

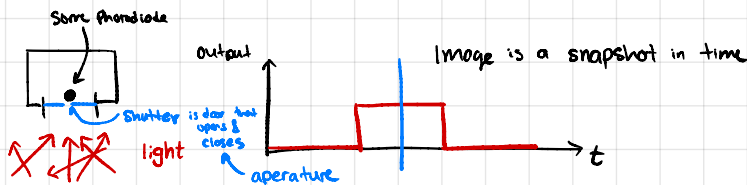
# 1/9/2020 Image Formation 1

- goes by Patrick, OH by email request
- Probably Monday 7PM TA OH

Image processing:  $\text{image(s)} \rightarrow \text{image}$   
Computer vision:  $\text{image(s)} \rightarrow \text{information}$  } Lots of overlap  
Machine Learning: 588 (ECE)  
- Python this semester!

- ① Bad image  $\rightarrow$  Better
- ② Computational photography

Image is 1 value (pixel)



Mechanism is complicated

- open shutter for a moment, then close it
- do it fast so we get info for an instant instead of averaging
- too short of a time  $\rightarrow$  not enough photons
- compromise: microsecond/ns shutter speed/exposure time

- 1) Open shutter
  - 2) Collect light
  - 3) Close shutter
- Shutter speed is a controllable parameter (exposure time)  $\rightarrow$  more photons flow in  $\rightarrow$  brighter image  
③ aperture control

Knobs we can turn  $\rightarrow$  4) Amplification (ANALOG GAIN) Camera can mess with this for best experience with digitizing

- 5) A  $\rightarrow$  D conversion (DIGITAL DOMAIN)
- $\rightarrow$  range of integers [0-255] (1 Byte / 8 bits) } very standard: 1 Byte per color channel per pixel

Some terminology:

- $\rightarrow$  Power (watts) (Energy flux (energy/time))
  - $\rightarrow$  Radiant Flux (watts) (light per time)
  - $\rightarrow$  Luminant flux (radiant flux, but only visible light) (Lumens/Lm)  $\rightarrow$  weighted wavelengths based on eye sensitivity
  - $\rightarrow$  Irradiance - radiant flux/area  $\rightarrow$  matters to the human eye
  - $\rightarrow$  Illuminance - Luminant flux/area (lux) (lumens/m<sup>2</sup>·time)  $\rightarrow$  important for cameras
- $\rightarrow$  from empirical experiments in the twenties

Illuminance  $\rightarrow$  voltage

$\rightarrow$  luminant flux per area  
 $\downarrow$   
amount of visible light (energy) per area

Python Environment  $\rightarrow$  set up before Tuesday!

use 3.8

- python -m venv venv
  - source venv/bin/activate
  - pip install numpy
  - pip install -u pip
  - pip install matplotlib
  - pip install pyqt5
- $\rightarrow$  starts virtual environment and with deactivate

pip install opencv-python

Visual studio code

Environment setup

File setup.py

```
import numpy as np
import matplotlib.pyplot as plt
import cv2
```