THE CYCLES RENDERING ENGINE

Introduction
Blender is in the process of moving from a traditional scanline/raytraced rendering engine to one that is more realistic. We look at both in CMPM25, but because the more advanced renderer is processor intensive and slow on less powerful machines we will use only the older rendering engine for practice and homework assignments. This has been around since Blender was first created and has undergone many changes. It is relatively fast, offers many material and textures options, but does not simulate light in an accurate way. Since the introduction of Cycles the basic rendering engine is no longer being developed.

Cycles
Cycles is what is known as an unbiased renderer. In simple terms this means a rendering technique that does not introduce any systematic error, or bias, into the way the image is calculated. Intensive development is now going into the Cycles rendering engine. Select it as the active renderer by choosing it from the drop down menu in the information bar:

When you go to the Materials tab in Properties you will now see this has completely changed. There is no longer a preview window or any of the long column of option panels. Instead there is a large button saying Use Nodes. Click this button.
When you do so the Surface panel will be populated by options for the type of surface, the color, and so on. However, the Material tab in Properties is no longer the place to develop your material. Instead, open up a Node Editor window.

Now you have the same information in a more compact, graphical, and versatile form.

**Exercise**

Download and open the CyclesRoomScene-01.blend file in [https://users.soe.ucsc.edu/~yonge/05_homework_resources/](https://users.soe.ucsc.edu/~yonge/05_homework_resources/) (5Mb download) if you want to see the same images on your screen as in this handout.

This is a scene with a table, two chairs, and a stage set wall and floor. If you render it you’ll see it’s only lit by ambient occlusion, and the rendering is dark with undefined shadows. The sky is a blend of a dark zenith to a light horizon.
Change the renderer to Cycles in the drop down menu in the Info window.

Disable Ambient Occlusion in the World tab in Properties and rerender.
Even though there are no lights in the scene and no AO, we can still see the scene - though it looks very blue. The reason for this is that now, in Cycles, the sky color is lighting the scene. This might be fine for an outside twilight scene, but not for a stage set. We need to disable the sky as a lighting element.

Go to the World panel in Properties and click on Use Nodes.

Immediately the panel will look very different. We can see now where the blue light comes from, but if we click on that color bar and reduce the Alpha to zero, the light is unchanged when the scene is rendered again. The answer is to click on the Surface option and choose Transparent BSDF. BSDF stands for Bidirectional Scattering Distribution Function, by the way, but that’s not important right now. What matters is that the sky is now transparent and cannot give any color to the scene. When we render now the entire image is black. That’s what we want - no lighting - but not good in another, since we want the sky to be transparent so we can composite our render over another image. In the Internal renderer we simply define the alpha to be transparent in the Render
tab, but when we go there now that panel has disappeared - in fact the whole panel has changed, with many different options.

We created transparency in the unrendered parts of the image by going to the Film panel and checking the

---

![Film panel](image)

---

So let’s add some lighting. Although it’s possible to use Blender Internal lights with the Cycles rendering engine, it’s not best practice. We create Cycles lights by defining a mesh - preferably a 3D mesh to avoid rendering artefacts - as having an Emission color and value.

Switch to a wireframe view of the scene and create a Cylinder above the table. Rescale and rotate it to resemble a four Blender unit long, 0.2 BU diameter, fluorescent tube. Increase the Emission strength to 40. Switch to a Rendered view of the scene.

---

![Scene with lighting](image)

---

You should see a grainy but realistically lit version of the scene similar to the image above.
This accurately previews the lighting you will see in a full render of the scene. It’s clearly too dark; select the light and duplicate it with Shift-D. You’ll notice that there are no outline indications of selection in Rendered view, but you can see single selections in the Outliner as the names are in white. You can also select multiple objects in the Outliner by holding down the Shift key and left clicking (note - LEFT clicking) but the interface does not show multiple selections at this point. Commands such as G-Y or R-Z will work as in all other viewing modes, though the preview may take a couple of seconds to update depending on the speed of your graphic card.

**GPU vs. CPU**

Cycles is much slower than the internal renderer at this stage, though most of the Blender development team’s attention is directed towards speeding it up. One way to help, if you have a good graphics card in your computer, is to enable GPU rendering. Do this by going to User Preferences and then the System tab, then check CUDA and your graphic card option. Most laptops have fairly unimpressive graphic cards, however, and this may not make as much difference as it will with a well configured recent desktop machine.

**Back to our scene ...**

Let’s duplicate the light with Shift_D and move it along the X axis. This is when the tripod in the lower left of the 3D windows proves its value. Note how the scene is much brighter. The illumination in a scene depends on the area of the light emitting meshes times the Emission value for each of those meshes. If you double an emission object’s dimension alone one dimension, the illumination of the scene will be doubled as you’ve doubled that Emission object’s surface area. However, you’ve also changed its shape, so the lighting and shadows will not be precisely the same. If you keep it the same size but double its Emission value, then the illumination will also be doubled, but the lighting and shadows will be unchanged.

Let’s say this is good enough for illumination and color, and start to add some other materials.
Cycles materials

These are best created and editing in the Node Editor, and there is no avoiding the fact that you have to relearn everything about materials and textures to do this. Let’s start by creating a semi-gloss wood floor.

Select the floor and give it a basic Diffuse BSDF shader. This means it’s going to be rendered as matte with a white color. But we want it to be semi-gloss; this means adding another shader, and the best way to do this is in the Node Editor.

Split the 3D window horizontally and make one of them a Node Editor window. The leftmost of the highlighted three buttons should be selected, and you should see these two nodes in the Editor window.

The round cornered boxes are nodes and the flexible line connecting them is a noodle. Move the Diffuse BSDF node up and to the left, then hit Shift-A and add a Shader-Glossy BSDF node. We can only select one node to the Surface input of the Material Output node, so we need to mix the output of these two nodes in another node, a Shader-Mix Shader. Dragging the Mix Shader node over the noodle connecting Diffuse BSDF with Material Output will drop it into the chain. Next drag a noodle from the Glossy BSDF
node output to the lower of the Mix shader node inputs.

Look at the rendered preview in the 3D window and you should see a faint reflection of the furniture and walls in the floor. It’s faint for two reasons: the matt and shiny materials are mixed evenly (a factor of 0.500 in the Mix shader node) and also the Glossy BSDF shader has a default roughness value of 0.200.

Reduce the roughness value of the Glossy shader to zero, and increase the Mix shader to 1.000 (100% of input 1; remember Blender starts numbering from zero, and the top input is 0). Now the floor is a perfect mirror.

Now reduce the Mix shader value to zero (ie: 100% of the top, or number 0, input). The floor is now matt.

Give the Diffuse BSDF shader some color to make this clear. Notice the large number of sparks in the scene, known as fireflies. We can reduce this by going to the Render tab and changing Clamp Indirect to 0.03; don’t change Clamp Direct or the lighting will become much darker. Now the preview and rendered version will look much better. Note that both are at 10 samples right now; you’ll want to change the Render value to something much higher, at least 50.
The fireflies have largely gone in preview mode, but you’ll notice that reflections are darker. Changing the clamping values is a tradeoff between quality and consistency.

Let’s change the Mix shader Factor to zero again to get all Diffuse shading.

We want to make this a wood floor, so we need to bring in an image. In Cycles this is done with the dot at the end of the Diffuse shader in the Properties panel.

Click on this to bring up the Color input options.

Choose Image Texture.
The floor will turn purple, as no texture image has been defined (just as in the internal renderer). Use the Open button to navigate to a suitable image.

Because Cycles defaults to Generated mapping rather than UV, the image will appear. But what if we want to control its size and rotation? First, we need to define some UV mapping. Go to a top view of the floor in the 3D Window, enter Edit mode, select all, and hit U to make a default UV map. Go back to the Node Editor.

We want to use the UV mapping as the vector input to the Image Texture node.
Add an Input-Texture Coordinate node before the Image Texture node and connect the UV output to the Image Texture vector input.

But we still don’t have any control over the scale, position, and rotation of the texture image on the floor. To do that, we need to add one more node - a Vector-Mapping node between the Texture Coordinates and the Image Texture.

Make sure the new node’s Texture tab is enabled and then you can change the wood texture’s UV mapping as needed.
**Procedural materials**

These materials don’t rely on an image and so never repeat. Developing procedural materials is complex, but fortunately there is a growing library of Blender procedural materials online. Start with the CG Masters tutorial at [http://www.cgmasters.net/free-tutorials/blender-cycles-tutorial-stonemarble-how-to-make-any-texture/](http://www.cgmasters.net/free-tutorials/blender-cycles-tutorial-stonemarble-how-to-make-any-texture/).

**Conclusion**

Cycles is highly complex, but is capable of producing dazzling images - a video of professional quality animations is at [https://www.youtube.com/watch?v=wDRTlzLNK0g](https://www.youtube.com/watch?v=wDRTlzLNK0g). Learning Blender is at least a quarter's work in itself, but there is a good series of videos beginning at [http://www.youtube.com/watch?v=-CYNKSxE7368](http://www.youtube.com/watch?v=-CYNKSxE7368). This is number 3 in the Blender Bootcamp series but the first to introduce materials and rendering in Cycles. BlenderGuru also has some excellent tutorials starting with [http://www.blenderguru.com/tutorials/introduction-to-cycles/](http://www.blenderguru.com/tutorials/introduction-to-cycles/) as well as a comprehensive guide to shaders at [http://www.blenderguru.com/articles/cycles-shader-encyclopedia/](http://www.blenderguru.com/articles/cycles-shader-encyclopedia/) and to inputs at [http://www.blenderguru.com/articles/cycles-input-encyclopedia/](http://www.blenderguru.com/articles/cycles-input-encyclopedia/).

Chris Yonge  -  20170318