COM SCI 188

Intro to Robotics

Lecture 5

Yuchen Cui

Winter 2025

Agenda

- Announcements
- Recap:
  - DH Parameters
  - Robot Control - PID

- Robot Control - MPC
- Robot Perception:
  - Cameras
  - Computer Vision

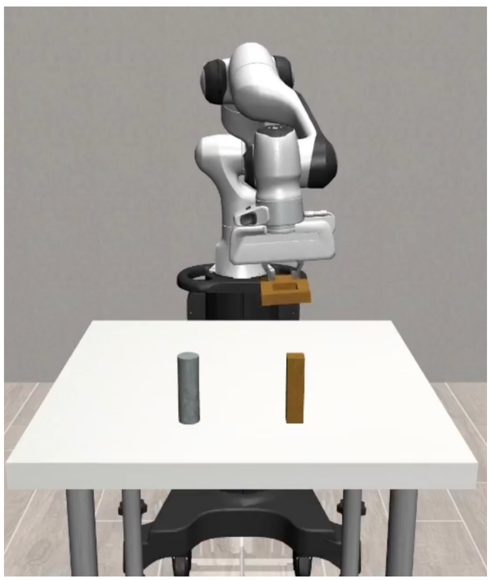
Announcements

- Coding Assignment 1 due Friday
  - Get started now if you haven't!
  - It can take a while to figure out all the parameters you need
  - Robosuite teleop demo script `demo_devices.py` can be useful!
- Problem Set 2 is out, due next Friday

Announcements

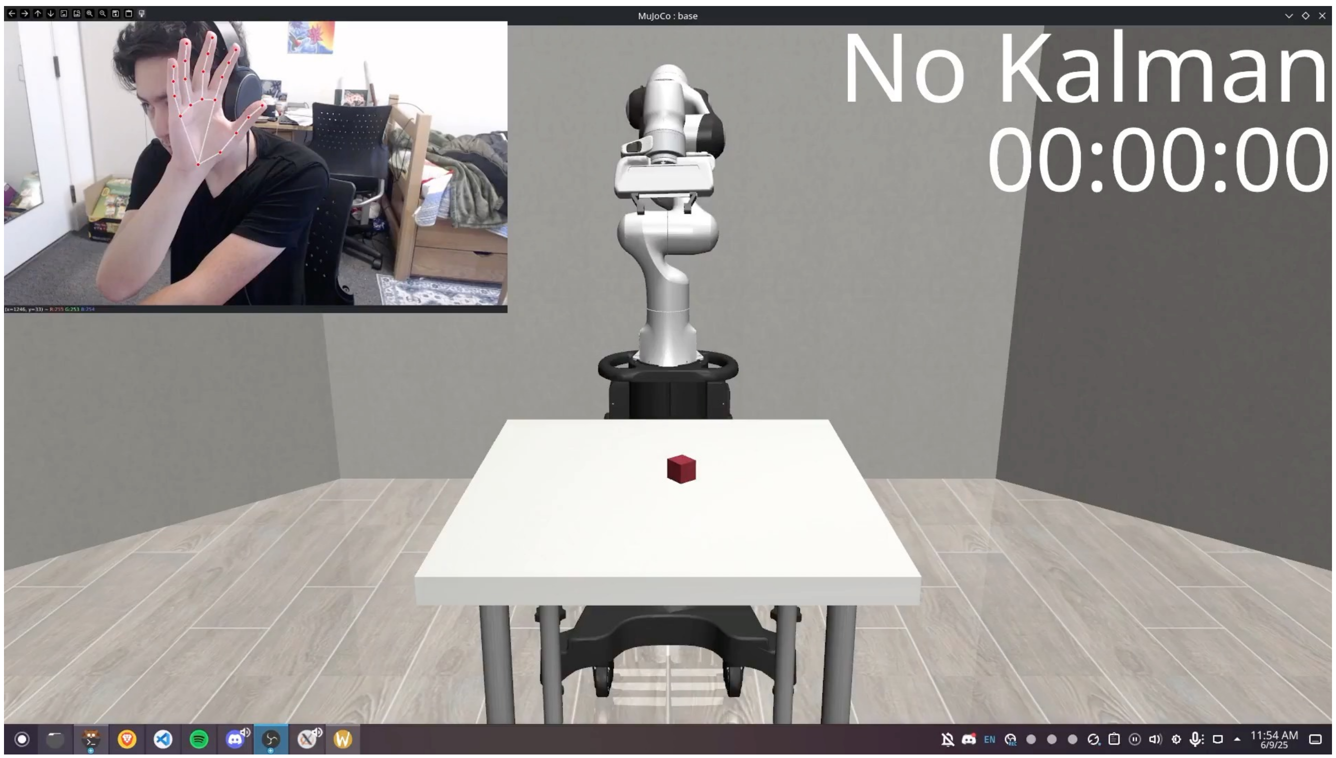
- Final project
  - **Start finding teammates now**
  - You will work in groups of 2 for the default project
  - You can work in groups up to 4 people if you do the open project

Default project from last year

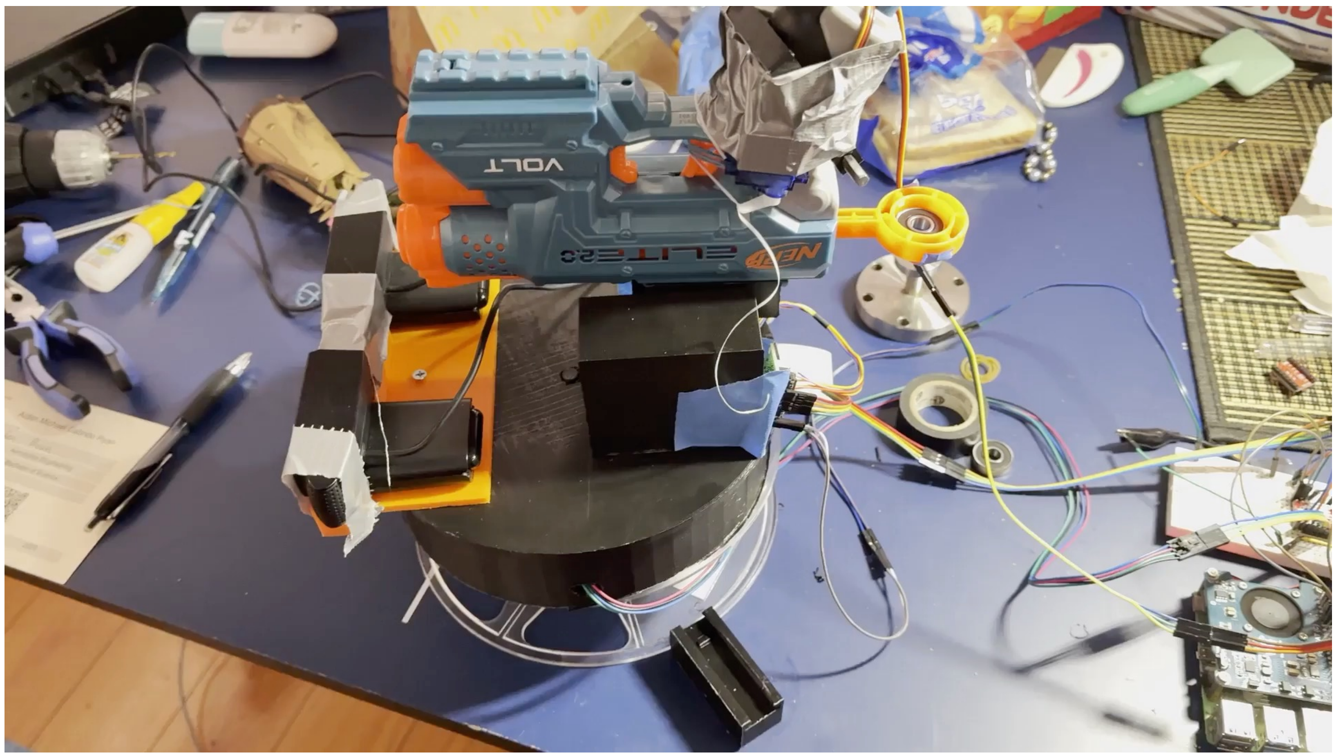


- Imitation Learning
  - Square Nut assembly
  - 100 human demos

Example open projects from last year



Example open projects from last year



Questions?

Recap: DH Parameters

Link twist · **Link length** · **Link offset** · **Joint angle**

$$T_{i-1,i} = \mathrm{Rot}(\hat{\mathrm{x}}, \alpha_{i-1})\mathrm{Trans}(\hat{\mathrm{x}}, a_{i-1})\mathrm{Trans}(\hat{\mathrm{z}}, d_i)\mathrm{Rot}(\hat{\mathrm{z}}, \phi_i)$$

Rotation of z around **x{i-1}** axis

displacement of z along **x{i-1}** axis

displacement of x along **z{i}** axis

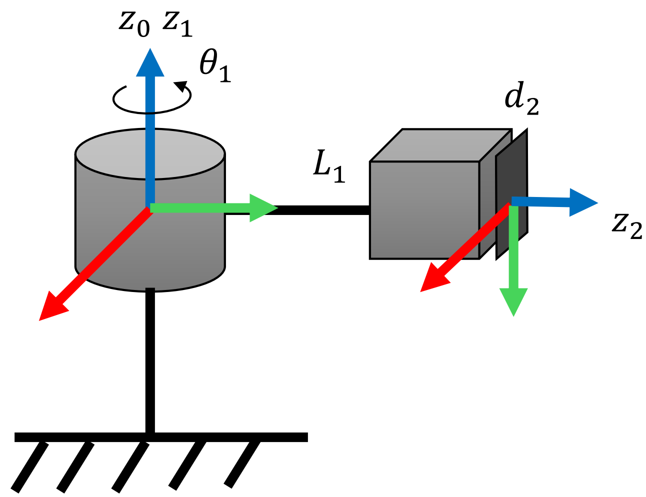
rotation of x around **z{i}** axis

Rules:

- ***Z axis*** is the joint axis
  - direction of rotation axis for revolute joints
  - direction of positive displacement for prismatic joints
- ***Origin*** is the point where mutually perpendicular line intersect axis {i-1} **
- ***X axis*** is along the direction of mutually perpendicular line pointing from {i-1} to {i}

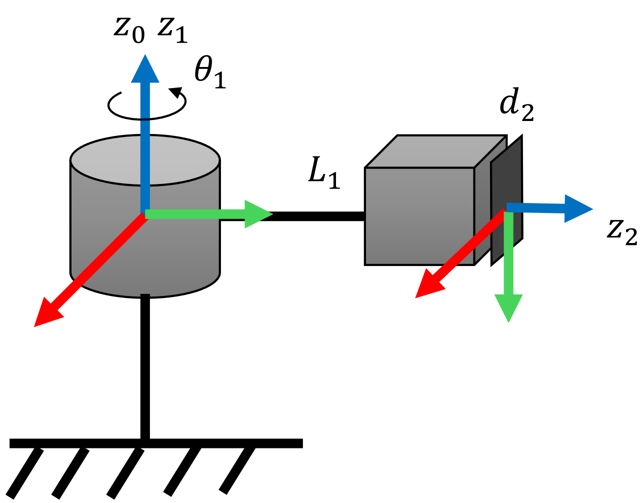
- If z axes intersect or are parallel, there are different ways of assigning origin and x
- (*the goal is to induce as many zeros as possible*)

DH Parameters



	Link length	Link twist	Link offset	Joint angle
i	a_{i-1}	α_{i-1}	d_i	ϕ_i
1	0	0	0	θ_1
2	0	$-\pi/2$	$L_1 + d_2$	0

DH Parameters



	Link length	Link twist	Link offset	Joint angle
i	a_{i-1}	α_{i-1}	d_i	ϕ_i
1	0	0	0	θ_1
2	0	$-\pi/2$	$L_1 + d_2$	0

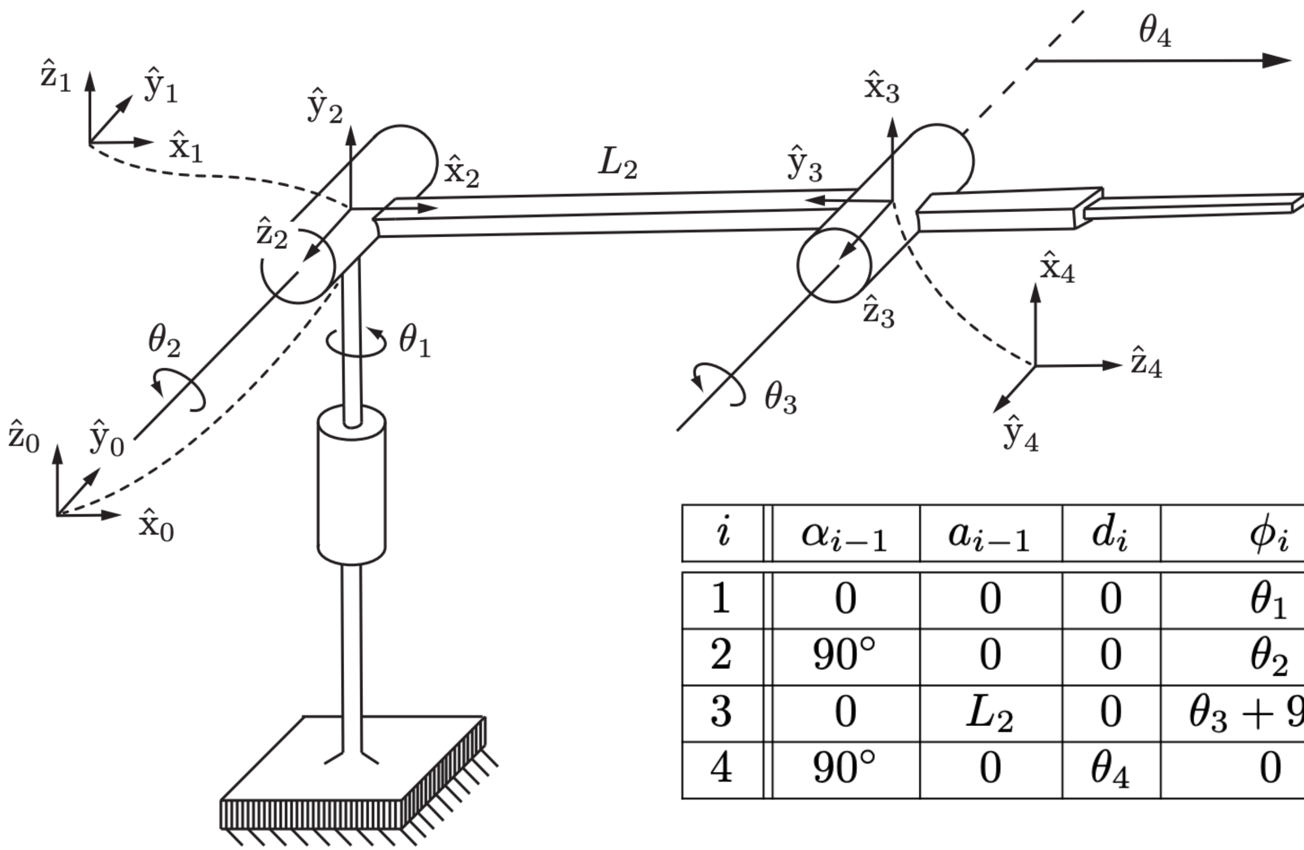
$$T_{i-1,i} = \mathrm{Rot}(\hat{\mathrm{x}}, \alpha_{i-1})\mathrm{Trans}(\hat{\mathrm{x}}, a_{i-1})\mathrm{Trans}(\hat{\mathrm{z}}, d_i)\mathrm{Rot}(\hat{\mathrm{z}}, \phi_i)$$

$$= \begin{bmatrix} \cos\phi_i & -\sin\phi_i & 0 & a_{i-1} \\ \sin\phi_i\cos\alpha_{i-1} & \cos\phi_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -d_i\sin\alpha_{i-1} \\ \sin\phi_i\sin\alpha_{i-1} & \cos\phi_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & d_i\cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{0,2} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & L_1 + d_2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & -(L_1 + d_2)\sin\theta_1 \\ \sin\theta_1 & 0 & \cos\theta_1 & (L_1 + d_2)\cos\theta_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Link twist Link length Link offset Joint angle

$$T_{i-1,i} = \mathrm{Rot}(\hat{\mathrm{x}}, \alpha_{i-1})\mathrm{Trans}(\hat{\mathrm{x}}, a_{i-1})\mathrm{Trans}(\hat{\mathrm{z}}, d_i)\mathrm{Rot}(\hat{\mathrm{z}}, \phi_i)$$



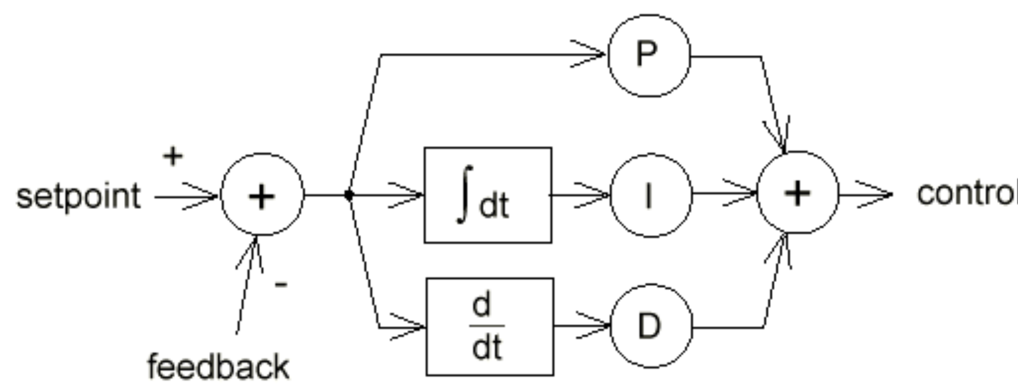
i	α_{i-1}	a_{i-1}	d_i	ϕ_i
1	0	0	0	θ_1
2	$90°$	0	0	θ_2
3	0	L_2	0	$\theta_3 + 90°$
4	$90°$	0	θ_4	0

Recap: PID control

- PID control combines P, I and D control:

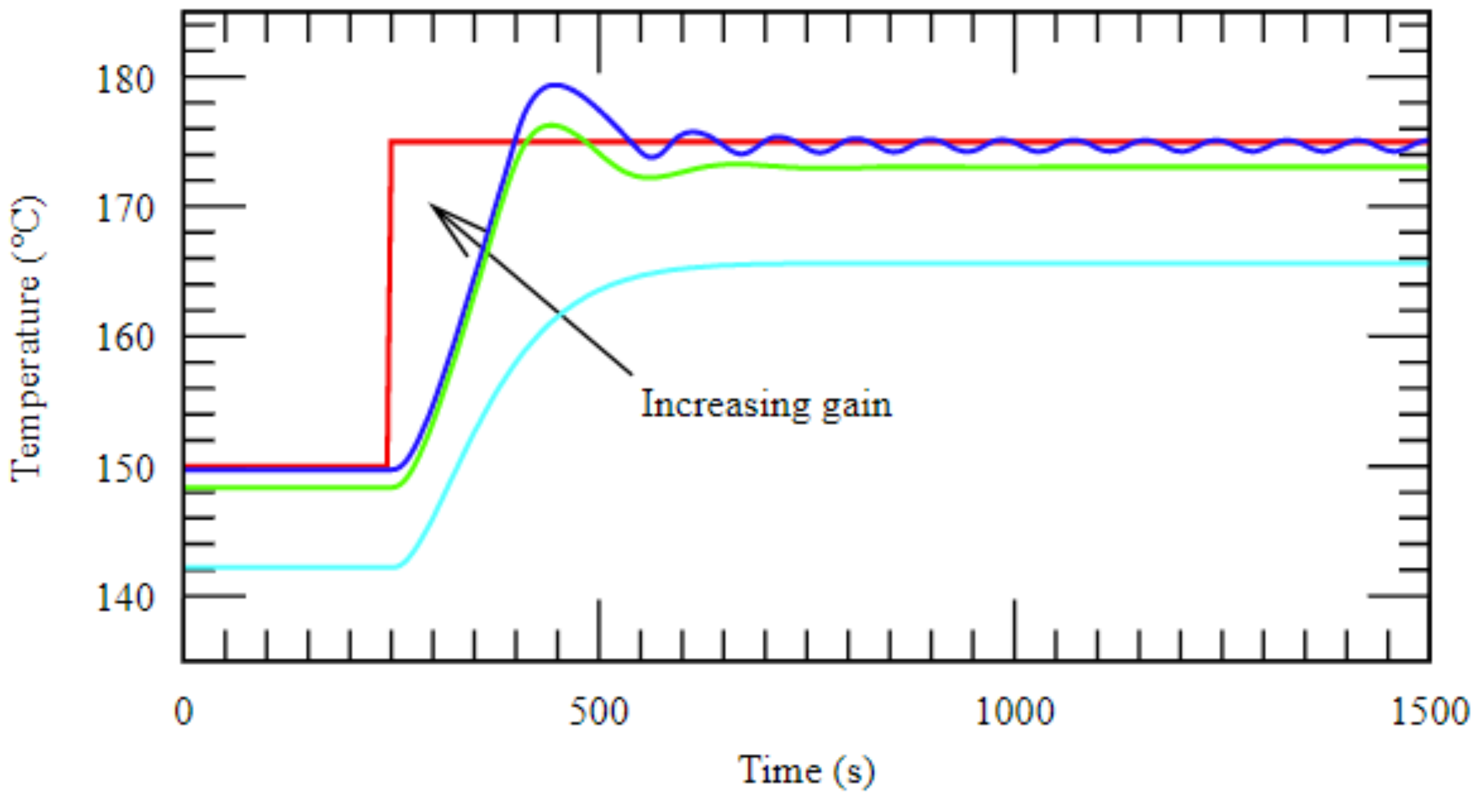
$$o = K_p e + K_i \int e(t)dt + K_d \frac{de}{dt}$$

  - P component minimizes instantaneous error
  - I component minimizes cumulative error
  - D component provides damping



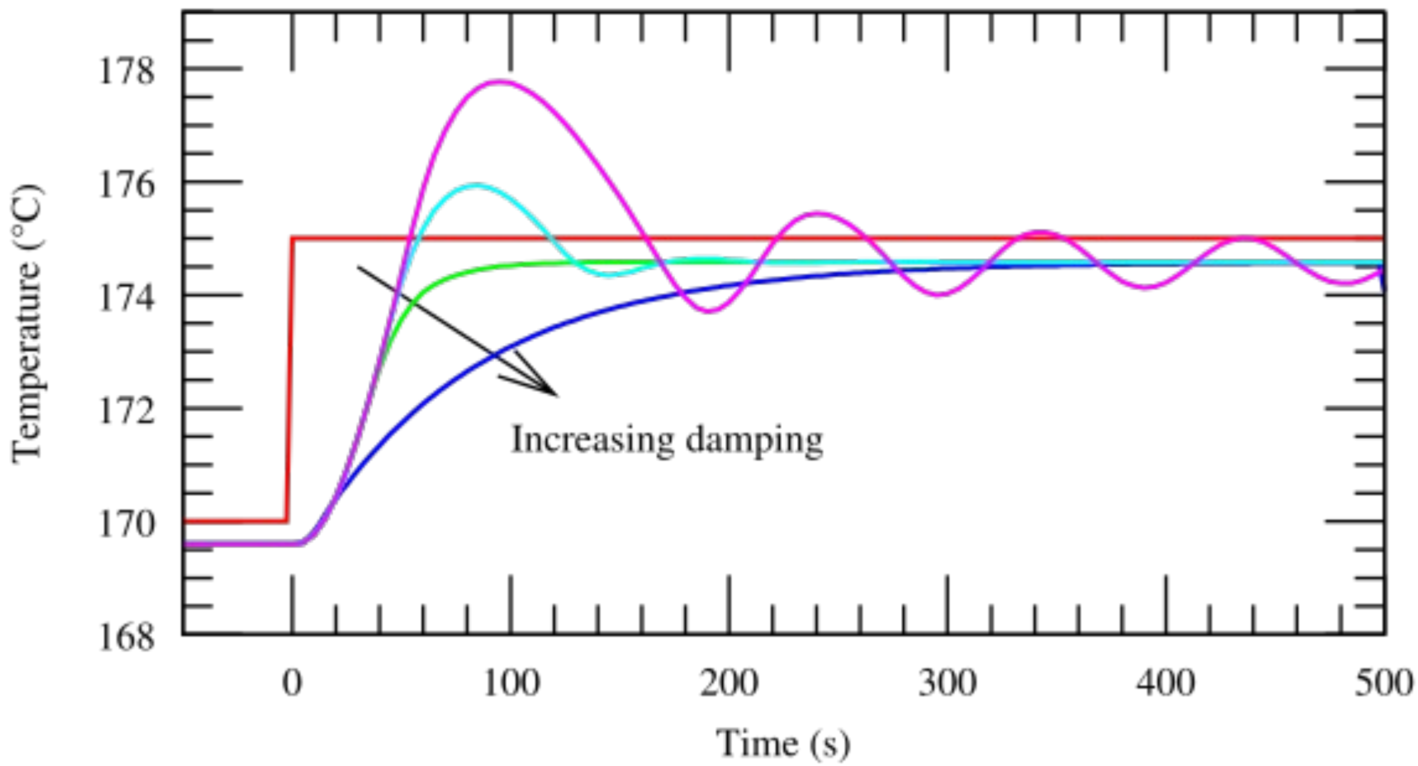
PID control

Varying Kp



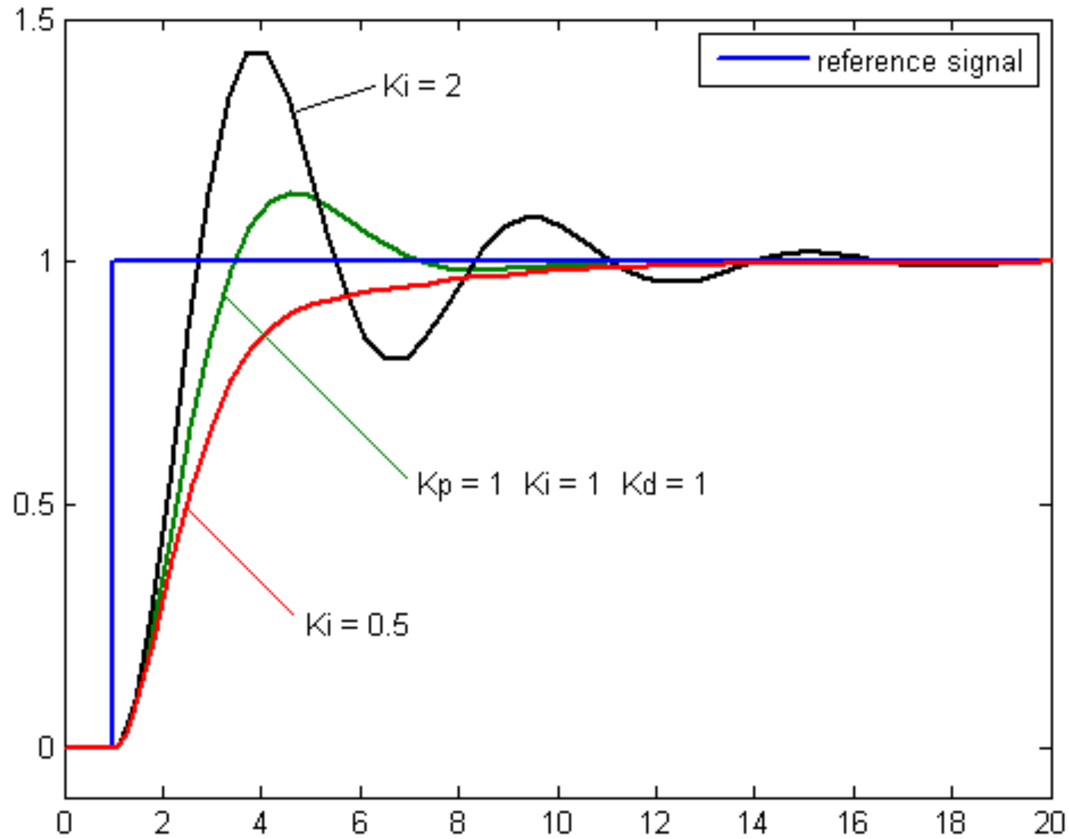
PID control

Varying Kd



PID control

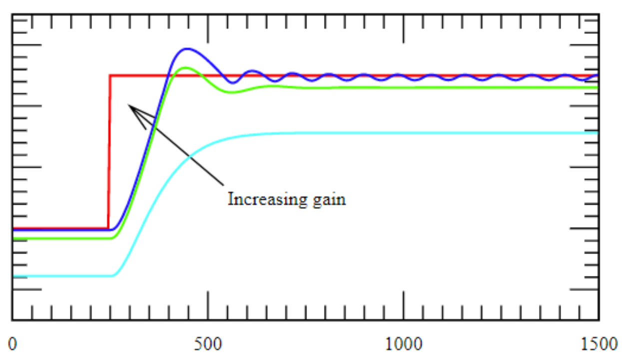
Varying Ki



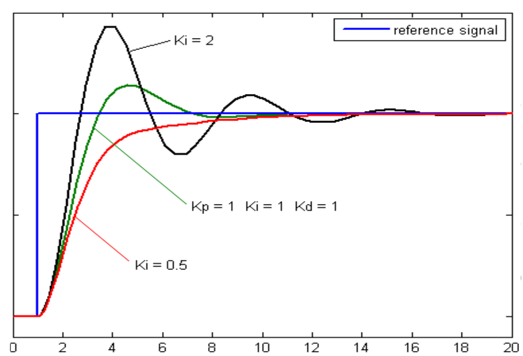
PID control summary

	Rise Time	Overshoot	Settling time
K_p	Decrease	Increase	
K_i	Decrease	Increase	Increase
K_d		Decrease	Decrease

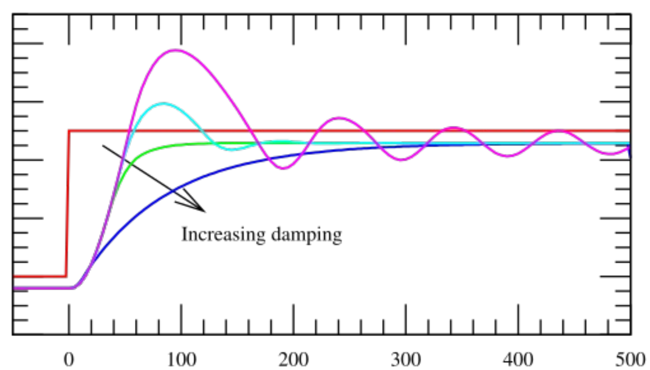
Varying Kp



Varying Ki



Varying Kd



Recap

task space: 2D plane

Forward Kinematics

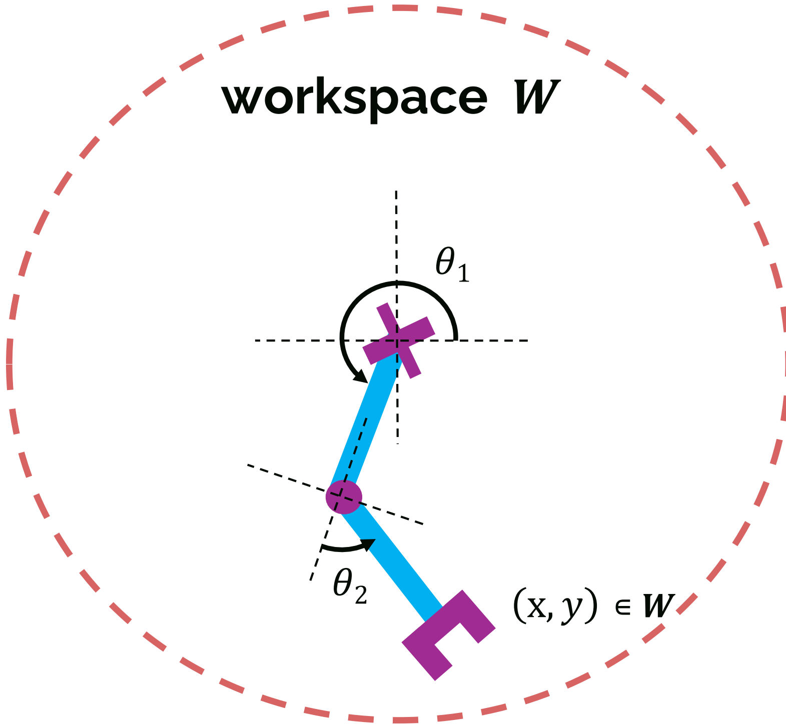
FK: $\boldsymbol{C} \rightarrow \boldsymbol{W}$

$$FK\big((\theta_1, \theta_2)\big) = (\mathrm{x}, y)$$

Inverse Kinematics

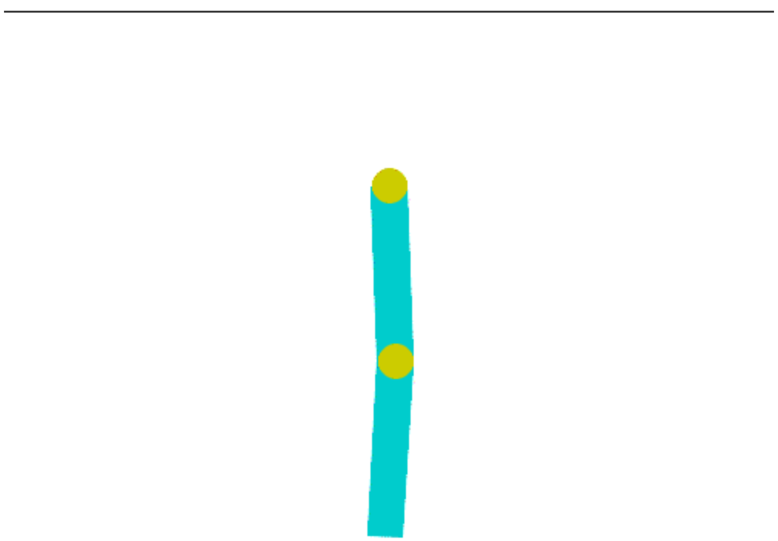
IK: $\boldsymbol{W} \rightarrow \boldsymbol{C}$

$$IK\big((\mathrm{x}, y)\big) = (\theta_1, \theta_2)$$



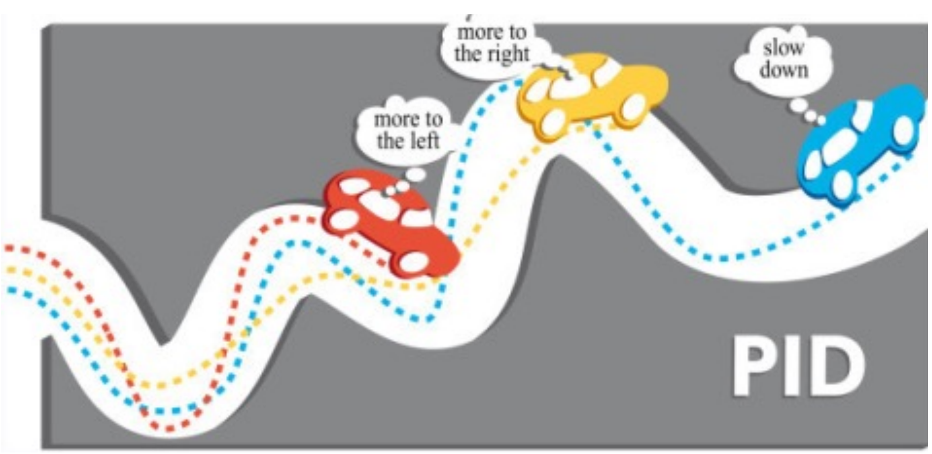
configuration space $\boldsymbol{C}$

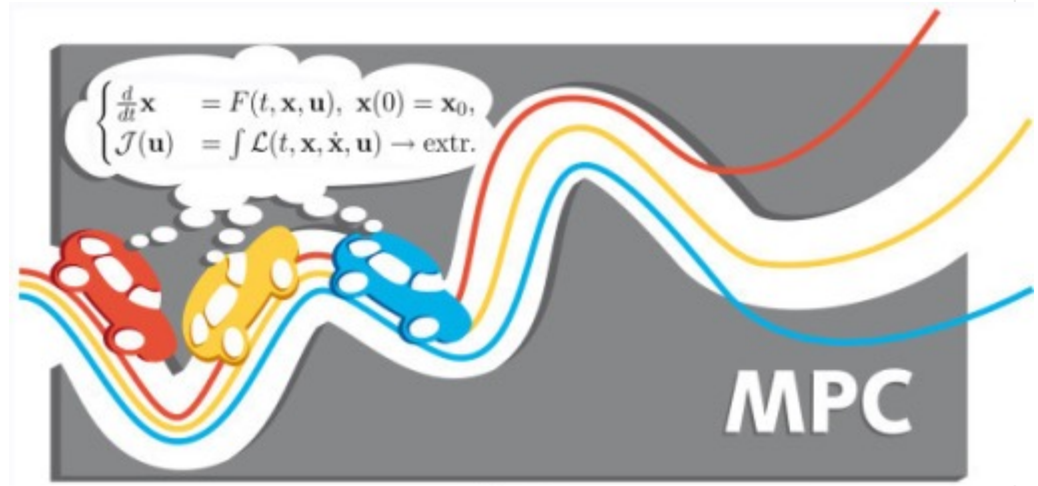
$(\theta_1, \theta_2) \in \boldsymbol{C}$



Model Predictive Control

PID (Reactive Control) vs Model Predictive Control



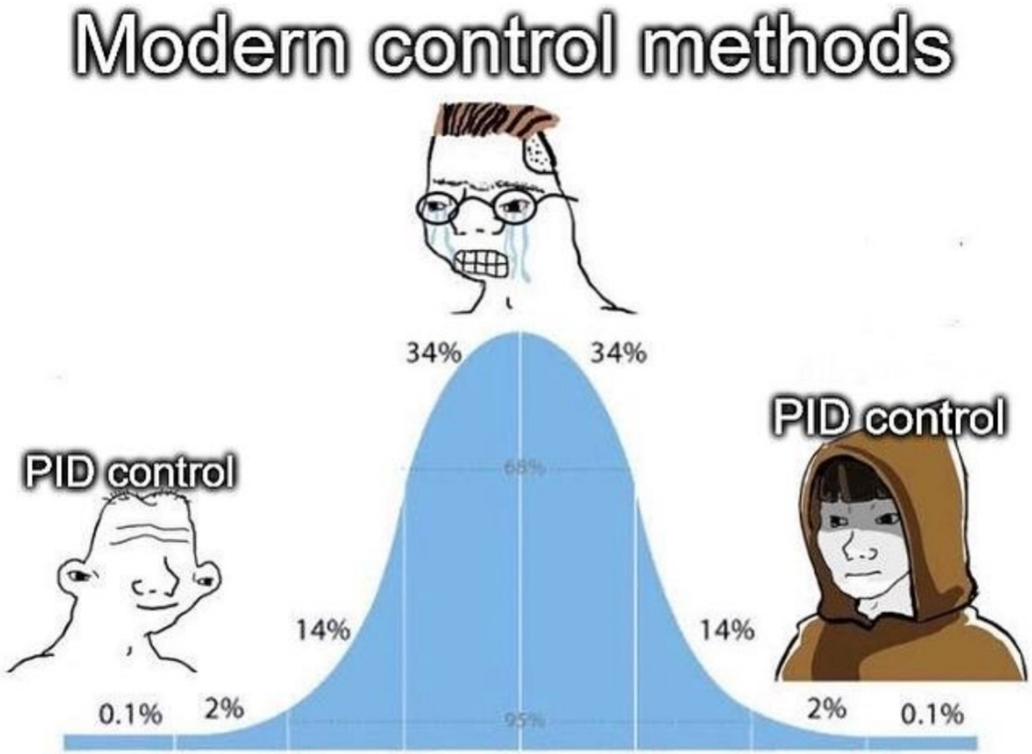


Model predictive control (MPC) is an optimal control technique in which the calculated control actions minimize a **cost function** for a constrained dynamical system over a **finite, receding, horizon**.

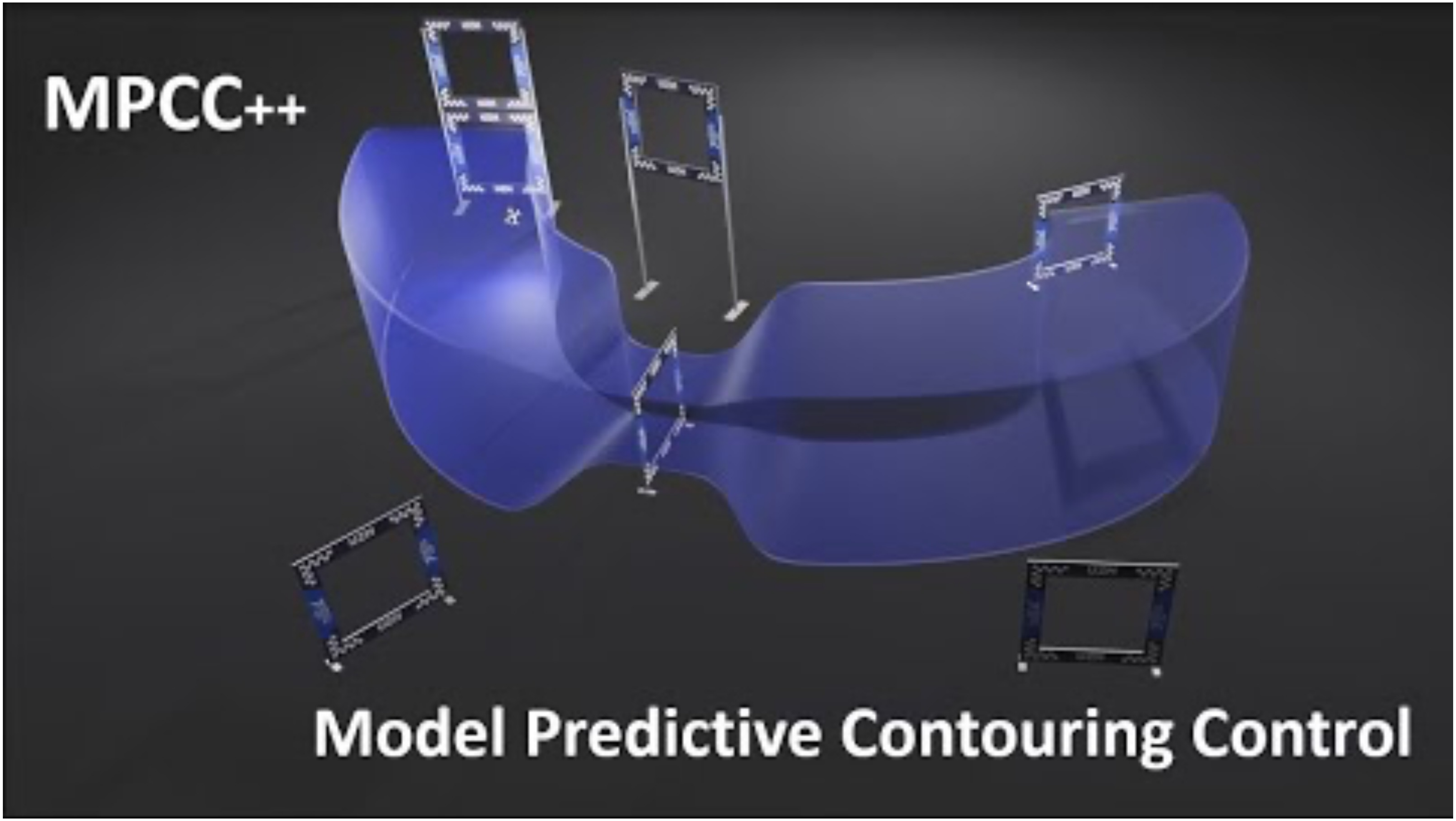
- Predicts future system behavior using a **model**.
- Solves an **optimization** problem at each step.
- Applies only the first control input at each step (iterative).
- Repeats this process continuously (receding horizon).
- Handles input and output **constraints** directly.

System Identification

...building a mathematical model of a dynamic system **from measured data**



"PID is used in 99% of applications and 99% of research efforts go into the remaining 1% that can't be solved with PID"


MPCC++
Model Predictive Contouring Control

Cameras & 2D Perception

Slides Adapted from Stanford EE227/CS227A: Robot Perception

Color camera

- Cameras are the primary sensor for many robotic platforms
- One of the cheapest and richest sensors is a camera
- Many other sensors are build on top of color camera

Single view
RGB image

Stereo

Tactile
(Gelsight)

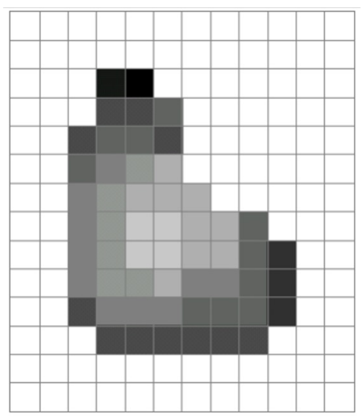
Images Representation

- An image is basically a 2D array of intensity/color values
- Image types:

Color Gray Scale B-W or Binary

Grayscale Images

A grid (2D matrix) of intensity values:



=

X

255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	20	0	255	255	255	255	255	255	255
255	255	255	75	75	75	255	255	255	255	255	255
255	255	75	95	95	75	255	255	255	255	255	255
255	255	96	127	145	175	255	255	255	255	255	255
255	255	127	145	175	175	175	255	255	255	255	255
255	255	127	145	200	200	175	175	95	255	255	255
255	255	127	145	200	200	175	175	95	47	255	255
255	255	127	145	145	175	127	127	95	47	255	255
255	255	74	127	127	127	95	95	95	47	255	255
255	255	255	74	74	74	74	74	74	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255

Y

<u>Pixel:</u> A "picture element" that contains the light intensity at some location *(x,y)* in the image Referred to as *I(x,y)*

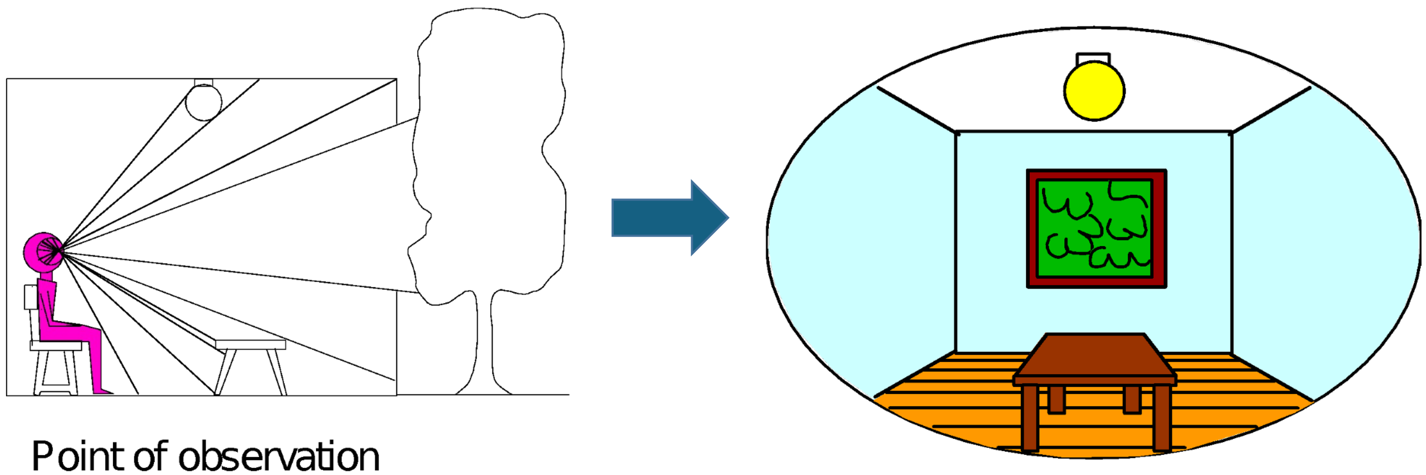
<u>Image Resolution:</u> expressed in terms of Width and Height of the image

Camera & Image formation

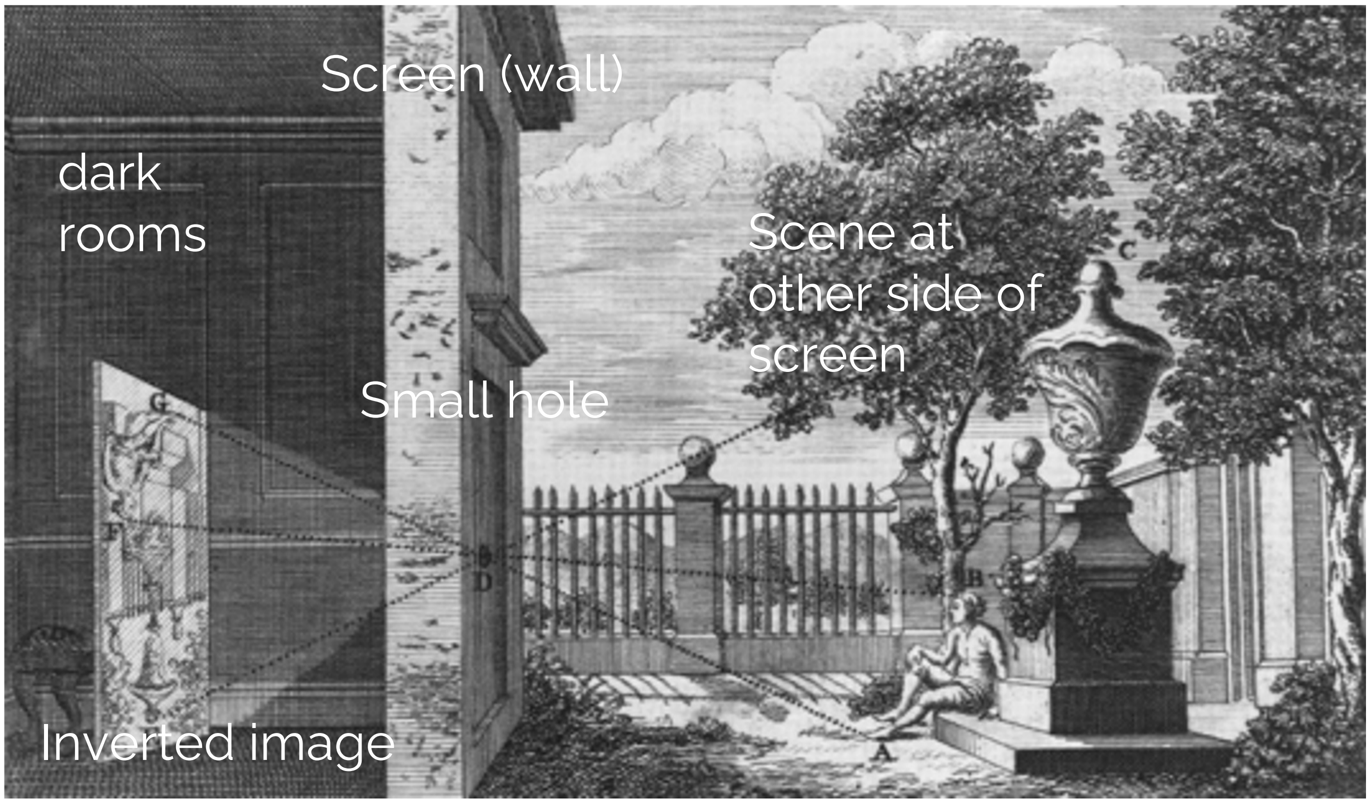
- How we get the image (Image formation)
- Pinhole Camera Model
- 2D computer vision tasks and challenges

Image Formation

3D world *2D image*



Pinhole camera model



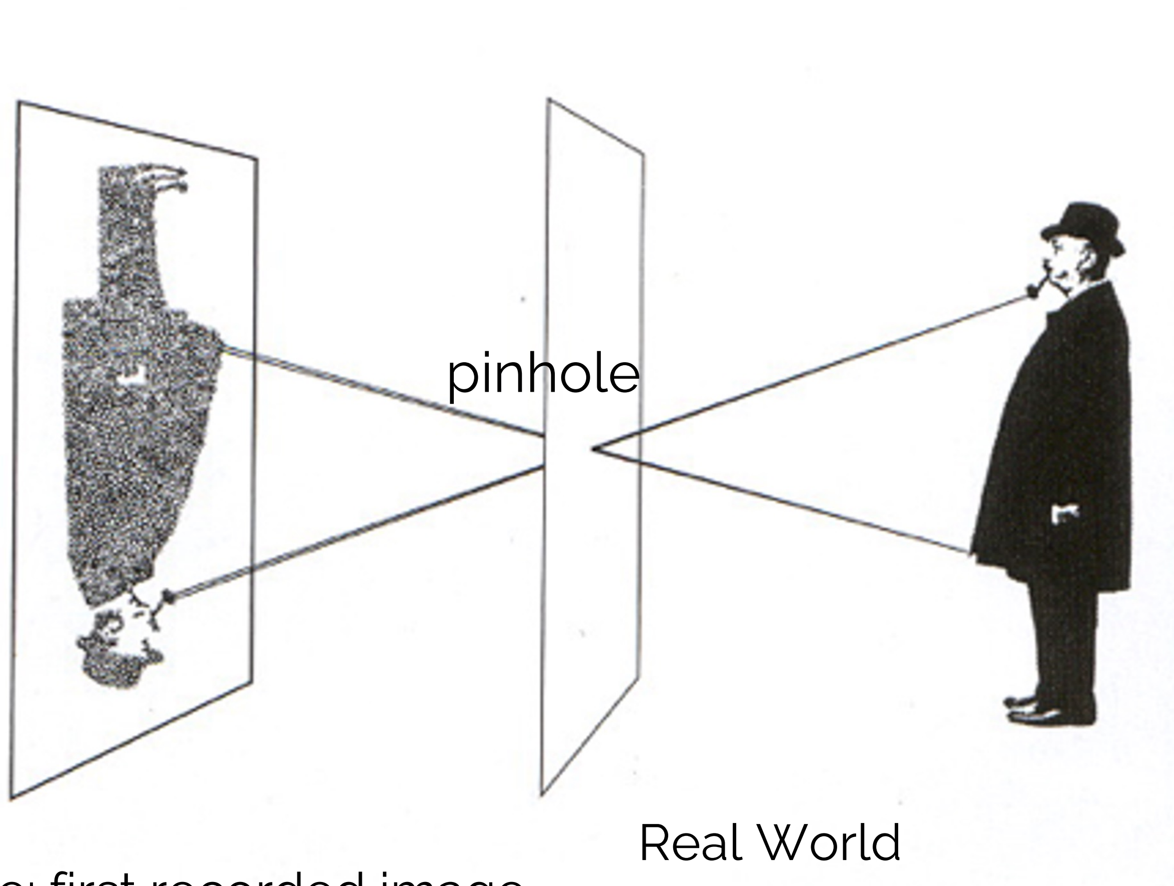
Pinhole image:
Natural phenomenon. Known during classical period in China and Greece (e.g. 470BC to 390BC)

Used for <u>art creation</u> and religious ceremony in ancient times

Expensive to record the image (drawing)

Pinhole camera model

Light sensitive material as film (the local region change color when it expose to light)

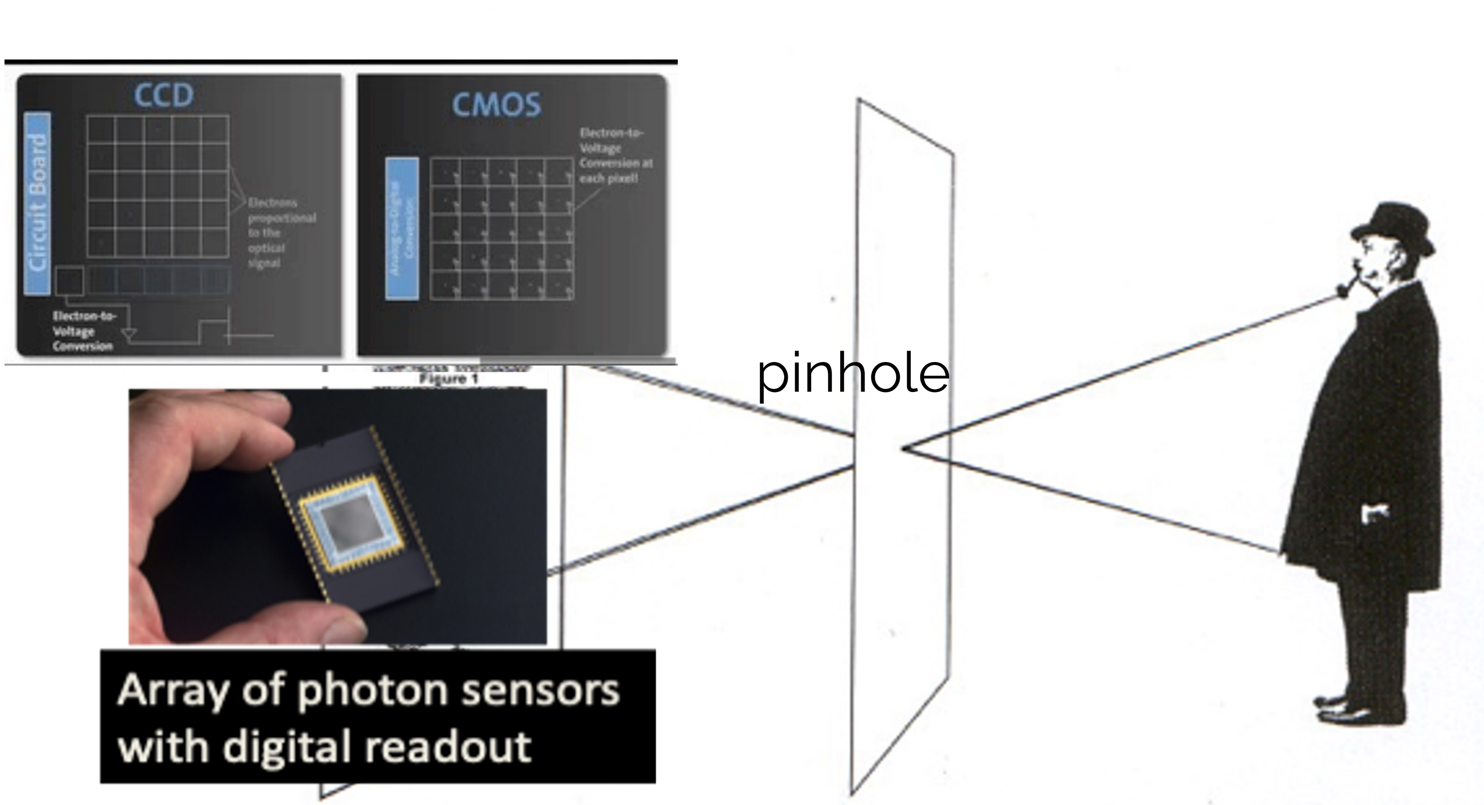


Hard to store, lose color after awhile

Joseph Nicéphore Niépce: first recorded image

Pinhole camera model

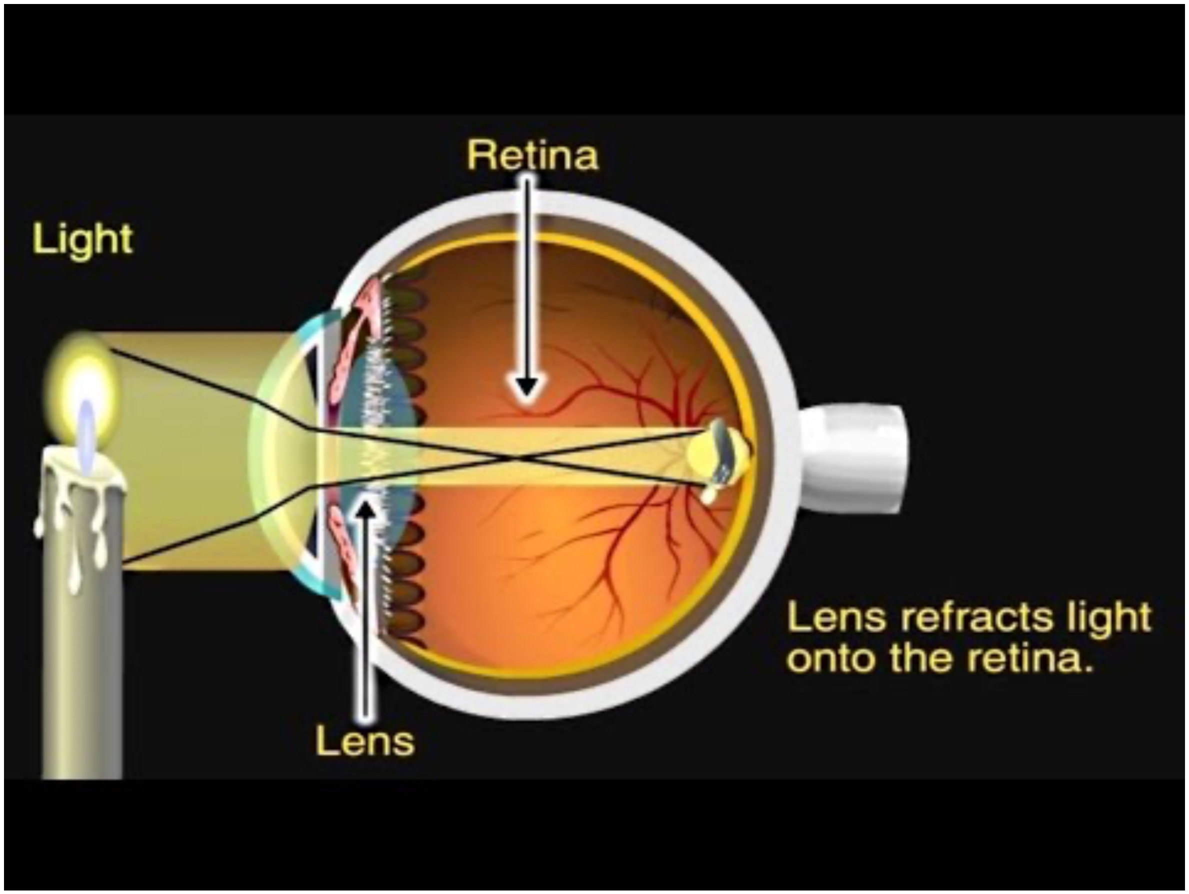
Today:
photon
sensors are
CCD,
CMOS, etc.



Real World

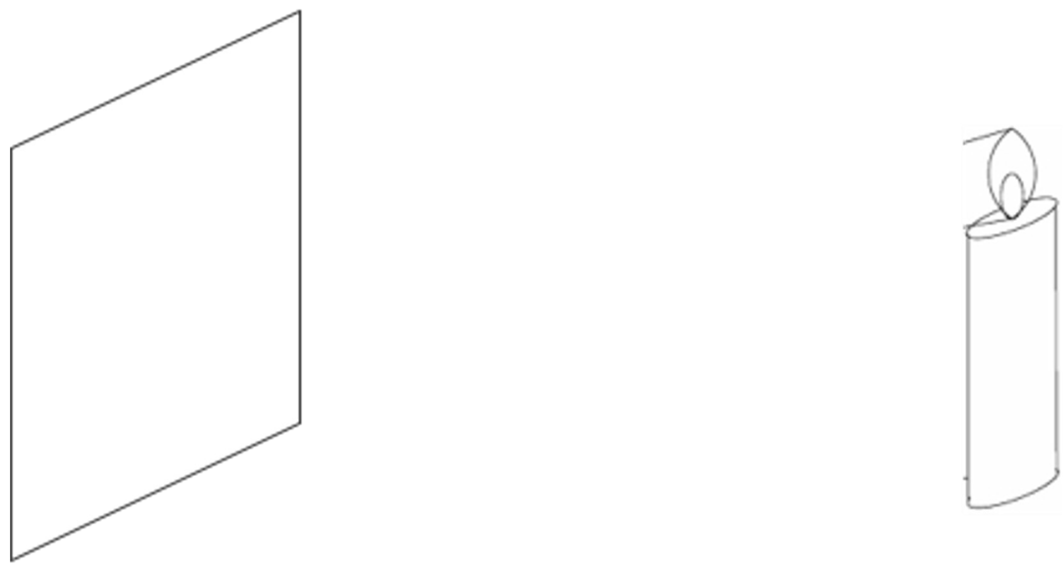
* Willard Boyle and George Smith, the co-inventors of the <u>CCD (Charge-Coupled Device)</u>, were awarded the 2009 Nobel Prize in Physics for their invention.

Human Vision shares similar structure



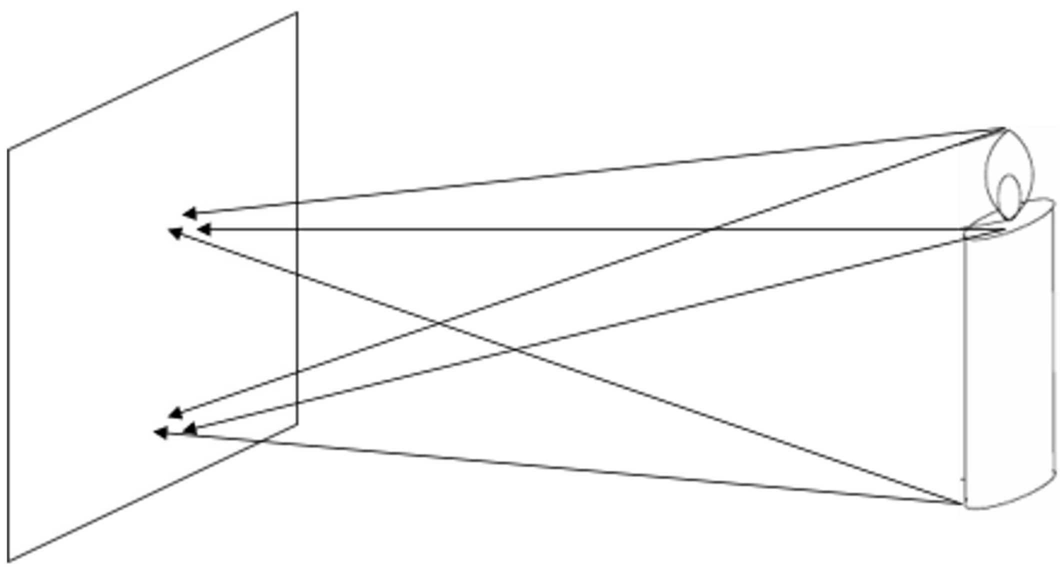
Why we need a pinhole?

Why is there no image on a white piece of paper?



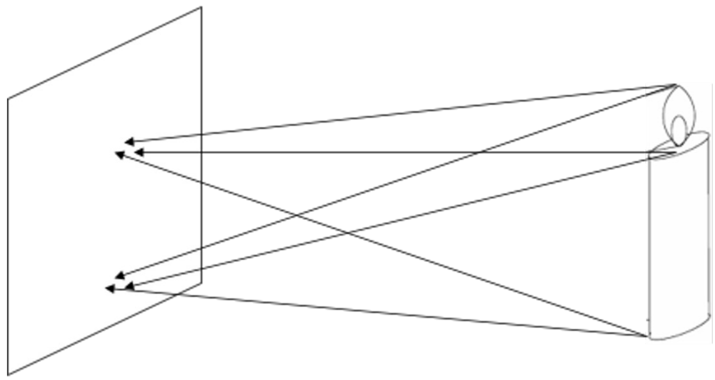
Why we need a pinhole?

Why is there no image on a white piece of paper?

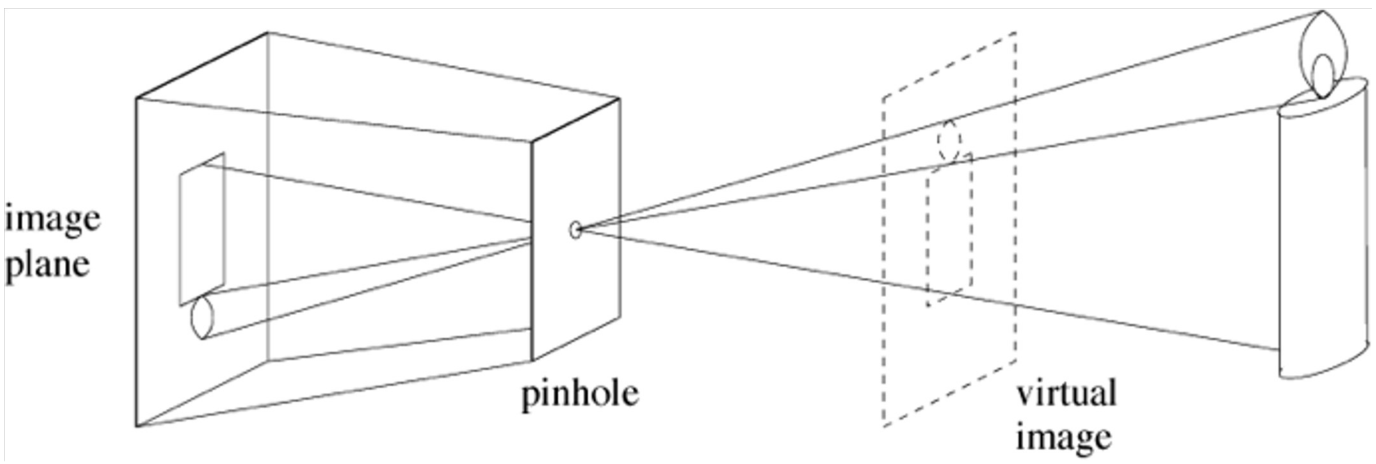


Light rays from many different parts of the scene strike the same point on the paper.

Why we need a pinhole?



Light rays from many different parts of the scene strike the same point on the paper.

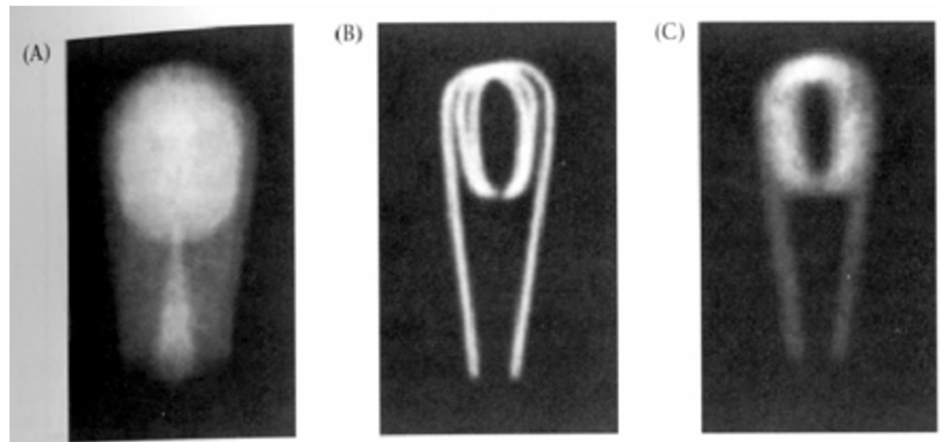


Each point on the image plane sees light from only one direction --- the one that passes through the pinhole.

Problem with Pinhole Camera

The pinhole size:

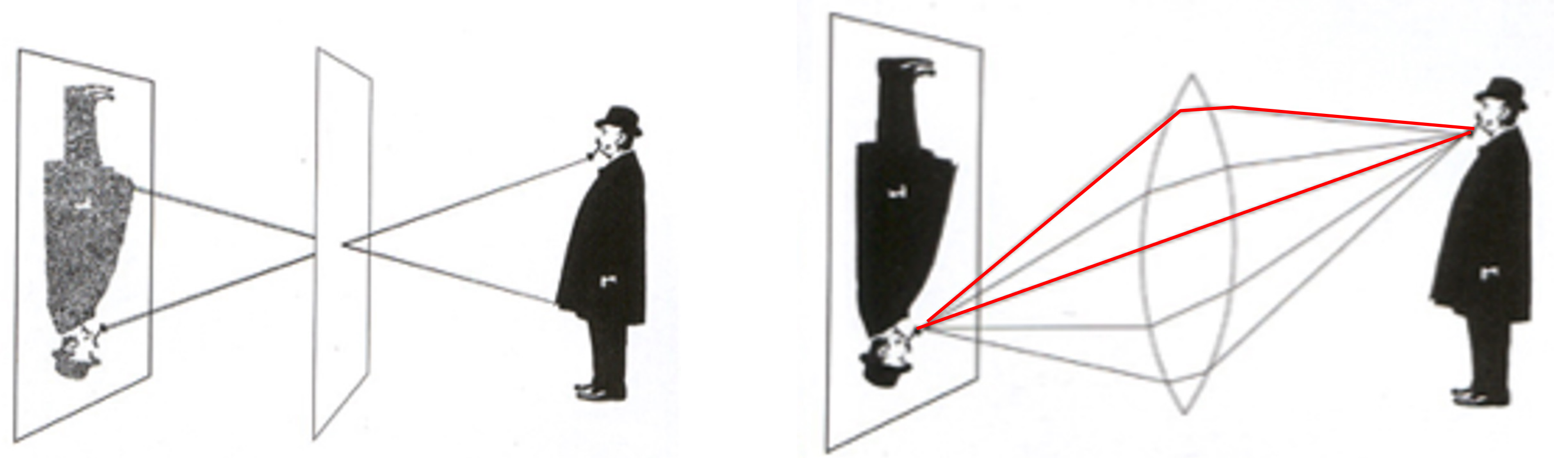
- If large, blurry
- If small, not enough light
- When the pinhole size is extremely small, we will see the diffraction effect through pinhole, result in blurry image



2.18 DIFFRACTION LIMITS THE QUALITY OF PINHOLE OPTICS. These three images of a bulb filament were made using pinholes with decreasing size. (A) When the pinhole is relatively large, the image rays are not properly converged, and the image is blurred. (B) Reducing the size of the pinhole improves the focus. (C) Reducing the size of the pinhole further worsens the focus, due to diffraction. From Ruechardt, 1958.

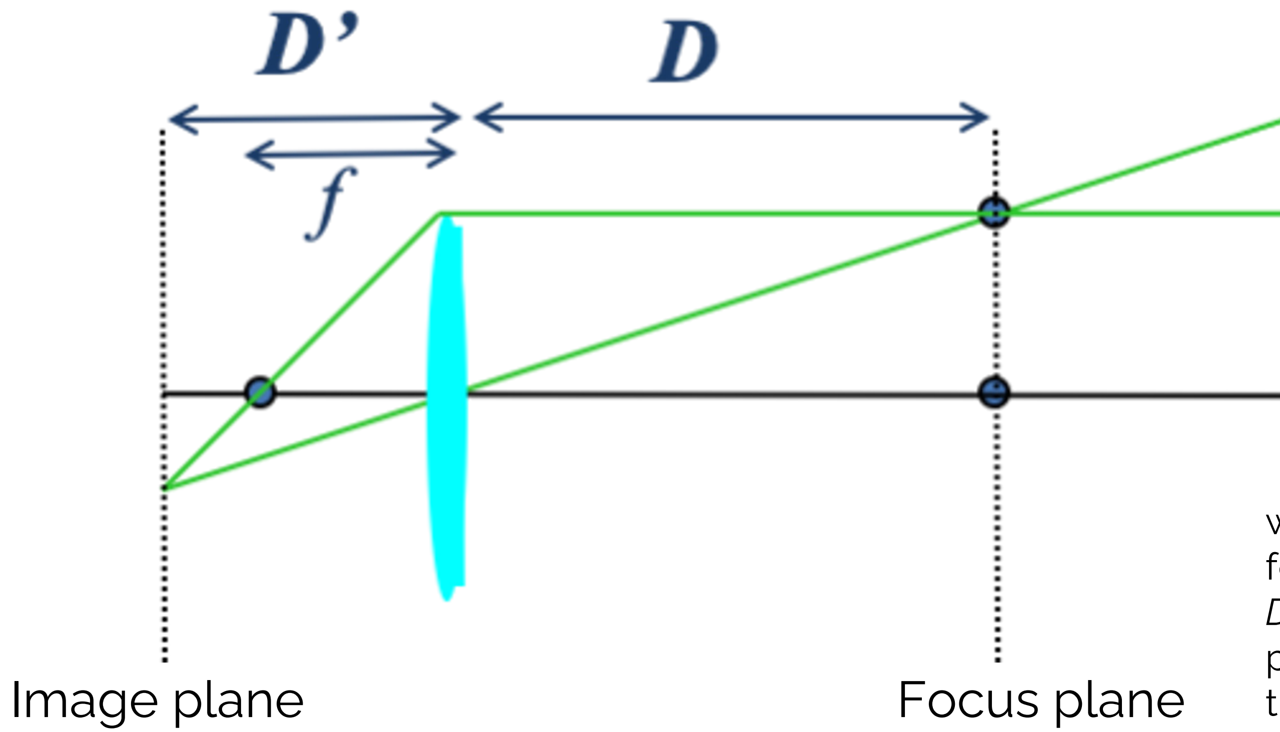
Solution: refraction (lenses)

- Essentially add multiple pinhole images
- Shift them to align using the light **refraction**
- However, this alignment works only for <u>one depth</u> (need the object and image plane stay in focus.)



Lenses issues (depth of field)

Only objects on focus plane are in “perfect” focus

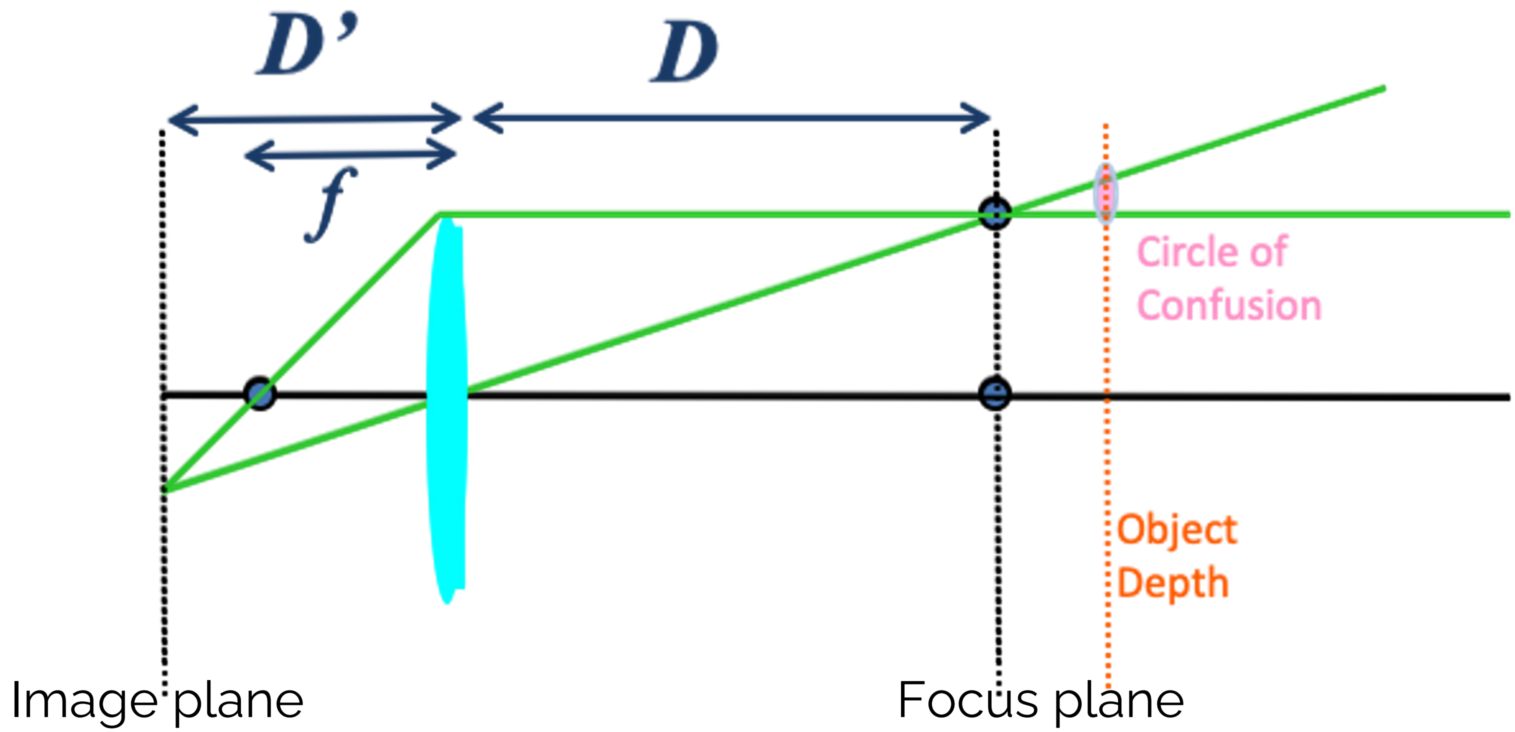


$$\frac{1}{D} + \frac{1}{D'} = \frac{1}{f}$$

where D is the distance of an focus plane to the lens plane D' is the distance of image plane to lens plane, and f is the focal length of the lens

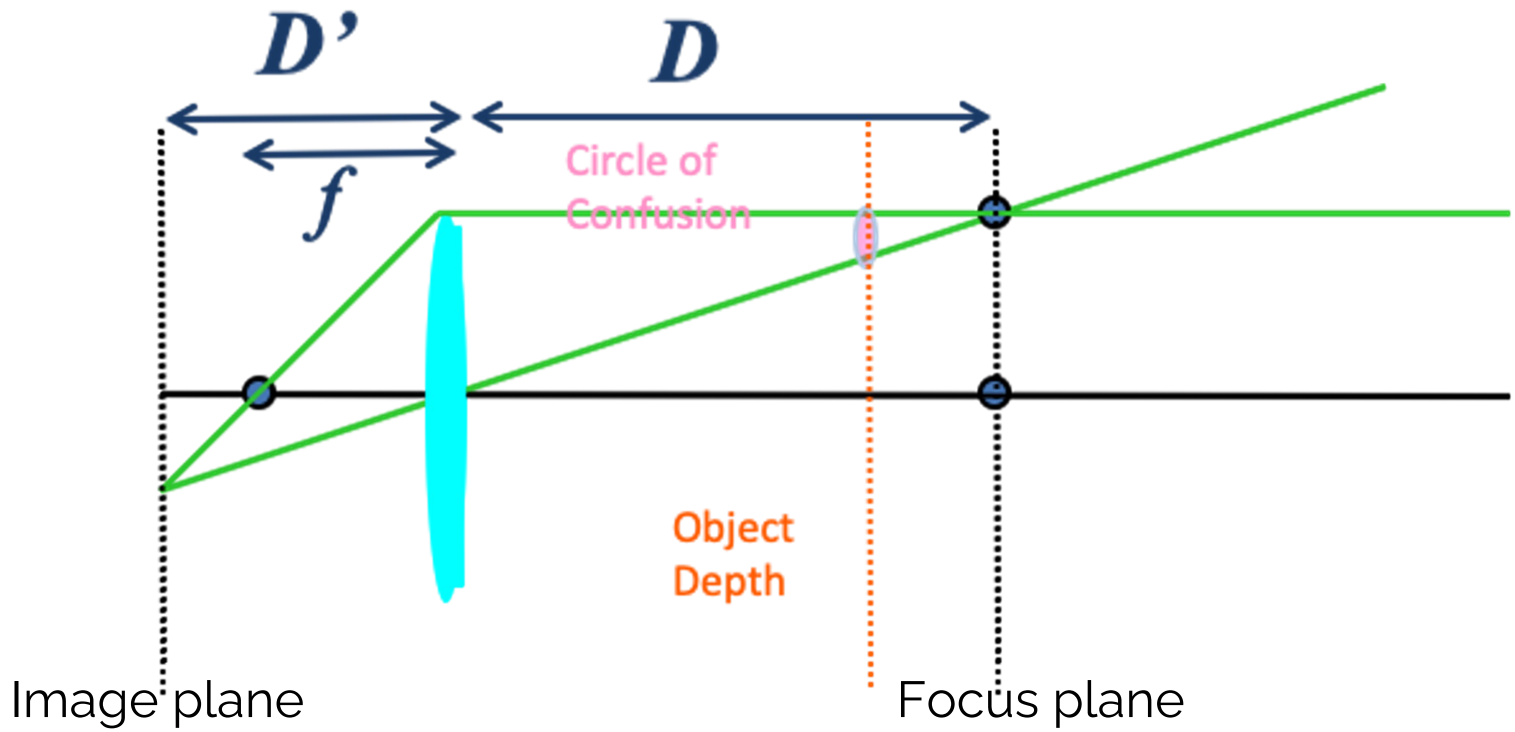
Lenses issues (depth of field)

Only objects on focus plane are in “perfect” focus



Lenses issues (depth of field)

Only objects on focus plane are in “perfect” focus



Lenses issues (depth of field)

Objects close to focus plane are in better focus


That’s it for today!
Questions?