Assignment Outline

This week, we will explore lighting and shading using Blender’s built-in, ray tracing render engine: Cycles. Cycles implements all that you coded up in the last assignment and much more. It provides most of the features and power of modern ray tracing under the hood, allowing the average user to utilize it all without needing any programming experience.

This assignment document is a mix of instructional tutorials and exercises. The exercises are marked with **Action:** and are what you’ll demonstrate during the grading session. As with the previous assignments, the complete list of what to submit are at the end of the document.

Collaboration, Office Hours, and Piazza Policies

All policies from here on will be the same as they were for the past assignments.

Blender Cycles

Turning on Cycles is as simple as going to the Render Properties in the Properties Editor and changing the Render Engine to Cycles.

![Cycles in Blender Properties Editor](image)

Figure 1: **Cycles** is Blender’s ray tracer. By default, Blender has the render samples for Cycles set to some large number. You may want to change it to a much lower number, e.g. 4, to make the render faster (for now as you play around with it).
You can preview the render with the right-most button in the Viewport toolbar above the camera view toggle. Note though that Blender may start running very slowly if you try to make scene edits while previewing the Cycles (ray traced) render. As mentioned in lecture, ray tracing isn’t suited for real time rendering!

Figure 2: Toggling to camera view in the Blender Viewport. The button for toggling the render preview is also located in the Viewport toolbar right above the camera view toggle.

For now, as you experiment with Cycles, you may want to change the default Max Samples under the Render settings from 4096 to a much lower number, e.g. 4 in Figure 1. We’ll talk about sampling later in the class, but for now, know that the max samples is basically how many times Blender will repeat the render for better results. Too high of a number will cause your render to take an incredibly long time.

To render your scene as an image, you can go to Render near File, then click Render Image, or press the F12 hotkey. When the Blender Render window finishes, save the result under Image to a directory, e.g. $CS148_DIR/hw4/imgs/images. The following sections will have you set up various scenes to try rendering with Cycles.

4 Lighting in Blender

Blender by default provides 4 different types of lights for you to place into your scene: point, area, spot, and directional lights. We covered two of these in Lectures 7-8 (point and area), and the other two are pretty intuitive to pick up.

There are two ways to add any of these lights to your scene. One option is to add a new Light from the Add menu. The other is to modify an existing light in your scene in the Properties Editor under Object Data Properties. There, you can change your light between all four types.
Figure 3: Ways to add a particular light type to your scene.

Figure 4: Transforming a light or camera using the Properties Editor. Note that in the case of a point light, only translations matter. Scaling has no effect on either.
4.1 Point Lights

Let’s try playing with a point light. First, we’ll need to set up an actual scene. Here are the steps we took to set up the scene in Figures 1 and 2:

1. Import the provided Stanford bunny.obj file that came with the assignment into a new Blender scene (with the default cube deleted), and translate it along the y-axis by 5. If you’ve forgotten how to import an .obj file, then you might want to look back at HW2. For speed and simplicity, you may just use the Properties Editor for transformations here, e.g. as shown in Figure 4 (for objects too in addition to the light(s) and camera).

2. Add a plane object via **Add → Mesh → Plane**, and scale it by 10 in both the x and y directions. This will be the floor plane.

3. Add another plane object, also scale it by 10 in both the x and y directions, then rotate it about the x-axis by 90 degrees, and translate it along the y-axis by 10. This will be the back wall plane.

4. Change the camera transformations from the default numbers to simply \((0, -10, 2.5)\) for the location and \((90, 0, 0)\) for the rotation.

5. Change the default light location to \((0, 0, 5)\). The rotation doesn’t matter, since this is a point light (self check: can you answer why?).

6. Toggle to the camera view in the Viewport as shown in Figure 2. Change your Render engine to Cycles, and toggle to the render preview. You should get a similar result to what’s shown in Figure 1.

Alternatively, if you’ve already begun to make objects for your final project, or just have some spare objects from previous work and experience, then you are very welcome to set up your own scene instead of the above! You may want to toggle the camera view back and forth when arranging your object(s), light(s), and camera as shown in Figure 2. Additionally, you may want to toggle off the render preview and go back to the default solid mode view to lighten the computational load on your computer as you make modifications.

Once you’re done setting up your scene, try experimenting with the placement of the point light(s). Remember from lecture that the irradiance (i.e. the light power or amount of light that is hitting an object surface) varies based on the tilt angle of the surface with the light as well as the distance to the light. See Figure 5.

![Figure 5: Irradiance varies based on the tilt angle and distance of the surface to the (point) light.](image-url)
4.2 Area Lights

Now let’s try playing with area lights. First, disable your point light(s) from Section 4.1 (using the eyeball and camera icons in the Scene Collection tree above the Properties Editor), and add an area light instead. Transform the area light however you see fit for your scene. Recall that the radiance of an area light varies based on the tilt angle between it and the object surface. See Figure 6.

Notice how the area light results in a much less uniform spread of light, as all the light rays are restricted to coming from a particular area. This is in contrast to the point light, where light rays are sent out indiscriminately in all directions around the point. The larger area of an area light results in softer looking shadows (e.g. less dark contrast) than what might result from a point light (see Figure 7 vs. Figure 8).

Figure 6: Radiance varies based on the tilt angle and distance of the surface to the (area) light.

Figure 7: An example of a scene illuminated with only an area light. The light rays are restricted to coming only from a particular area, though having multiple rays going in the same direction can lead to softer shadows than what would be produced by a point light.
Figure 8: An example of a scene illuminated with only a point light. Point lights tend to produce much harsher shadows with darker contrasts to that of the surroundings, as each direction only gets one ray of light.

4.3 Spotlights

Spotlights model lights that are engineered to emit light rays in a cone-like shape. Imagine lamps or streetlights in real life where the light is often surrounded by some sort of lampshade or outer object designed to concentrate the light. This leads to a light type that also aims its light rays at a specific area, like area lights, but still spreads its light rays in multiple directions, similar to point lights.

Spotlights are popular in modeling due to how common they tend to be in our everyday lives. In addition, they also are able to produce more dramatic lighting compared to a point light or area light. For instance, consider Figure 9 compared to Figure 8. It’s much easier to concentrate the light from a spotlight on a particular part of a scene than from a point light, whose light rays go in every direction.

Figure 9: An example of a scene illuminated with only a spotlight for more dramatic lighting. The spotlight excels at concentrating light all into one spot in the scene.
4.4 Directional Lights

The last type of light is the directional light or “sunlight” as Blender calls it. The idea of directional lights is that they don’t have their own locations (i.e. no actual positions in world space), but rather, they shoot light rays uniformly across all of space in the same direction. Think of the sun in real life. The sun is so far away from us that all its light rays at any given instant might as well be shining in the same direction. Hence why Blender calls them sunlights.

Mathematically, when we consider directional lights for lighting computations (i.e. the lighting equation), we just ignore any concept of distance and use the same direction for the light vector regardless of where we are in the scene. So unlike what happens with point lights or area lights, an object that’s twice as far away as another from a directional light still gets the same amount of light. Only the direction matters for a directional light.

Figure 10: An example of a scene illuminated with only a directional aka “sun” light. Directional lights have no concept of position. Instead, they shine uniformly across all of space in one direction.

4.5 Lighting a Scene

**Action:** Now try lighting up your own scene with all 4 types of light. Try to get a feel of how each light might come up for what you want to do in your final project. From Figure 11, we see that using a mix of lights can lead to a more natural look with soft shadows and more even illumination.

When you’re satisfied with your scene, render and save it as an image using Blender Cycles (in an appropriate enough resolution to see the details) for the grading session. See Section 3 for instructions on how to render using Cycles. You can change the resolution of your output in the Output Properties tab of the Properties Editor, though the standard 1920x1080 at 50% should be good enough. Note that you do not have to use the example bunny scene if you want to make your own individual scene!
5 Smooth Shading

In lecture, we talked about shading techniques to make geometry appear smoother without actually adding more geometry. More specifically, we talked about Phong shading. In Blender, we can toggle Phong shading for objects in the Blender Viewport and also for rendering with Cycles.

5.1 Shade Flat vs Shade Smooth

**Action:** Let’s take a look at Phong (aka smooth) shading in Blender with an example scene (like how we did so for lighting in Section 4.1):

1. Make a new Blender scene and replace the default cube with a UV sphere.
2. Add a plane and scale it by 5 in both the x and y directions. Then move it along the z-axis by -1.
3. Change your Render engine to Cycles, and toggle to the render preview.

The scene setup should look like the one in Figure 12. Select the UV sphere, then right-click to choose between Shade Flat and Shade Smooth. Render the scene once with flat shading, and again with smooth shading. Save both images for the grading session. If you want, you can also experiment with smooth shading for your objects in the scene you built for Section 4.5.
6 Environment Lighting

In lecture, we talked about measuring incoming light and then using it to render synthetic objects in the scene. We can do this in Blender using (1) environment textures, also known as High Dynamic Range Images (HDRIs), or (2) the Nishita sky model. HDRIs are captures of real-world scenes, while Nishita Sky is a sky model that approximates the sun and the sky.

**Action:** Add either a HDRI or Nishita Sky to your scene from Section 4.5. Render and save the image for the grading session. You may want to remove the back plane if you’re copying the example bunny scene.

6.1 Environment Texture - HDRI

High Dynamic Range Image, or HDRI, is a type of image frequently used for realistic lighting in 3D environments, usually in EXR format. It is created by combining pictures taken in the same scene with different exposure values, allowing it to contain much more color and illumination information than normal images. It used to be that we needed professional equipment to capture HDRIs, but nowadays you can do it yourself using 360 cameras. There are free CC0 HDRI libraries online, such as the popular HDRI Haven, which has a wide variety of environments from sunny outdoors to indoor studios.

To use HDRI textures in Blender, download an HDRI online (1k resolution is enough for our use) or pick one that comes with Blender (in the installation folder: Blender 3.3/3.3/datafiles/studiolights/world). Then, in the Properties Editor, go to World Properties and click on the yellow dot in the Color field to change the background color to Environment Texture. From there, click Open to browse to your HDRI. Swap to the render preview (with Cycles enabled) to see your HDRI in effect.
6.2 Nishita Sky

Nishita Sky is a recent addition to Blender starting from version 2.9. It is an improved version of the sky model proposed by Nishita et al. in 1993. This model lets you control the sky texture with only a few intuitive parameters. You can read more about the Nishita sky model here and see a video demo in Blender 2.9’s release notes.

In Blender, we can use Nishita Sky by setting the Color field to Sky Texture in World Properties. Play with the settings to get the sky you like.
7 BSDF Materials

BSDF (Bidirectional Scattering Distribution Function) is a generalized version of the BRDF and BSDF that were discussed in lecture. BSDF describes all the possible ways that light can be scattered by a surface. Most of the materials in Blender have BSDF suffixes, which means they all scatter light (in different ways).

**Action:** Select 3 objects from your scene in Section 4.5, go to Material Properties, and add materials to each of them. Render and save the image using Cycles for grading. You can expand the Preview panel to see how the material looks on template models.

The default material is the Principled BSDF, a shader based on the Disney principled model. It can emulate a wide variety of materials in our daily life. You can also switch to another BSDF from the Surface field. You can learn about the parameters and view some examples for the Principled BSDF in the Blender documentation as well as other BSDFs here.

We will discuss how to use the Shader Editor to make complex materials in HW8. If you’d like to try it out now, then you can watch the official tutorials here and here, or Blender Guru’s explanation here.
8 Quiz Questions

During the grading session, the CA will choose one of the following questions at random to ask. Please be prepared to give a brief (~1 min) answer based on the lecture material.

- How does the distance between a light and an object as well as the tilt angle between the two affect the irradiance on the surface of the object? How does the concept of a tilt angle also come up for the radiance of a light?
- What causes color bleeding in real life? Give a high level description of what we would need to do with our objects in a scene to model color bleeding.
- Conceptually, what is a BRDF and how are they involved in the lighting equation? For instance, if the incoming radiance is red light, and the BRDF involved is for a blue material object, what might you expect for the color of the outgoing radiance?
- What is the difference between the way we compute Gouraud shading vs the way we compute Phong shading? What is an issue with Gouraud shading that does not appear with Phong shading?
- Explain the ambient, diffuse and specular components of the Phong Reflection Model. What does each component model visually for the shading of an object?

9 How to Submit / How to get Graded

On Monday, October 24:

- if you are assigned the 2-3 or 3-4 PM time block, then please join this Zoom call.
- if you are assigned the 5-6 or 6-7 PM time block, then please join this other Zoom call.

Those with alternative times as scheduled via email should join the 2nd (5-7) Zoom call at the same alternative time window.

If you are working with a partner, then both of you may get graded together at a single time block. Both of you need to be in the Zoom call to get graded. When one partner is let in, they should tell the CA to also let the other partner into the same room.

10 What to Demo / Submit

During the grading session, please screenshare with your CA and have all of the following ready to show beforehand:

- (1 pt) A Cycles rendered image from Section 4.5 that utilizes all the types of lights in Blender. Briefly describe the use of each type of light.
- (1 pt) The images showing flat vs smooth shading in Blender. Briefly describe the difference between the two.
- (1 pt) A Cycles rendered image featuring either a HDRI or Nishita Sky.
- (1 pt) A Cycles rendered image featuring 3 different BSDF materials. Briefly describe the materials you explored.
- (1 pt) Answer the quiz question given by the CA correctly.