CS557 Project #1

Yunfan Li

[liyunf@oregonstate.edu](mailto:liyunf@oregonstate.edu)

**1 Source Listings**

a1.rib

a1.sl

**2 What I did & Why it worked**

***2.1 What I did***

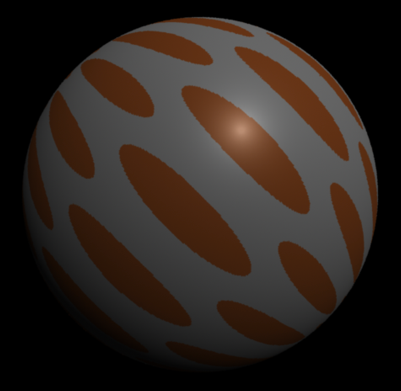
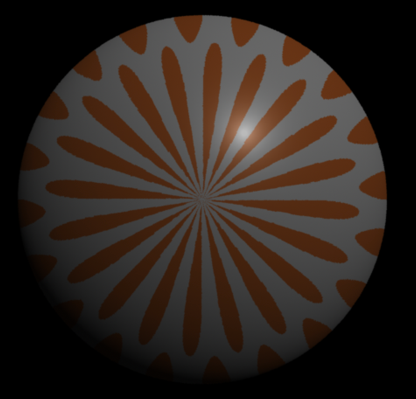
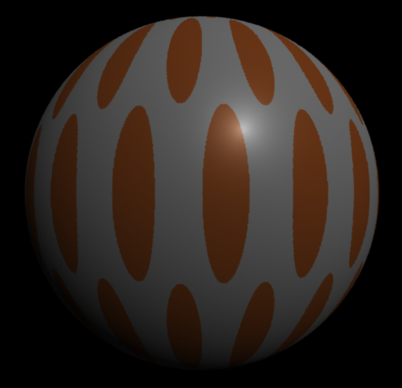
In my a1.rib file, I created two “uniform float” variables named “Ar” and “Br” respectively. Then tiff file was denoted as my output file format and the color mode of output file is RGB, the width and height of output file were both 512 with square pixels. Next two lines was used to create a specific light source so that any objects in this scene became more real. In next line, the projection of this scene was declared as “perspective”. After “WorldBegin”, I translate the canvas to (0, 0, 6) and specify a1.sl as surface shader of this scene. At the same time, two variables whose name is “Ar” and “Br” were assigned their value. Then the color and opacity of objects in this scene were set up. From “TransformBegin” to “TransformEnd”, a sphere of this scene was created and rotated 90 degrees on X-axis.

In my a1.sl file, global variable “u” and “v” were firstly used to calculate which box does this point belong to, “u” was timed 2.0 because it just covers 180 degrees instead of 360 degrees which “v” covers. Then the center point coordinate of this box was calculated using “numinu” and “numinv”. Eventually, the distance between this point and center point was checked to see whether it is less than 1. If it is, then changed color of this point to orange. If not, do nothing.

***2.2 Why it worked***

Because I checked every single point to see whether it located in an ellipse. If it is truly in an ellipse, the color of this point was changed. In macroscopic scene, the sphere will have a bunch of ellipses. So it should work as what project requires.

**3 Images**



**4 My Code**

***4.1 a1.rib***

##RenderMan RIB

version 3.03

Declare "Ar" "uniform float"

Declare "Br" "uniform float"

Display "a1.tiff" "file" "rgb"

Format 512 512 -1

ShadingRate 1

LightSource "ambientlight" 1 "intensity" [0.25]

LightSource "distantlight" 2 "intensity" [0.75] "from" [5 8 -10] "to" [0 0 0]

Projection "perspective" "fov" [70]

WorldBegin

Translate 0 0 6

Surface "a1" "Ar" 0.025 "Br" 0.1

Color [1 1 1]

Opacity [1 1 1]

TransformBegin

Rotate 90 1. 0. 0.

Sphere 3 -3 3 360

TransformEnd

WorldEnd

***4.2 a1.sl***

surface

a1 (float

Ar=0.4,

Br=0.4,

Ks=0.5,

Kd=0.5,

Ka=0.1,

roughness=0.1;

color specularcolor=color(1, 1, 1))

{

float up = 2. \* u;

float vp = v;

float numinu = floor( up / (2 \* Ar) );

float numinv = floor( vp / (2 \* Br) );

float uc = numinu \* 2. \* Ar + Ar;

float vc = numinv \* 2. \* Br + Br;

color dotColor = Cs;

if (mod(numinu+numinv, 2) == 0) {

if (pow((up-uc)/Ar, 2)+pow((vp-vc)/Br,2) <= 1 ) {

dotColor = (1., .5, .2);

}

}

varying vector Nf = faceforward( normalize(N), I );

vector V = normalize( -I );

Oi = 1.;

Ci = Oi \* ( dotColor \* ( Ka \* ambient() + Kd \* diffuse(Nf) ) + specularcolor \* Ks \* specular( Nf, V, roughness ) );

}