it by the normal matrix.
    2. Transform T and B from model to camera space by multiplying it first by the model matrix, then by the view matrix. Note that, unlike the normal vector, the tangent and bitangent vectors are parallel to the plane tangent to the surface of the object, thus we transform them using the same affine transformations (model and view matrices) that we apply to vertex positions.
    3. Output those vectors to the fragment shader.

  - **Fragment Shader**

    1. Given the fragment's texture coordinates, read the normal from the normal map texture and transform it to the correct range of values.
    2. Build the matrix to represent the transformation from tangent space to camera space.
    3. Use the modified normal, the light and view vectors to implement the Phong Reflection Model (note that all three vectors should be in camera space at this point).

- **Scene Setup**
  - Render a 50x50 units 2D plane aligned with the XZ plane and 15 units under the origin, i.e. the center is at (0,−15,0) and normal is (0,1,0).

- Render one shape with size 10x10 at a time (from the shape library), arrows keys will rotate that shape.
- Place the light(s) equally spaced along a circle centered at (0,0,0) with radius of 20 and have them rotate slowly around the Y axis. While rotating they should move up/down in a sinusoidal manner along the Y axis.
- If light type is spotlight or directional, have them look at the object center, i.e. the light(s) follows the object center.
- Apply the shader to both the object and the plane.
- Render a small sphere (radius = 1) to represent where the light(s) is (do not apply the shader on these spheres, they should be always white).
- Load the normal map from file and apply it to every object on the scene.

- **Input Handling**
  - Move the camera around in always looking at the object.
    - W: Move up.
    - S: Move down.
    - A: Move left.
    - D: Move right.
    - E: Further from object.
    - Q: Closer to object.

  - Select shape to be rendered through the number keys.
    - Numbers 1 to 5: Change the shape to be rendered
      - 1: Plane
      - 2: Cube
      - 3: Cone
      - 4: Cylinder
      - 5: Sphere
    - +: Increase the shape subdivisions
    - -: Decrease the shape subdivisions

  - Select the light types:
    - 7: All lights become point lights.
    - 8: All lights become spot lights.
    - 9: All lights become directional lights.

  - P: Toggle to pause/start the light animation.
  - N: Toggle normal rendering
  - F: Toggle face/averaged normal
  - M: Toggle wireframe mode on/off

  - T: Change between rendering modes:
    - Normal Mapping: Scene with lighting and normal mapping with texture
    - Normals: Geometry normal in camera space as color
    - Tangent: Geometry tangent in camera space as color
    - Bitangent: Geometry bitangent in camera space as color

- Object rotation for center shape.
  - Arrows Up/Down: Rotate the shape along Y-axis
  - Arrows Right/Left: Rotate the shape along X-axis

Assignment Submission

Please refer to the syllabus for assignment submission guideline. Failure to the submission guidelines correctly might cause you to lose point.

Grading Rubrics

The following is a rough guideline on how your assignment will be graded and the weight of each part.

- Tangent and Bitangent calculation 30%
- Correct tangent/bitangent transformation and usage 30%
- Normal map usage 20%
- Rendering modes and presentation 20%