Assignment 3 : Direct Lighting and Texturing

YONGJIAWEI 2019533095 YONGJW@SHANGHAITECH.EDU.CN

1 INTRODUCTION

In the last assignment,we need to create our first B-spline curve with an iterative evaluation algorithm and render the surface mesh.

In this assignment, we are required to implement the basic ray-tracing with direct illumination using Phong lighting model as well as texturing algorithms.

2 IMPLEMENTATION DETAILS

2.1 Implement a pin-hole camera

In this part, I implement two function, camera::lookat and camera::generateRay, the function lookat is to calculate axes of camera since the z_axis by position and look_at,then x_axis by z_axis ans ref_up,finally y_axis by x_axis and z_axis. the function generateRay take the coordinate in a NDC space ,then change it to the world space, then generate a ray.The concrete transform is to multiply a transform matrix of the origin coordinate, this matrix is in fact a change basis matrix.

2.2 Implement algorithms for ray-objet intersection

In this part I implement one function Sphere::rayIntersection, this function is to calculate if a ray is passing a sphere object,if so, calculate the relative information of the intersection point, including position, normal,uv coodinate and so on.

2.3 Implement a Phong lighting integrator for are lights with uniform samples

Phong lighting modle is in fact consists of there parts, ambient,specular and diffusion,ambient is the basic color of an object even if ther is no direct light, specular can be calculate as below

$$\text{Specular} \approx (V \cdot R)^n.$$

Where V is the view direction and R is in fact:

$$R = 2(N \cdot L)N - L.$$

and L is the incident light direction at P. Specular is actually the cosine of normal and view direction, I calculte these three and add them up, mulitiply the object color and light color, the final color is calculated.

2.4 Implement texture

In this part I implement texture , I apply a sample texture algorithm, map the coordinate in the sphere to a square checkboard. x and y coordinates is mapping to u coordinate by the angle of x and y, z coordinate is mapping to v coordinate samilarly.

2.5 Implement anti-aliasing for ray-tracing

In this part I implement anti-alising for ray-tracing, when calculating each pixel in the screen , some edge of the modle might be wrongly assigned a color, so what I need to do is to generate muliple rays form one single screen pixel , the actual color of this point is the average of these colors of each ray.

3 RESULTS

Fig 1 and Fig 2 show that the correction of lighting.
Fig 1 and Fig 3 show that the correction of textureing.
Fig 1 and Fig 4 show that the correction of sampleing.
Fig 1 and Fig 5 show that the correction of ant_alising.

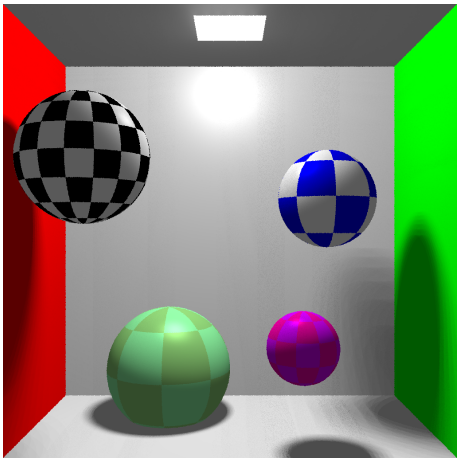

Fig. 1. normal case.

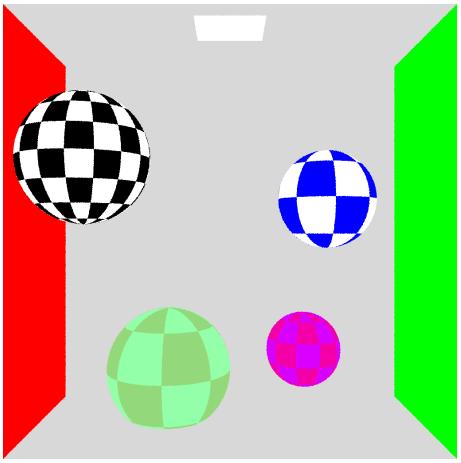

Fig. 2. nolight case.

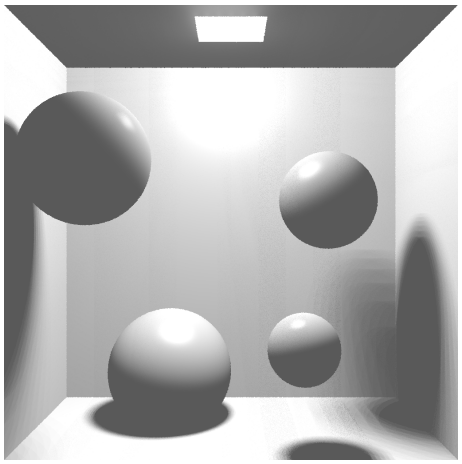

Fig. 3. notexture case.

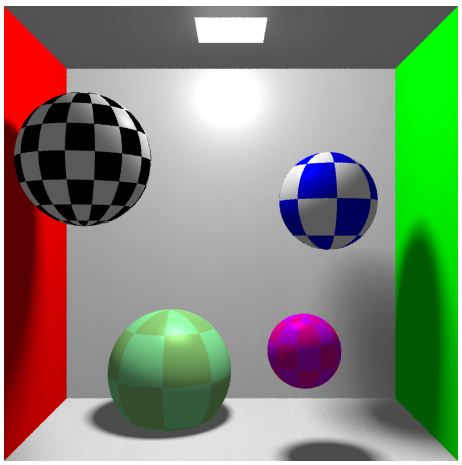

Fig. 4. smaller sample case.

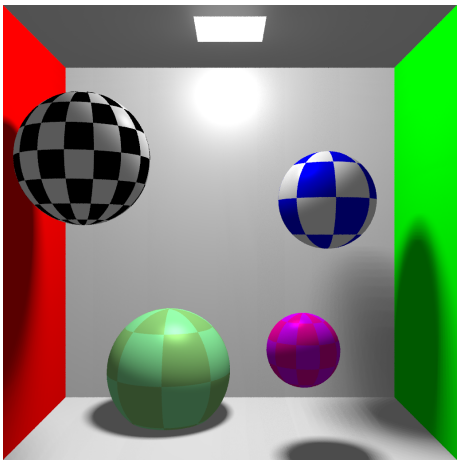

Fig. 5. anti-anilise case.

4 CONCLUSION

In this assignment, I learned how to create a pin-hole camera modle, and render some objects with different texture algorithms,which is interesting and useful.In the modern society, games, movies and other industry are tightlt relative to graphics. form texture mapping, we can generate more and more real image to make imaginary to reality.