Contents

Chapter 1

How Cameras Work

Let's say it is a sunny day and you are standing in a field a few meters from a cow. You use the camera on your phone to take a picture of the cow. How does that whole process work?

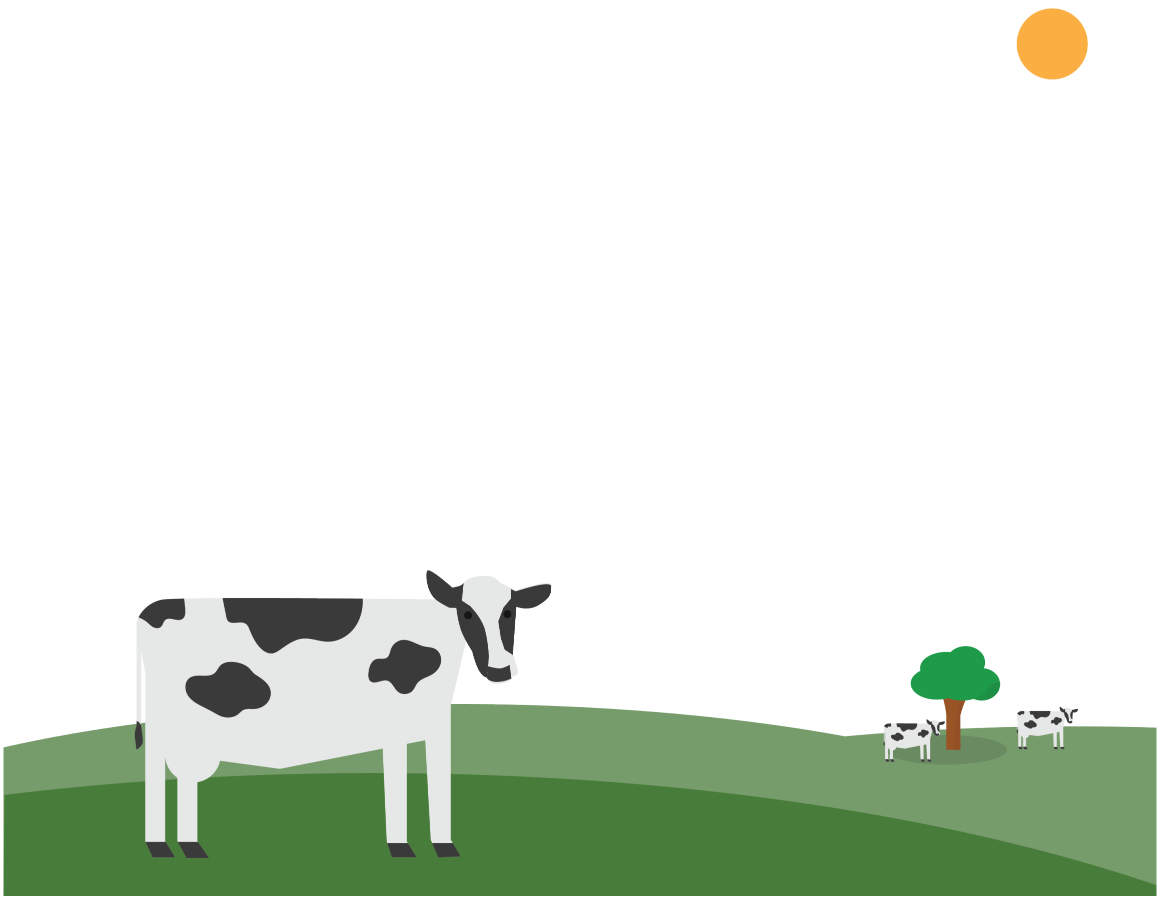


Figure 1.1: A cow in a field.

1.1 The Light That Shines On the Cow

The sun is a sphere of hot gas. About 70% of the gas is hydrogen. About 28% is helium. There's also a little carbon, nitrogen, and oxygen.

Gradually, the sun is converting hydrogen into helium through a process known as "nuclear fusion" (which we will be discussing more in a future chatper). A large amount of heat is created in this process. This heat makes the gases glow.

How does heat make things glow? The heat pushes the electrons into higher orbitals. When they come back down to a lower orbital, they release a photon of energy, which travels away from the atom as an electromagnetic wave.

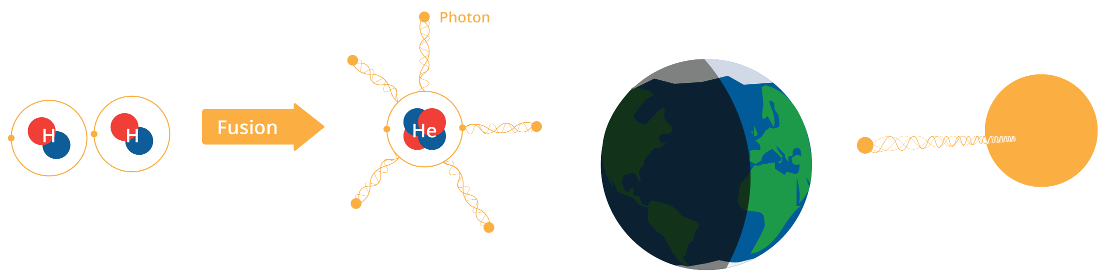


Figure 1.2: Photons are released when the sunlight hits the cow.

Heat is not the only way to push the electrons into a higher orbital. For example, a fluorescent lightbulb is filled with gas. When we pass electricity through the gas, its electrons are moved to a higher orbital. When they fall back to a lower orbital, light is created.

What is the frequency of the wave that the photon travels on? Depending on what orbital it falls from and how far it falls, the photon created has different amounts of energy. The amount of energy determines the frequency of the electromagnetic wave.

Formula for energy of a photon

If you want to know the amount of energy E in a photon, here is the formula:

$$E = \frac{hc}{\lambda}$$

where c is the speed of light, λ is the wavelength of the electromagnetic wave, and h Planck's constant: $6.63 \times 10^{-34} m^2 kg/s$

For example, a red laser light has a wavelength of about 630 nm. So, the energy in each photon is:

$$\frac{(300 \times 10^6)(6.63 \times 10^{-34})}{630 \times 10^{-9}} = 3.1 \times 10^{-19} \text{ joules}$$

In the sun, there are several kinds of molecules and each has a few different orbitals that the electrons can live in. Thus, the light coming from the sun is made up of electromagnetic waves of many different frequencies.

We can see some of these frequencies as different colors, but some are invisible to humans, such as ultraviolet and infrared.

1.2 Light Hits the Cow

When these photons from the sun hit the cow, the hide and hairs of the cow will absorb some of the photons. These photons will become heat and make the cow feel warm. Some of the photons will not be absorbed – they will leave the cow. When you say "I see the cow," what you are really saying is "I see some photons that were not absorbed by the cow."

Different materials absorb different amounts of each wavelength. A plant, for example, absorbs a large percentage of all blue and red photons that hit it, but it absorbs only a small percentage of the green photons that hit it. Thus, we say "That plant is green."

White things absorb very small percentages of photons of any visible wavelength. Black things absorb very *large* percentages of photons of any visible wavelength. That is why, on a hot summer day, a black car with black seats and interior will heat up on the inside much hotter than a white car.

Before we go on, let's review: The sun creates photons that travel as electromagnetic waves of assorted wavelengths to the cow. Many of those photons are absorbed, but some are not. Some of those photons that are not absorbed go into the lens of our camera.

1.3 Pinhole camera

The simplest cameras have no lenses. They are just a box. The box has a tiny hole that allows photons to enter. The side of the box opposite the hole is flat and covered with film or some other photo-sensitive material.

The photons entering the box continue in the same direction they were going when they passed through the hole. Thus, the photons that entered from high hit the back wall at a low point. The photons that came from the left hit the back wall on the right. This is how the image is projected onto the back wall, rotated 180 degrees; what was up is down,

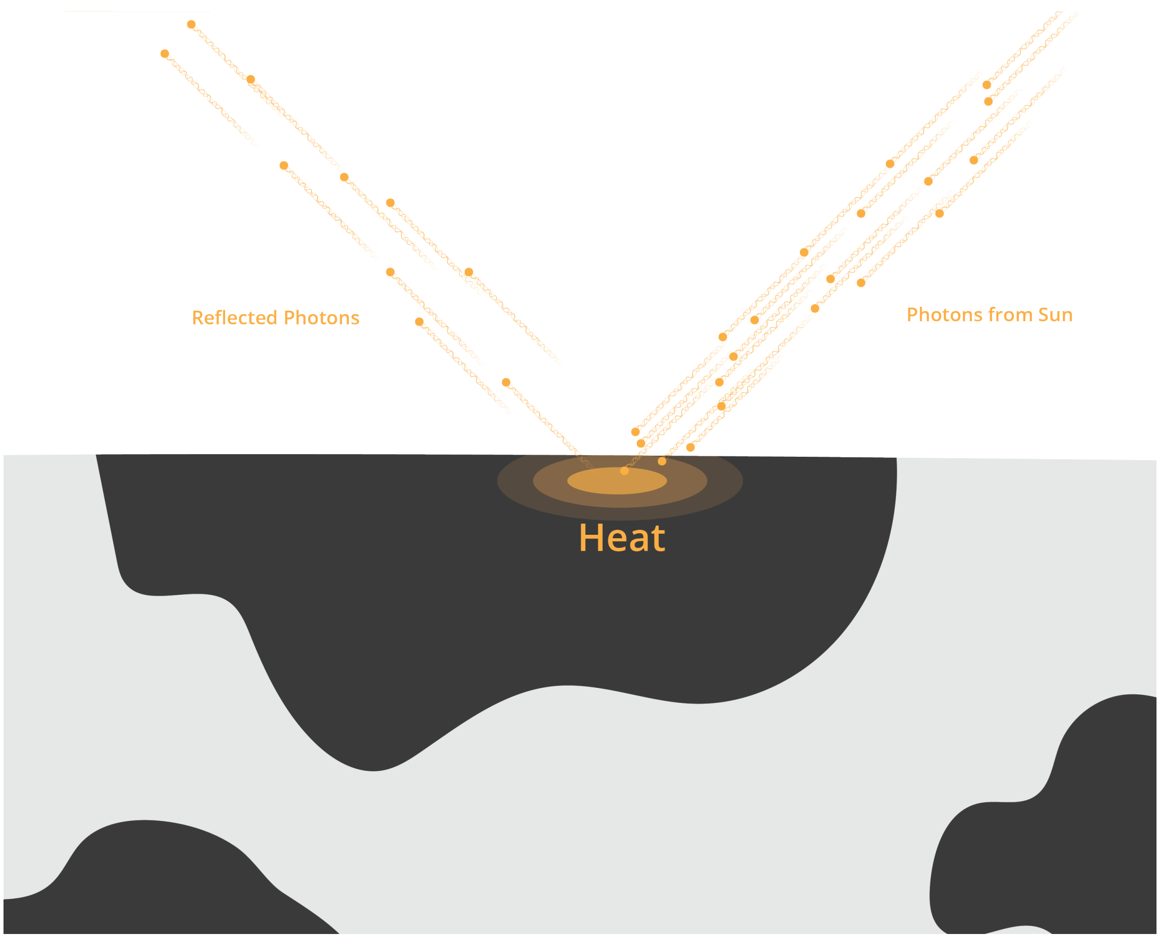


Figure 1.3: Photons hitting the cow create a heated spot, where some are absorbed.

what was on the left is on the right.

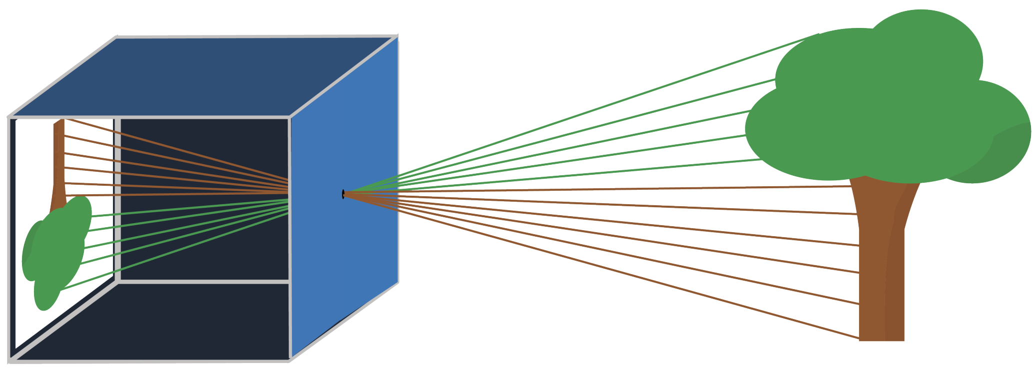


Figure 1.4: A pinhole camera flips the image it “sees” by 180°.

Exercise 1 Height of the image

FIXME: cow swap Let's say that that the pinhole is exactly the same height as the shoulder of the cow, and that the shoulder is directly above one hoof. This means the pinhole, the shoulder, and the hoof form a right triangle.

Now, let's say that the camera is being held perpendicular to the ground. The pinhole, the image of the shoulder, and the image of the hoof on the back wall of the camera now also form a right triangle.

These two triangles are similar.

The shoulder is 2 meters from the hoof. The cow is standing 3 meters from the camera. The distance from the pinhole to the back wall of the camera is 3 cm. How tall is the image of the cow on the back wall of the camera?

Working Space

Answer on Page 25

1.4 Lenses

Now, a quick review: A photon leaves the sun in some random direction. It travels 150 million km from the sun and hits a cow. It is not absorbed by the cow, and heads off in a new direction. It passes through the pinhole and hits the back wall of the camera. That seems incredibly improbable, right?

It actually is relatively improbable, especially if there isn't a lot of light — like you are taking the picture at dusk. To increase the odds, we added a *lens* to the camera. If you focus a lens on a wall and you draw a dot on that wall, the lens is designed such that all the photons from the dot that hit the lens get redirected to the same spot on the back wall of the camera — regardless of which path it took to get to the lens.

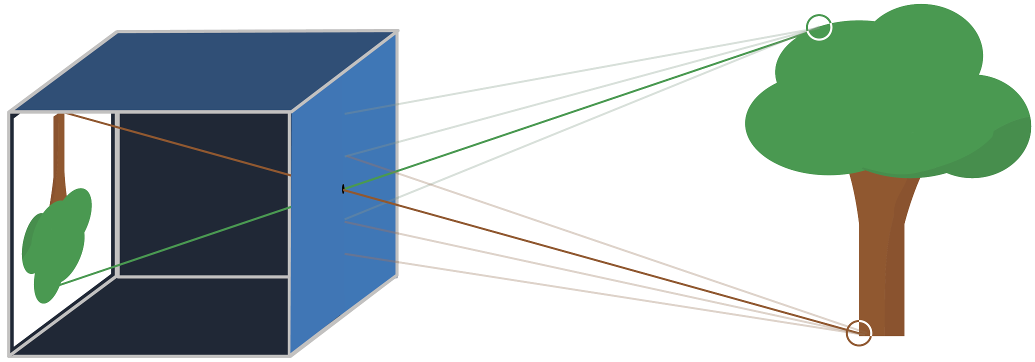


Figure 1.5: A pinhole with a lens allows you to choose the points it reflects.

Note that the image still gets flipped. There is a *focal point* that all the photons pass through.

The distance from the lens to its focal point is called the lens's *focal length*. Telephoto lenses, that let you take big pictures of things that are far away, have long focal lengths. Wide-angle lenses have short focal lengths.

1.5 Sensors

The camera on your phone has a sensor on the back wall of the camera. The sensor is broken up into tiny rectangular regions called pixels. When you say a sensor is 6000 by 4000 pixels (most common ratio for photography), we are saying the sensor is a grid of 24,000,000 pixels: 6000 pixels wide and 4000 pixels tall.

Each pixel has three types of cavities that take in photos. One of the cavities measures the amount of short wavelength light, like blues and violets. One of the cavities measures the long wavelength light, like reds and oranges. One of the cavities measures the intensity of wavelengths in the middle, like greens. Thus, if your camera has a resolution of 6000×4000, the image is 72,000,000 numbers: Every one of the 24,000,000 pixels yeilds three numbers: intensity of long wavelength, mid wavelength, and short wavelength light. We call these numbers "RGB" for Red, Green, and Blue. The RGB values range from $0 - 255$ for each channel. We will talk more about this when hexidecimal is introduced.

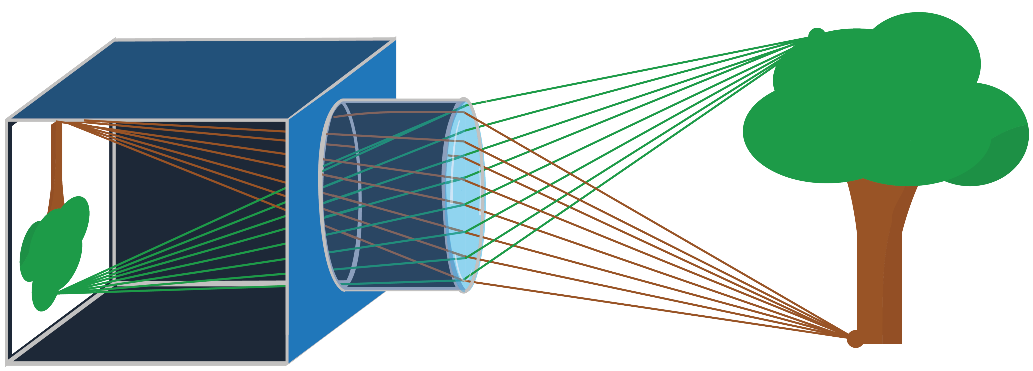


Figure 1.6: A lens still flips the image, but introduces a focal point.

Chapter 2

How Eyes Work

Dr. Craig Blackwell has made a great video on the mechanics of the eye. You should watch it: https://youtu.be/Z8asc2SfFHM

Mechanically, your eye works a lot like a camera. The eye is a sphere with two lenses on the front: The outer lens is called the *cornea*, while the second lens is simply called "the lens." Between the two lenses is an aperture that opens wide when there is very little light, and closes very small when there is bright light. The opening is called the *pupil* and the tissue that forms the pupil is called the *iris*. When people talk about the color of your eyes, they are talking about the color of your iris. The blackness at the center of your iris is your pupil.

There are two types of photoreceptor cells in your retina: rods and cones. The rods are more sensitive; in very dark conditions, most of our vision is provided by the rods. The cones are used when there is plenty of light, and they let us see colors.

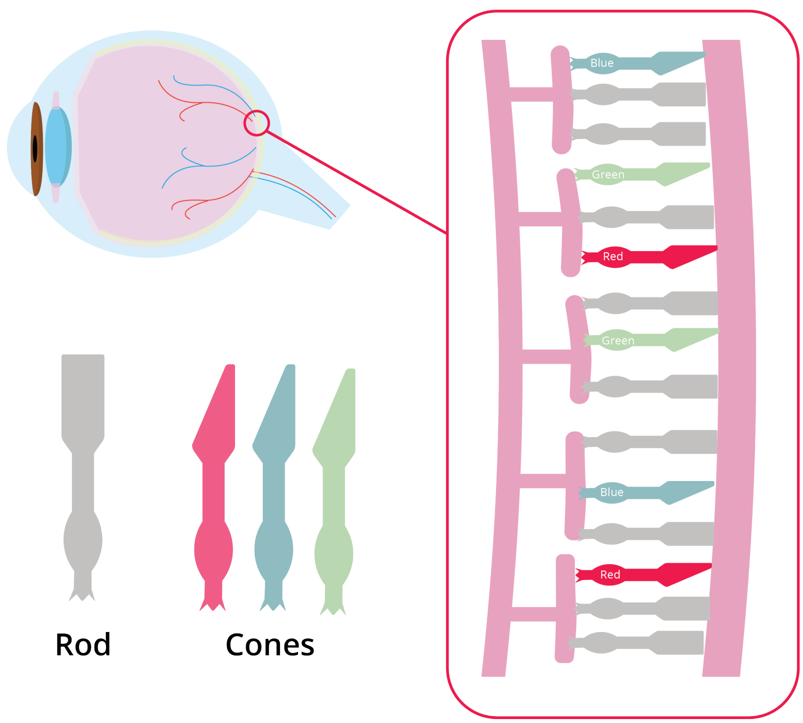


Figure 2.1: The cones and rods of the eye act as sensors and cognitive vision.

The white part around the outside of the eyeball? That is called the *sclera*.

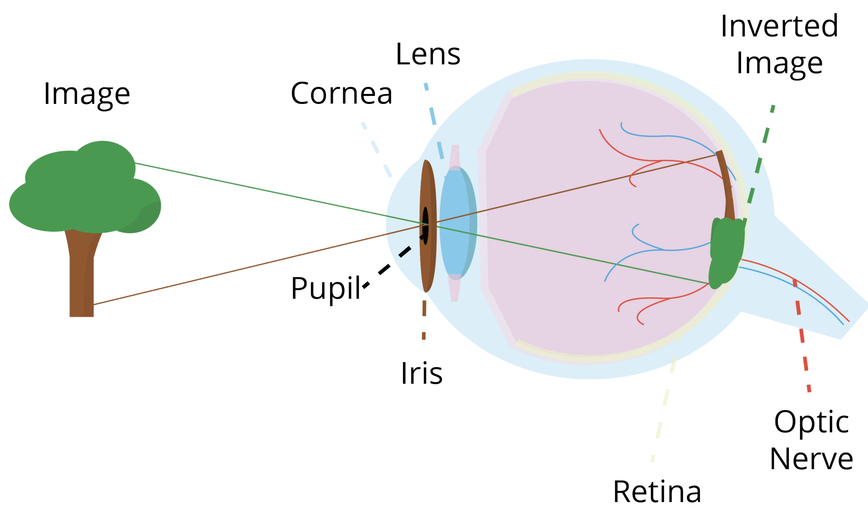


Figure 2.2: The eye works just like a camera's lens.

The walls of the eye are lined inside with the *retina*, which has sensors that pick up the light and send impulses down the optic nerve to your brain.

Just like a camera, the images are flipped when they get projected on the back of the eye (see Figure 2.2).

2.1 Eye problems

Now that you know the mechanics of the eye, let's go over a few things that commonly go wrong with the eye.

2.1.1 Glaucoma

The space between your cornea and lens is filled with a fluid called *aqueous humor*. To feed the cells of the cornea and lens, the aqueous humor carries oxygen and nutrients like blood would, but unlike blood, it is transparent so you can see. Aqueous humor is constantly being pumped into and out of that chamber. If aqueous humor has trouble exiting, the pressure builds up and can damage the eye. This is known as *glaucoma*. See Figure 2.3.

2.1.2 Cataracts

The lens should be clear. As a person ages, the proteins in the lens break down and clump together, becoming opaque. This can also be accelerated by diabetes, too much exposure to sunlight, obesity, and high blood pressure. From the outside, the eye will look cloudy. This is called a *cataract*, and it makes it difficult for the person to see.

This problem can be corrected, however. The person's cloudy lens is removed and replaced with a clear, manufactured lens.

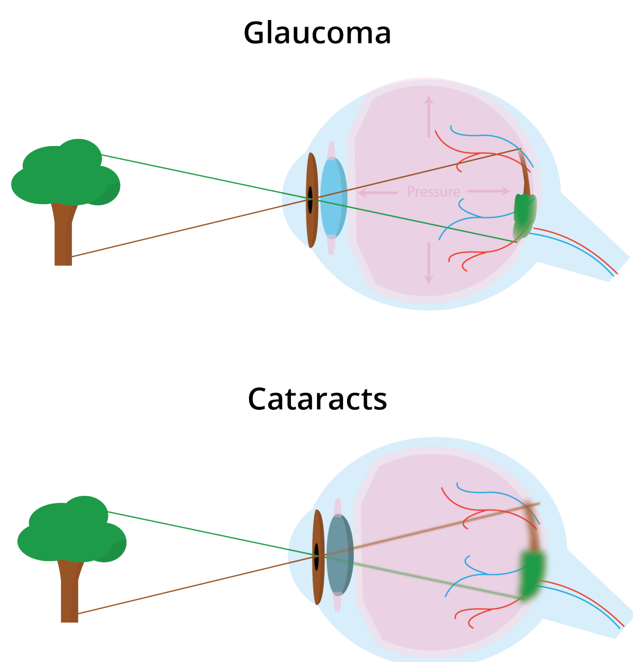


Figure 2.3: Cataracts and Glaucoma represented in the eye cross section.

2.1.3 Nearsightedness, farsightedness, and astigmatism

If you are in a dark room and a tiny LED is turned on, the photons from that LED can pass through your cornea in many different places. If your eye is focusing on that light correctly, all the photons should meet up at the same place on the retina.

FIXME: Diagram here

If the lenses are bending the light too much, the photons meet up before they hit the retina and get smeared a bit across it. To the person, the LED would appear blurry. The eye is said to be *nearsighted* or *myopic*. This signals that near objects appear blurry, but farther objects appear correctly.

If the lenses are not bending it enough, the photons would meet up behind the retina. Once again, they get smeared a bit across the retina and the LED looks blurry to the person. The eye is said to be *farsighted* or *hyperoptic*. The objects which are distant can appear clearly, while closer objects are heavily blurred.

Your lenses are supposed to bend the photons the same amount vertically and horizontally. If one dimension is focused, but the other is myopic or hyperoptic, the eye is said to have *astigmatism*.

Myopia, hyperoptia, and astigmatism can be corrected with glasses or contact lenses. Doctors can also do surgical corrections, usually by changing the shape of the cornea.

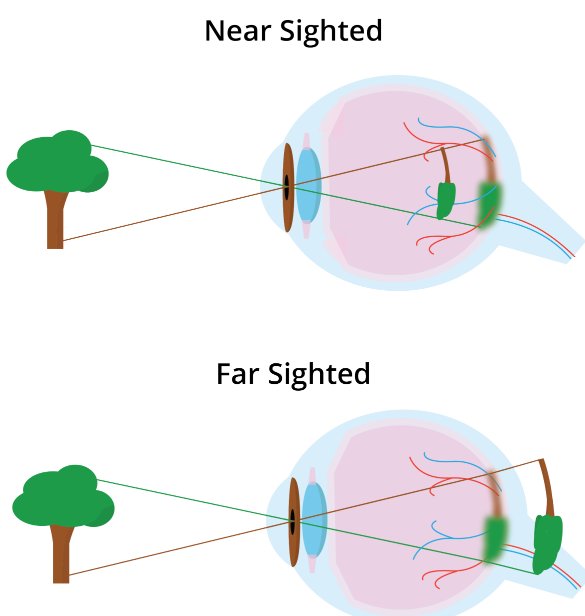


Figure 2.4: Nearsightedness versus farsightedness.

2.2 Seeing colors

TED-Ed has made a good video on how we see color. Watch it here: https://youtu.be/l8_fZPHasdo

When a rainbow forms, you are seeing different wavelengths separating from each other. In the rainbow:

- Red is about 650 nm.
- Orange is about 600 nm.
- Yellow is about 580 nm.
- Green is about 550 nm.
- Cyan is about 500 nm.
- Blue is about 450 nm.
- Violet is about 400 nm.

If you shine a light with a wavelength of 580 nm on a white piece of paper, you will see yellow.

However, if you shine two lights with wavelengths of 650 nm (red) and 550 nm (green), you will also see yellow.

Why? Our ears can hear two different frequencies at the same time. Why can't our eyes see two colors in the same place?

As mentioned above, the cone photoreceptors in our eyes let us see colors. There are three kinds of cones:

- Red: Cones that let us see the frequencies up to about 700nm.
- Green: Cones that are most sensitive to frequencies near 550nm.
- Blue: Cones that are most sensitive to frequencies near 450nm.

When a wavelength of 580 nm hits your retina, it excites the red and green receptors, and your brain interprets that mix as yellow.

Similarly, when light that contains both 650 nm and 550 nm waves hits your retina, it excites the red and green receptors, and your brain interprets that mix as yellow.

You can't tell the difference!

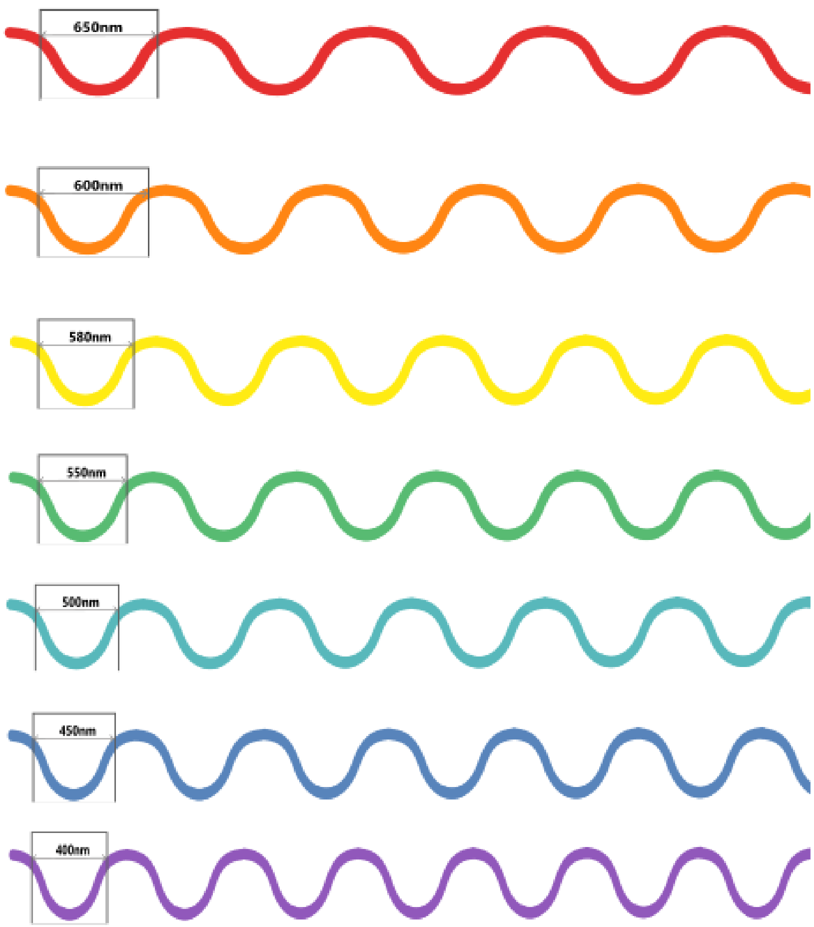


Figure 2.5: All colors have different wavelengths.

Now we know why the sensors on the camera are RGB. The camera is recording the scene as closely as necessary to fool your eye.

A TV or a color computer monitor only has three colors of pixels: red, green, and blue. By controlling the mix of them, it creates the sensation of thousands of colors to your eye.

2.3 Pigments

A color printer works in the opposite fashion. Instead of radiating colors, it puts pigments on the paper that absorb certain frequencies. A pigment that absorbs only frequencies near 650 nm (red) will appear to your eye as cyan. This makes sense, because the sensation of cyan is created when your blue and green receptors are activated.

Thus, pigment colors come in:

- Cyan: absorbs frequencies around red
- Magenta: absorbs freqencies around green
- Yellow: absorbs frequencies around blue

If you buy ink for a color printer, you know there is typically a fourth ink: black. If you put cyan, magenta, and yellow pigments on paper, the mix won't absorb all the visible spectrum in a consistent manner. Our eyes are pretty sensitive to this, so we would see brown. This is why we add black ink to get pretty grays and blacks.

We call this approach to color CMYK (as opposed to RGB). If an artist is creating an image to be viewed on a screen, they will typically make an RGB image. If they are creating an image to be printed using pigments, they typically create a CMYK image. (Most of us don't care so much — we just let the computer do conversions between the two color spaces for us.)

Chapter 3

Images in Python

An image is usually represented as a three-dimensional array of 8-bit integers. NumPy arrays are the most commonly used library for this sort of data structure.

If you have an RGB image that is 480 pixels tall and 640 pixels wide, you will need a $480 \times 640 \times 3$ NumPy array.

There is a separate library (imageio) that:

- Reads an image file (like JPEG files) and creates a NumPy array.
- Writes a NumPy array to a file in standard image formats

Let's create a simply python program that creates a file containing an all-black image that is 640 pixels wide and 480 pixels tall. Create a file called `create_image.py`:

```
import NumPy as np
import imageio
import sys

# Check command-line arguments
if len(sys.argv) < 2:
    print(f"Usage {sys.argv[0]} <outfile>")
    sys.exit(1)

# Constants
IMAGE_WIDTH = 640
IMAGE_HEIGHT = 480

# Create an array of zeros
image = np.zeros((IMAGE_HEIGHT, IMAGE_WIDTH, 3), dtype=np.uint8)

# Write the array to the file
imageio.imwrite(sys.argv[1], image)
```

To run this, you will need to supply the name of the file you are trying to create. The extension (like .png or .jpeg) will tell imageio what format you want written. Run it now:

```
python3 create_image.py blackness.png
```

Open the image to confirm that it is 640 pixels wide, 480 pixels tall, and completely black.

3.1 Adding color

Now, let's walk through through the image, pixel-by-pixel, adding some red. We will gradually increase the red from 0 on the left to 255 on the right.

```
import NumPy as np
import imageio
import sys

# Check command-line arguments
if len(sys.argv) < 2:
    print(f"Usage sys.argv[0] <outfile>")
    sys.exit(1)

# Constants
IMAGE_WIDTH = 640
IMAGE_HEIGHT = 480

# Create an array of zeros
image = np.zeros((IMAGE_HEIGHT, IMAGE_WIDTH, 3), dtype=np.uint8)

for col in range(IMAGE_WIDTH):

    # Red goes from 0 to 255 (left to right)
    r = int(col * 255.0 / IMAGE_WIDTH)

    # Update all the pixels in that column
    for row in range(IMAGE_HEIGHT):
        # Set the red pixel
        image[row, col, 0] = r

# Write the array to the file
imageio.imwrite(sys.argv[1], image)
```

When you run the function to create a new image, it will be a fade from black to red as you move from left to right:

Now, inside the inner loop, update the blue channel so that it goes from zero at the top to 255 at the bottom:

```
    # Update all the pixels in that column
    for row in range(IMAGE_HEIGHT):

        # Update the red channel
        image[row,col,0] = r

        # Blue goes from 0 to 255 (top to bottom)
        b = int(row * 255.0 / IMAGE_HEIGHT)
        image[row,col,2] = b

imageio.imwrite(sys.argv[1], image)
```

When you run the program again, you will see the color fades from black to blue as you go down the left side. As you go down the right side, the color fades from red to magenta.

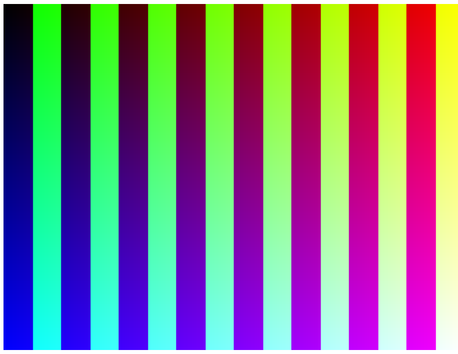
Notice that red and blue with no green looks magenta to your eye.

Next, let's add some stripes of green:

```
import NumPy as np
```

```
import imageio
import sys

# Check command line arguments
if len(sys.argv) < 2:
    print(f"Usage sys.argv[0] <outfile>")
    sys.exit(1)

# Constants
IMAGE_WIDTH = 640
IMAGE_HEIGHT = 480
STRIPE_WIDTH = 40
pattern_width = STRIPE_WIDTH * 2

# Create an image of all zeros
image = np.zeros((IMAGE_HEIGHT, IMAGE_WIDTH, 3), dtype=np.uint8)

# Step from left to right
for col in range(IMAGE_WIDTH):

    # Red goes from 0 to 255 (left to right)
    r = int(col * 255.0 / IMAGE_WIDTH)

    # Should I add green to this column?
    should_green = col % pattern_width > STRIPE_WIDTH

    # Update all the pixels in that column
    for row in range(IMAGE_HEIGHT):

        # Update the red channel
        image[row,col,0] = r

        # Should I add green to this pixel?
        if should_green:
            image[row,col,1] = 255

        # Blue goes from 0 to 255 (top to bottom)
        b = int(row * 255.0 / IMAGE_HEIGHT)
        image[row,col,2] = b

imageio.imwrite(sys.argv[1], image)
```

When you run this version, you will see the previous image in half the stripes. In the other half, you will see that green fades to cyan down the left side, and yellow fades to white down the right side.



3.2 Using an existing image

imageio can also be used to read in any common image file format. Let's read in an image and save each of the red, green, and blue channels out as its own image.

Create a new file called separate_image.py:

```
import imageio
import sys
import os

# Check command line arguments
if len(sys.argv) < 2:
    print(f"Usage {sys.argv[0]} <infile>")
    sys.exit(1)

# Read the image
path = sys.argv[1]
image = imageio.imread(path)

# What is the filename?
filename = os.path.basename(path)

# What is the shape of the array?
original_shape = image.shape

# Log it
print(f"Shape of {filename}:{original_shape}")

# Names of the colors for the filenames
colors = ['red','green','blue']

# Step through each of the colors
```

```
for i in range(3):

    # Create a new image
    newimage = np.zeros(original_shape, dtype=np.uint8)

    # Copy one channel
    newimage[:,:,i] = image[:,:,i]

    # Save to a file
    new_filename = f"{colors[i]}_{filename}"
    print(f"Writing {new_filename}")
    imageio.imwrite(new_filename, newimage)
```

Now, you can run the program with any common RGB image type:

```
python3 separate_image.py dog.jpg
```

This will create three images: `red_dog.jpg`, `green_dog.jpg`, and `blue_dog.jpg`.

Appendix A

Answers to Exercises

Answer to Exercise 1 (on page 8)

The two triangles are similar; one is 2 m and 3m, the other is x cm and 3 cm.

The image of the cow is 2 cm tall.

Index