THE PINHOLE CAMERA

Every point on the sensor plane corresponds to a unique ray.

ADMINISTRIVIA

- Read through course website
  - Syllabus
  - Problem Set and Resource Section
  - Late Policies
  - Collaboration & Academic Honesty Policies
- Join Piazza
- Review Pre-req Slides
- Install Anaconda, LaTeX, Git
- Submit your Public Key
- Submit Problem Set 0
- Post on piazza if you have issues with your setup.
The pinhole camera

\[ s_y = ? \]

By Similar Triangles:\n\[ s_y = -\frac{f}{z} \]

We will go in-depth into camera projection in a few weeks....

By Similar Triangles:\n\[ s_y = -\frac{f}{z}, \quad s_x = -\frac{f}{z} \]

\[ (x, y, z) \Rightarrow \left(-\frac{f}{z}, -\frac{f}{z}, \frac{f}{z} \right) \]
THE PINHOLE CAMERA

We will go in-depth into camera projection in a few weeks ....

\[(x, y, z) \mapsto (\frac{-f_x}{z}, -\frac{f_y}{z})\]

Patterns on a "Fronto-parallel Pane"

We will go in-depth into camera projection in a few weeks ....

\[(x, y, z) \mapsto (\frac{-f_x}{z}, -\frac{f_y}{z})\]

Patterns on a "Fronto-parallel Pane" scaled by the same \( z \)

THE PINHOLE CAMERA

\[(x, y, z) \mapsto (\frac{-f_x}{z}, -\frac{f_y}{z})\]

THE PINHOLE CAMERA

\[(x, y, z) \mapsto (\frac{-f_x}{z}, -\frac{f_y}{z})\]

Different \( z \) will get scaled differently; "perspective distortion"
We will go in-depth into camera projection in a few weeks....

Different z will get scaled differently: "perspective distortion"

\[(x, y, z) \mapsto \left(-\frac{f_x}{z}, -\frac{f_y}{z}\right)\]

Not new to digital cameras!

Basic principle known to Mozi (470-390 BCE),
Aristotle (384-322 BCE)

Gemma Frisius 1558

Eclipsomania 2017!

Basic principle known to Mozi (470-390 BCE),
Aristotle (384-322 BCE)

Drawing aid for artists: described by Leonardo da Vinci (1452-1519)

Image via Subhra Rana Maji

Source: A. Efros

Source: A. Efros
THE PINHOLE CAMERA

Sensor:

- \( E(x, y, t) \): Light energy, per unit area per unit time, arriving at point \((x, y)\) at time \(t\)
  - Here, \(x, y\) are real numbers (in meters) denoting actual position on the sensor plane.
- \( I[n_x, n_y] \): Intensity measured by the sensor element at grid location \(n_x, n_y\)
  - Here, \(n_x, n_y\) are integers, indexing pixel location.
  - Note the convention \([\cdot]\) for “indexing”, while \((\cdot)\) for \(E\) because it is a function.
- \( p(x, y) \): is a sensitivity function for a pixel, assuming \((0,0)\) is the center of the pixel.
  - \( p(\cdot, \cdot) \) is ideally 1 inside pixel (for example, within \([-W/2,-H/2]\) to \([W/2,H/2]\)), and 0 outside. But may have attenuation at boundaries.
  - \( \bar{x}_{n_x}, \bar{y}_{n_y} \) is the location (in meters) of the center of the sensor element
  - \( p(x - \bar{x}_{n_x}, y - \bar{y}_{n_y}) \) is the sensitivity function for the pixel at \([n_x, n_y]\), “centered” at \((\bar{x}_{n_x}, \bar{y}_{n_y})\).

Defining \( q \) as the “quantum efficiency” of the sensor: Ratio of Light Energy to Charge/Voltage

\[
\int E(x, y, t) \ p(x - \bar{x}_{n_x}, y - \bar{y}_{n_y}) \ q \ dx \ dy
\]
Rate at which charge/voltage increases in sensor element \(n_x, n_y\) at time \(t\).

An image capture involves “exposing” the image for an interval \(T\) (seconds).

So the total intensity is going to involve integrating the charge/voltage rate over that interval.
\[ I[n_x, n_y] = \int_{t=0}^{T} \left[ \int E(x, y, t) \ p(x - \bar{x}_{n_x}, y - \bar{y}_{n_y}) \ q \ dx \ dy \right] \ dt \]

- \( n_x, n_y \) are integers indexing pixels in image array.
- \((x, y)\) is spatial location
- \( I[n_x, n_y] \) is recorded pixel intensity.
- \( E(x, y, t) \) is light “power” per unit area incident at location \((x, y)\) on the sensor plane at time \( t \)
- \((\bar{x}_{n_x}, \bar{y}_{n_y})\) is the “center” spatial location of the pixel / sensor element at \([n_x, n_y]\).
- \( p(x, y) \) is spatial sensitivity of the sensor (might be lower near boundaries, etc.)
- \( q \) is quantum efficiency of the sensor (photons/energy to charge/voltage)
- \( T \) is the duration of the exposure interval.

CCD/CMOS sensors measure total energy or “count photons” that arrived during exposure.

\[ I^0[n_x, n_y] = \int_{t=0}^{T} \left[ \int E(x, y, t) \ p(x - \bar{x}_{n_x}, y - \bar{y}_{n_y}) \ q \ dx \ dy \right] \ dt \]

\[ I \leftarrow I^0 \]

**SHOT NOISE**

- Caused by uncertainty in photon arrival
- Actual number of photons \( K \) is a discrete random variable with Poisson distribution
  \[ P(K = k) = \frac{\lambda^k e^{-\lambda}}{k!} \]
- \( \lambda \) is the “expected” number of photons. In our case, \( \propto I^0 \)
\[ I^0[n_x, n_y] = \int_{2t=0}^{T} \left[ \int E(x, y, t) p(x - \bar{x}_n, y - \bar{y}_n) q \, dx \, dy \right] \, dt \]

**SHOT NOISE**

- Property of Poisson distribution: Mean and Variance both equal
- Often, shot noise is modeled with additive Gaussian noise with “signal dependent” variance:
  \[ I \sim I^0 + \sqrt{I^0} \, \epsilon_1 \]
  where \( \epsilon \sim \mathcal{N}(0, 1) \) (Gaussian random noise with mean 0, variance 1).

\[ I \sim \mathcal{N}(I^0, I^0) \]

**DIGITIZATION**

- Final step is rounding and clipping (by an analog to digital converter)
  \[ I \leftarrow \text{Round} \left( gI^0 + g\sqrt{I^0} \, \epsilon_1 + \sqrt{\left( g^2 \sigma^2_{2a} + \sigma^2_{2b} \right)} \, \epsilon_2 \right) \]

\[ I = \min \left( I_{\text{max}}, \, \text{Round} \left( gI^0 + g\sqrt{I^0} \, \epsilon_1 + \sqrt{\left( g^2 \sigma^2_{2a} + \sigma^2_{2b} \right)} \, \epsilon_2 \right) \right) \]

**WHY STUDY THIS?**

- To understand the degradation process of noise (if we want to denoise / recover \( I^0 \) from \( I \)).
- To prevent degradation during capture, because we control exposure time \( T \) and ISO / gain \( g \).
- To understand the different trade-offs for loss of information from noise, rounding, and clipping.
Ignoring noise, what is the optimal $g$ for a given $I^0[n_x, n_y]$?

- Keep $g$ low so that most values of $gI^0[n_x, n_y]$ are below $I_{\text{max}}$.
- But if $g$ is too low, a lot of the variation will get rounded to the same value.
Note that here, our ‘ideal’ intensity is $g I^0$, everything else is noise.

**LIGHT VS AMPLIFICATION**

Say we have chosen the optimal target values for the product $g I^0$. Is it better:

- To have a higher $g$ and lower magnitude $I^0$
- To have a lower $g$ and higher magnitude $I^0$
- Depends, based on $\sigma_{2a}, \sigma_{2b}$

**Additional Reading (if interested):**

Increase exposure time $T$?

- If scene is static and camera is stationary:
  - $E(x, y, t)$ doesn’t change with $t \Rightarrow I^0 \propto T$
- If scene is moving …

\[
I^0[n_x, n_y] = \int_{t=0}^{T} \left[ \int E(x, y, t) \ p(x - \bar{x}_n, y - \bar{y}_n) \ q \ dx \ dy \right] dt
\]

Increase $E(x, y, t)$ itself. How?

- Take pictures outdoors, or under brighter lights.
- Don’t use a pinhole camera!
Small Pinhole: Focused beam, very little light.

Large Pinhole: More light, larger ‘circle of confusion’

Large Pinhole: More light, larger ‘circle of confusion’

Solution: Put a lens to focus
Thin lens model (approximation)

More light & localization of rays to sensor plane
but only for objects at the focusing distance D.
Photographers think about these tradeoffs every time they take a shot:

- Dynamic range and what part of the image should be well exposed (rounding and clipping)
- Choosing between:
  - ISO i.e. Gain & noise
  - Exposure Time & motion blur
  - F-stop i.e. aperture size & defocus blur

Can still tradeoff amount of light with amount of blur in out of focus regions (move up or down "F-stops"). Better trade-off than pinhole camera.
We left out an important term in this equation: wavelength

\[
I^0(n_x, n_y) = \int_{t=0}^{T} \left[ \int E(x, y, t) \ p(x - \bar{x}_n, y - \bar{y}_n) \ q \ dx \ dy \right] dt
\]

- Light carries different amounts of power in different wavelengths
- \( E(\lambda, x, y, t) \) now refers to power per unit area per unit wavelength
  - In wavelength \( \lambda \) incident at \((x, y)\) at time \(t\)
  - Both spectral and spatial density function
- \( q(\lambda) \): Quantum efficiency also a function of wavelength
  - CMOS/CCD sensors are sensitive (have high \( q \)) across most of the visible spectrum
  - Actually extend to longer than visible wavelengths (near infra red)
  - Why cameras have NIR filter, to prevent NIR radiation from being 'superimposed' on the image

Q: But this measures 'total' power in all wavelengths. How do we measure color?

Ans: By putting a color filter in front of each sensor element.

\[
I^0(n_x, n_y, c) = \int_{t=0}^{T} \left[ \int E(\lambda, x, y, t) \ p(x - \bar{x}_n, y - \bar{y}_n) \ \Pi_c(\lambda) \ q(\lambda) \ d\lambda \ dx \ dy \right] dt
\]

for \( c \in \{R, G, B\} \)

- \( \Pi_c \) is the transmittance of a color filter for color channel \( c \)
- E.g., \( \Pi_R \) will transmit power in (be high for) wavelengths in the red part of the visible spectrum and attenuate power in (be low for) other wavelengths.
- Sometimes also called "color matching functions"
COLOR

\[ I'[n_x, n_y, c] = \int_{t=0}^{T} \left[ \int E(\lambda, x, y, t) \ p(x - \bar{x}_n, y - \bar{y}_n) \ \eta(\lambda) \ d\lambda \ d\theta \ d\phi \right] \ dt \]

\[ \text{for } c \in \{R, G, B\} \]

- But we can only put one filter in front of each sensor element / pixel location.
- So color cameras “multiplex” color measurements: they measure a different color channel at each location.
- Usually in an alternating pattern called the Bayer pattern:

![Bayer Pattern Diagram]

- Note: a disadvantage is that color filters block light, so measured \( I' \) values are lower.
- That's why black and white / grayscale cameras are “faster” than color cameras.

FINAL STEPS

Final steps in camera processing pipelines (except for some DSLR cameras shooting in RAW):

- Filter Colors to Standard RGB:
  - Cameras often use their own color filters \( \mathbf{I}_c \).
  - Apply a linear transformation to map those measurements to standard RGB.
- White-balance: scale color channels to remove color cast from a non-neutral illuminant.
- Tone-mapping:
  - The simplest form is “gamma correction” (approximately raising each intensity to the power \((1/2.2)\))
  - Done based on standard developed for what old display devices expected
  - Fits the full set of measurable colors into the gamut that can be displayed / printed
  - Modern cameras often do more advanced processing (to make colors look vibrant)
- Compression

And that's how you get your PNG / JPEG images!

Optional Additional Reading: Szeliski Sec 2.3