CS 4476: Introduction to Computer Vision

14. Review Lecture

February 26, 2024

Agenda

- Logistics
- Format of the exam
- Section-wise analysis
- Pointers for the exam

Logistics

- Assignment 3 out last week, due March 10.

Assignment 3: Camera Calibration and Fundamental Matrix Estimation with RANSAC

FEB 18, 2024 11:59 PM MAR 10, 2024 11:59 PM
Late Due Date: MAR 15, 2024 11:59 PM

- Today is the last lecture in classical vision section.
- If you have attempted the Canvas exam check-in survey ***need not email TAs and prof.***

Format

- Time : 1 hour (2:10 p.m. to 3.10 p.m.), 28 February 2024 (**next class**)
- Mark Distribution:

Question	Points	Score
Filters & Features	25	
Transformations & Fitting	24	
Hough Transform	15	
Projective Geometry & Camera Calibration	14	
Epipolar Geometry	22	
Total:	100	

Filters and features

a) Different types of filters and their functions

(a) (6 points) Identify the filters below and their functions:

a) $\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ b) $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ c) $\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

Solution:

a) Box Blur - Smoothing the image

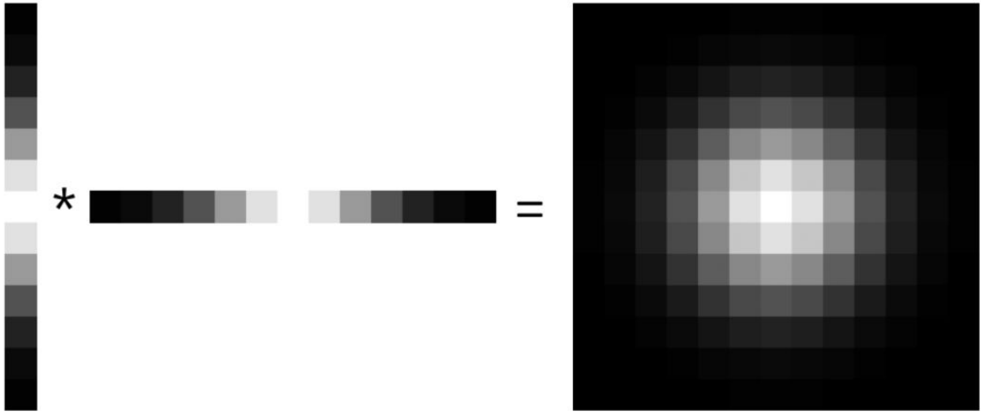
b) Laplacian filter - Identifies regions of rapid intensity change

c) Sobel filter/Horizontal edge detection - detects horizontal edges.

b) Separability of Gaussian Filters

1D Gaussian $*$ 1D Gaussian = 2D Gaussian

Image $*$ 2D Gauss = Image $*$ (1D Gauss $*$ 1D Gauss)
= (Image $*$ 1D Gauss) $*$ 1D Gauss



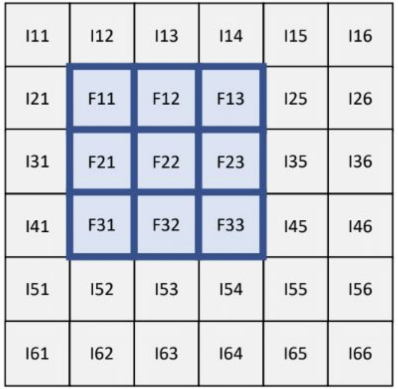
53
e credit: David Fouhey

Slide Credit: David Fouhey

Gaussian Filter

Runtime Complexity

Image size = NxN = 6x6
Filter size = MxM = 3x3




```
for ImageY in range(N):
        for ImageX in range(N):
                for FilterY in range(M):
                        for FilterX in range(M):
                                ...
```

Time: $O(N^2M^2)$

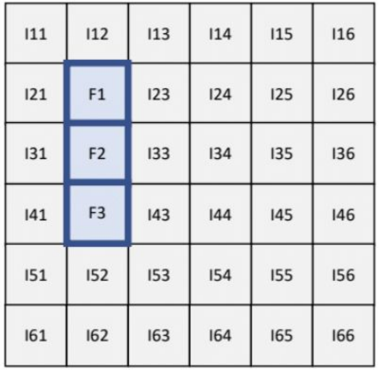
Slide Credit: David Fouhey

Separable Gaussian Filter

Runtime Complexity

Image size = NxN = 6x6

Filter size = Mx1 = 3x1




```
for ImageY in range(N):
        for ImageX in range(N):
                for FilterY in range(M):
                        ...
for ImageY in range(N):
        for ImageX in range(N):
                for FilterX in range(M):
                        ...
```

Time: $O(N^2M)$

le credit: David Fouhey

Slide Credit: David Fouhey

(b) (4 points) Write the time complexity of a Gaussian filter, and a Separable Gaussian filter, in big O notation.

i. Guassian filter ______ $O(N^2M^2)$ ______

ii. Seperable Guassian filter ______ $O(N^2M)$ ______

Which one of them exhibits a lower time complexity and why?

Solution: The separable Gaussian filter achieves a lower time complexity by decomposing the 2D convolution into two 1D convolutions, taking advantage of the separability of the Gaussian kernel.

c) Image Gradients

(c) (5 points) For an image, I, consider the following responses after filtering with an x and y derivative (difference) filter.

$$\frac{dI}{dx} = \begin{array}{|c|c|}\hline 1 & 4 \\ \hline 4 & 2 \\ \hline\end{array} \qquad \frac{dI}{dy} = \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & 2 \\ \hline\end{array}$$

i. Compute the corresponding gradient magnitude image:
(Answers may include square roots.)

$$\sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Solution: $\begin{array}{|c|c|}\hline \sqrt{2} & 4 \\ \hline 4 & \sqrt{8} \\ \hline\end{array}$

Thresholding

- Choose a threshold value t
- Set any pixels less than t to zero (off)
- Set any pixels greater than or equal to t to one (on)

$$v = \begin{cases} 0; & p < t \\ 1; & p \geq t \end{cases}$$

ii. Write the corresponding binary edge image for a threshold of 3.

$$\begin{array}{|c|c|} \hline \sqrt{2} & 4 \\ \hline 4 & \sqrt{8} \\ \hline \end{array} \xrightarrow[\text{3}]{\text{Threshold}} \begin{array}{|c|c|} \hline 0 & 1 \\ \hline 1 & 0 \\ \hline \end{array}$$

d) Harris Corner Detection

(e) (6 points) Below are three True/False question related to Harris Corner Detection. λ_1 and λ_2 are eigenvalues of the matrix M for a region. Write a brief explanation if the statement is False.

i. The region is an edge if $\lambda_1 >> \lambda_2$

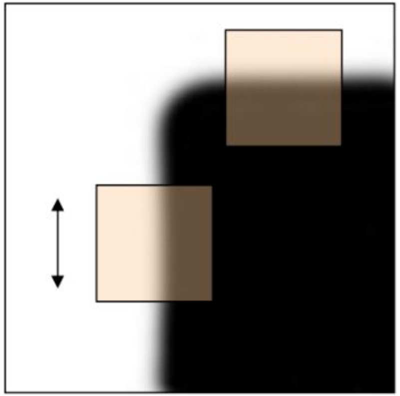
Solution: True.

ii. The region is a corner if λ_1 and λ_2 are small values.

Solution: False. λ_1 and λ_2 should be large values.

iii. The region is flat if $\lambda_2 >> \lambda_1$

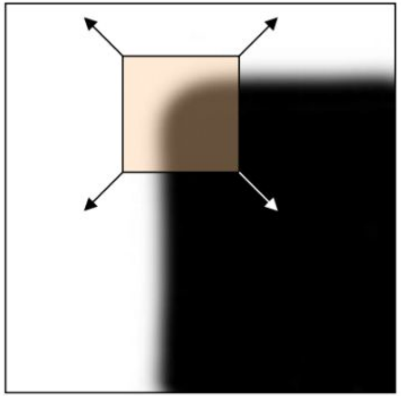
Solution: False. λ_1 and λ_2 should be small values.



"edge":

$\lambda_1 >> \lambda_2$

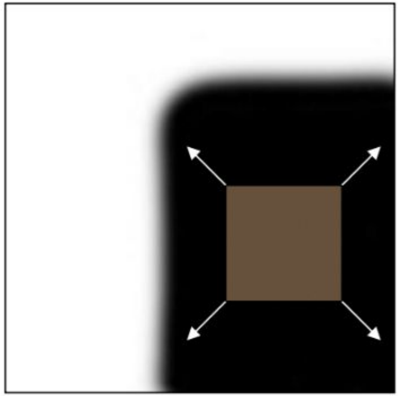
$\lambda_2 >> \lambda_1$



"corner":

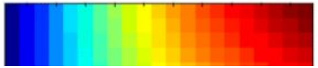
λ_1 and λ_2 are large,

$\lambda_1 \sim \lambda_2$;



"flat" region

λ_1 and λ_2 are small;



The conditions for λ_1, λ_2 in the previous slide are deduced based on the corner response function R below:

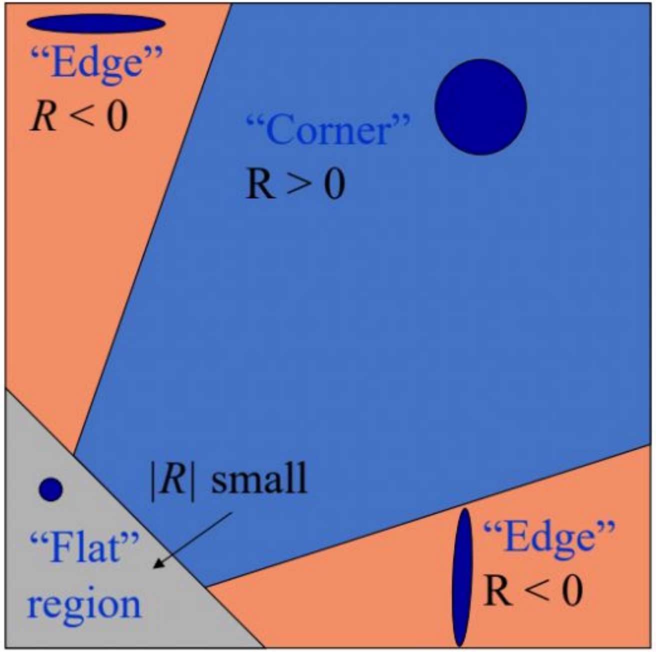
Choosing f

$$R = \det(\boldsymbol{M}) - \alpha\, trace(\boldsymbol{M})^2$$
$$= \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

α: constant (0.04 to 0.06)

Calculating eigenvalues is usually slow!

Determinant, trace have closed form solution



e) Feature Descriptors

iv. (4 points) You are given an interest point detector that can choose regions of interest at different scales and rotations. Now you need to select an algorithm to describe these regions. You have both RGB pixel values and per-pixel gradient angles, re-oriented by the direction of max gradient. You consider the following:

- **A**: RGB pixel values concatenated into a vector
- **B**: A normalized histogram of RGB pixel values
- **C**: Re-oriented gradient angles concatenated into a vector
- **D**: A normalized histogram of re-oriented gradient angles

To help you make your choice, select all features that satisfy the following properties.

i. Invariant to scaling

☐ A ■ B ☐ C ■ D

ii. Invariant to rotations

☐ A ■ B ☐ C ■ D

iii. Retains local geometric information

■ A ☐ B ■ C ☐ D

iv. Invariant to photometric variations

☐ A ☐ B ■ C ■ D

i) Invariant to Scaling:

Histograms represent overall distribution of pixel values, which do not change on scaling. However, A and C will change as the image is scaled.

ii) Invariant to Rotation:

Same reasoning as part i)

iii) Retains local geometric information:

Histograms provide overall distributions and thus lose the spatial/structural layout. Thus, A and C are correct.

i*v) Invariant to photometric variations:*

Changes in lighting alter the pixel values, thus affecting both A and B. Features C and D operate on changes in intensity rather than absolute pixel values and thus are less affected by photometric variations.

Transformations

2: Transformations & Fitting (25 points)

(a) (3 points) Which of the following are true for RANSAC? *(select all that apply)*

- ■ It employs an iterative algorithm.
- ■ Hyerparameter tuning is necessary.
- □ It is extremely sensitive to outliers.
- □ It will converge even when given a uniform random set of points.

Correct Answer: a, b

a. RANSAC involves repeatedly selecting a random subset of the data to fit the model.
b. RANSAC is sensitive to hyperparameters due to the stochastic nature of the fitting algorithm.

RANSAC - Iterative algorithm for affine transformation fitting

Motivation

Reduce the sensitivity towards outliers, as opposed to L1 or L2 norm fitting, which are sensitive to outliers.

Method

- Choose a random subset of dataset to fit the line to.
- Count how many points from the dataset lie in the “neighborhood” of this line.
- If these inliers are greater than a threshold, then select the line.

Slide credits: Lana Lazebnik

(b) (3 points) Construct a transformation matrix that results in a 2D rotation by $3\pi/4$ radians around (0,0). *Note:* $cos(\pi/4) = 1/\sqrt{2}$, $sin(\pi/4) = 1/\sqrt{2}$

Solution:

$$\begin{bmatrix} -1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

2D rotation around (0,0)
x' = x cos(θ) − y sin(θ)
y' = x sin(θ) + y cos(θ)

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

(c) (3 points) Construct a transformation matrix that results in a 2D shear by $\alpha = 6$ in the horizontal direction and by $\beta = 3$ in the vertical direction.

Solution:

$$\begin{bmatrix} 1 & 6 \\ 3 & 1 \end{bmatrix}$$

2D shear
x’ = x + αy
y’ = βx + y

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 6 \\ 3 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

1 * x	α * y
β * x	1 * y

(d) (2 points) Which of these 2D affine transformations can be represented by a single 2x2 matrix multiplication to the input vector: *(select all that apply)*

- ■ Shearing transformations
- ■ Rotation transformations
- □ Translation transformations
- ■ Scaling transformations

Correct Answer: a, c ,d

Affine transformations are combination of linear transformations and translations. However, the translation needs addition of an offset, and cannot be modeled by matmul. Thus only shearing, rotation and scaling.

2D Affine transformations

2D Scaling?

$$x' = s_x * x$$

$$y' = s_y * y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D Rotate around (0,0)?

$$x' = \cos\Theta * x - \sin\Theta * y$$
$$y' = \sin\Theta * x + \cos\Theta * y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

2D Shear?

$$x' = x + sh_x * y$$

$$y' = sh_y * x + y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & sh_x \\ sh_y & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Slide credits: Alyosha Efros

2D Affine transformations with homogeneous coordinates

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_x & 0 & 0 \\ 0 & \boldsymbol{s}_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

Scale

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Rotate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{sh}_x & 0 \\ \boldsymbol{sh}_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

Shear

Slide credits: Alyosha Efros

(e) (3 points) The following matrix represents a transformation. What is the transformation?

$$\begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$$

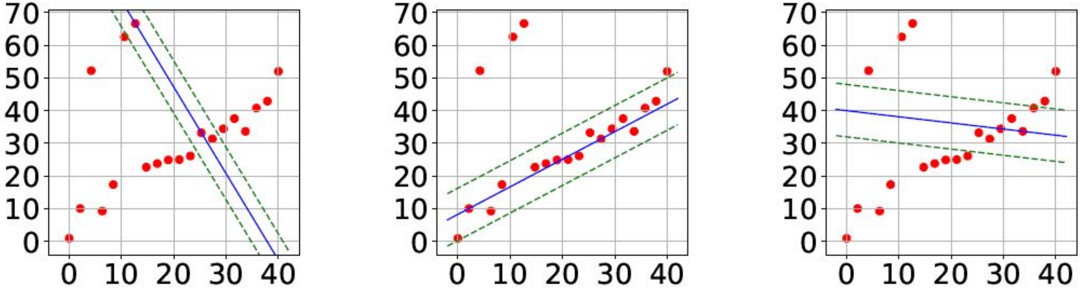
- ☐ Projection on $y = x$.
- ■ Reflection across $y = -x$.
- ☐ Rotation across $y = -x$.
- ☐ Rotation $y = x$.

Correct Answer: b

The x’ = -y and y’ = -x, which indicates the reflection across y = −x

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -y \\ -x \end{pmatrix}$$

Assume that we run RANSAC for 3 iterations as pictured below. The blue solid line represents the line we have fit, while green dotted lines represent the threshold for inliers (any points within or on the green dotted lines are inliers).



(a) iteration1 (b) iteration2 (c) iteration3

(f) (6 points) For each run of the algorithm, compute the number of inliers and indicate whether the model is stored or not.
You can assume that the hyperparameter for the minimum number of inliers is $d = 5$

Solution:

Iteration	Num of Inliers	Model Stored? (yes/no)
1	4	no
2	16	yes
3	6	no

Iteration	Number of Inliers Model	Stored as Current Best? (yes/no)
1	4 (4 points inside green lines)	No (4 < d)
2	16 (16 points inside green lines)	Yes (16 > d)
3	6 (6 points inside green lines)	No (6 > d, 6 < 16)

(g) (1 point) Which of the three iterations will result in the model selected by RANSAC?

☐ Iteration 1   ■ Iteration 2   ☐ Iteration 3

Correct Answer: b

- Iteration 1:
  - 4 inliers, less than d.
- Iteration 2:
  - 16 inliers, greater than d, stored as current best model.
- Iteration 3:
  - 6 inliers, greater than d, less than 16 (current best)

RANSAC for line fitting

Repeat ***N*** times:

- Draw ***s*** points uniformly at random
- Fit line to these ***s*** points
- Find inliers to this line among the remaining points
  - Inlier = points whose distance from the line is less than ***t***
- If there are ***d*** or more inliers, accept the line and refit using all inliers

Slide credits: Lana Lazebnik

(h) (2 points) In which situation would you prefer to use RANSAC over K-means? *(Select one)*

- ☐ The data has very few outliers.
- ■ The data has a lot of outliers.

Correct Answer: b

RANSAC identifies and uses inliers to fit the models, which makes it ideal for datasets with many outliers.

K-means is sensitive to outliers because an outlier can shift the mean of a cluster, then affecting the location of the cluster centroid and lead to incorrect clustering.

Note: Both methods will fail if outliers >> inliers (basically fitting to noise)

RANSAC - Pros and cons

Pros

- Simple, intuitive.
- Often works well.

Cons

- Many hyperparameters to tune.
- Won’t work well for data which has a lot of noise.
- A good initialization is not guaranteed.

Slide credits: Lana Lazebnik

(i) (1 point) After an affine transformation *(Select one)*

- ☐ Parallel lines become intersecting lines.
- ■ Parallel lines remain parallel.

Correct Answer: b

Affine transformations are combinations of linear transformations and translations, both of which preserve parallelism.

Hough transform

Hough Transform

3: Hough Transform (15 points)

Given the following (x, y) coordinates, what are their possible (d, θ) coordinates after being transformed into Hough Space? Recall that when using the Hough Transform, we consider lines in polar coordinates of the form:

$$d = x\cos(\theta) + y\sin(\theta)$$

(a) (2 points) $x = 0, y = 0$

■ $(d, \theta) = (0{,}0)$ ■ $(d, \theta) = (0, \frac{-8\pi}{33})$ ☐ $(d, \theta) = (2{,}0)$

(b) (2 points) $x = 2, y = -1$

☐ $(d, \theta) = (2{,}1)$ ■ $(d, \theta) = (\frac{\sqrt{2}}{2}, \frac{\pi}{4})$ ☐ $(d, \theta) = (0, \pi/6)$

(c) (2 points) $x = -1, y = -2$

■ $(d, \theta) = (\frac{\sqrt{2}}{2}, \frac{-\pi}{4})$ ■ $(d, \theta) = (-1{,}0)$ ☐ $(d, \theta) = (-\frac{1}{2}, \frac{\pi}{2})$

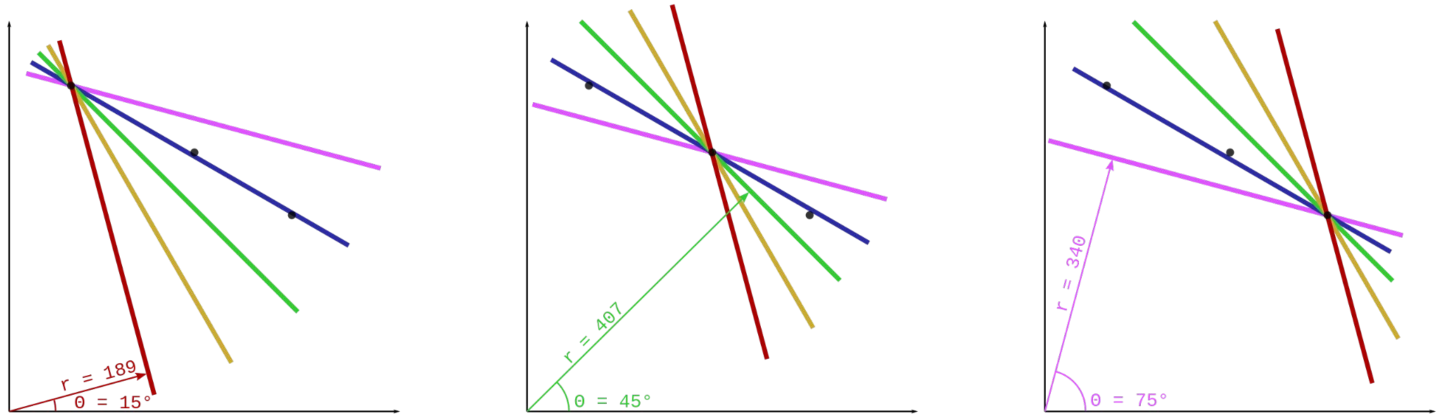
Explanation: The above highlighted options satisfy the equation of d in terms of x,y,theta.

(a) x=0, y=0 results in d=0, and satisfies equation for any theta (0, -8pi/33)

(b) x=2, y=(-1) substituting results in 2cos(theta) + (-1)sin(theta), which satisfies for theta=pi/4 which is d=1/sqrt(2)

(c) x=(-1), y=(-2) substituting results in d=(-1)cos(theta) + (-2)sin(theta), which satisfies when theta=-pi/4 or 0 for d=1/sqrt(2) and d=(-1)

Hough Transform Graph



θ	r
15	189.0
30	282.0
45	355.7
60	407.3
75	429.4

θ	r
15	318.5
30	376.8
45	407.3
60	409.8
75	385.3

θ	r
15	419.0
30	443.6
45	438.4
60	402.9
75	340.1

Notice the difference (and relationship) between the slope of the line and the theta of polar representation.

(Image from Wikipedia)

Hough Transform

Choose all the correct answers for the following questions.

(d) (2 points) What are the main differences between using Hough transform for detecting lines and Hough transform for detecting circles?

- ■ The Hough Transform for lines uses a 2D parameter space, while the Hough Transform for circles uses a 3D parameter space.
- □ The Hough Transform for circles can detect lines, but the Hough Transform for lines cannot detect circles.
- □ There is no difference; both use the same parameter space and techniques.

Solution: Hough transform for lines use theta and p (distance from origin), hence 2D parameter space. For circle, it is (cx,cy,r) where (cx,cy) are coordinates of circle and r is radius. Hence, hough transform for circle cannot directly detect lines as they are different in terms of parameter space and technique used.

Hough Transform

(e) (2 points) What are the benefits of using Hough Transform for line detection?

- ■ Robustness to noise
- ■ Tolerance to disconnected segments
- ☐ High detection speed and low memory use
- ☐ All of the above

- Solution: Hough Transform estimates parameters in hough space based on voting and hence it is robust noise. Further, it can detect broken/disconnected lines based on local gradients and voting.
- Also, Hough Transform usually has polynomial time complexity and polynomial space complexity (depends on specific implementations). So speed and space are not something making HT outstanding.

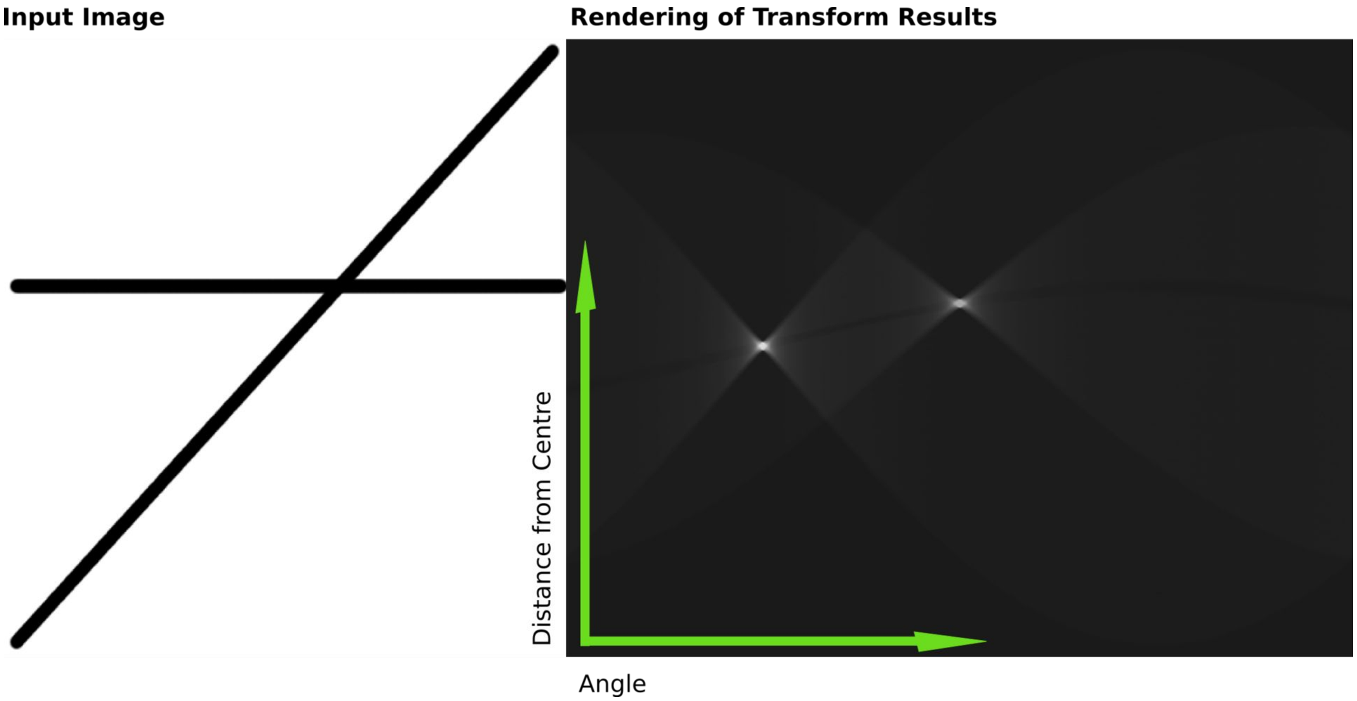
Hough Transform

(f) (2 points) What does a point in the accumulator space of a Hough Transform represent?

- ☐ The intensity of a pixel in the original image
- ■ A potential line in the image space
- ☐ The color of a detected line
- ☐ The gradient of an edge in the image

Solution: Accumulator space consists of slope (theta) and p (distance from origin) which represents a line in image space.

Hough Space



(Image from Wikipedia)

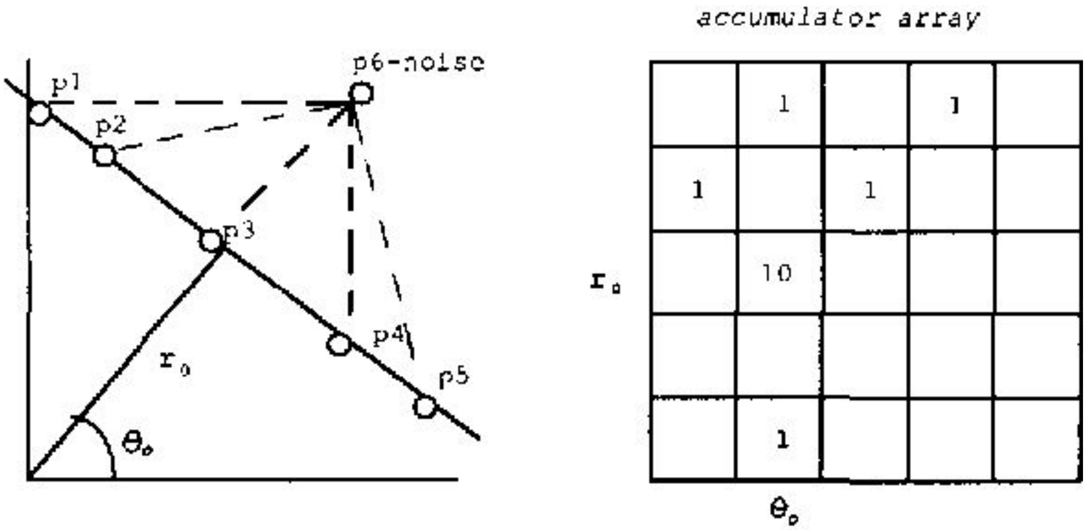
Hough Transform

(g) (3 points) In a 300x300 pixel image, the Hough Transform is applied to detect lines. After processing the image, the highest value in the accumulator matrix is found at the coordinates (90, 106). What do these coordinates represent in the context of the Hough Transform?

- ☐ The slope and intercept of the detected line in the image space.
- ■ The angle and distance from the origin of the detected line in the image space.
- ☐ The number of lines and their average length in the image space.
- ☐ The center and radius of a detected circle in the image space.

Solution: The point in accumulator space represents slope and distance of line from origin as shown in the slide above.

Voting Accumulator Array (Hough Array)

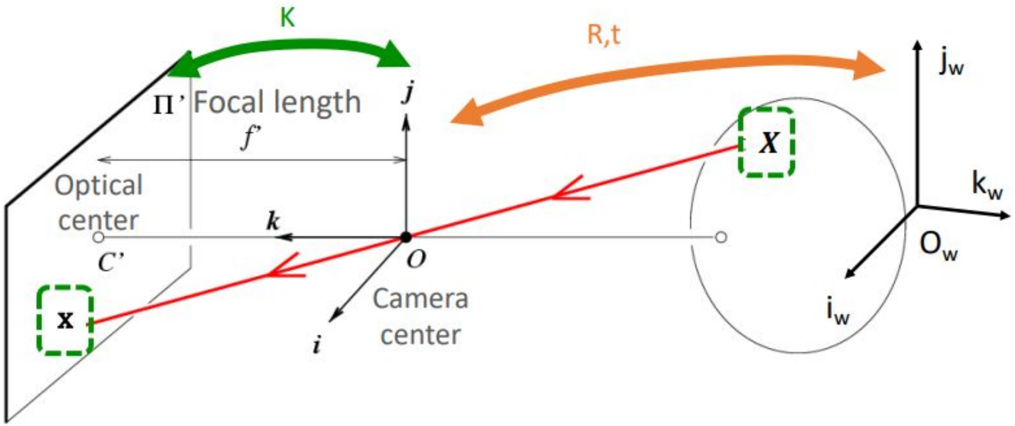


The left and right images don't match in number, just for reference. Accumulator array (hough array) can be considered as a low-resolution grid simulation of the hough space.

(Image from Ben-Tzvi, D. and Mark B. Sandler. "A combinatorial Hough transform." Pattern Recognit. Lett. 11 (1990): 167-174.)

Projective Geometry & Camera Calibration

Projection Matrix: Converting 3D → 2D



Intrinsic = Camera Properties

Extrinsic = World Properties

$$\mathbf{x} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\mathbf{X}$$

x: Image Coordinates: (u,v,1)

K: Intrinsic Matrix (3x3)

R: Rotation (3x3)

t: Translation (3x1)

X: World Coordinates: (X,Y,Z,1)

$$K = \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}$$

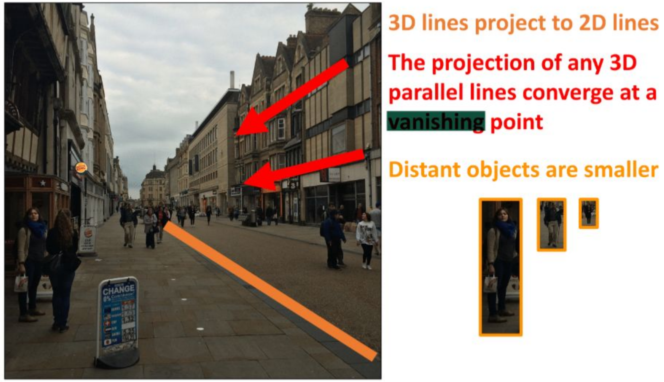
(a) (2 points) In the context of camera calibration, what does the extrinsic matrix account for?

- ☐ Properties of the camera lens
- ■ Rotation and translation of the camera with respect to the world
- ☐ The light absorption of the camera sensor
- ☐ The resolution of the camera sensor

The extrinsic parameters represent world properties, specifically the rotation and translation of the camera with respect to the world coordinate system

(b) (2 points) How many vanishing points can be found in an image where parallel lines are converging?

- ■ One for each set of parallel lines
- ☐ Two for each set of parallel lines
- ☐ Only one regardless of the number of parallel lines
- ☐ It varies depending on the angle of the camera



List of properties from M. Hebert

Parallel lines in the world intersect in the image at a "vanishing point", implying a single vanishing point for each set of parallel lines.

c) True/False Questions

1. Perspective projection always maintains the parallelism of lines from the 3D world in the 2D image plane.

   False. Perspective projection does not always maintain the parallelism of lines from the 3D world in the 2D image plane. Parallel lines in the 3D world may converge in the image, leading to vanishing points.

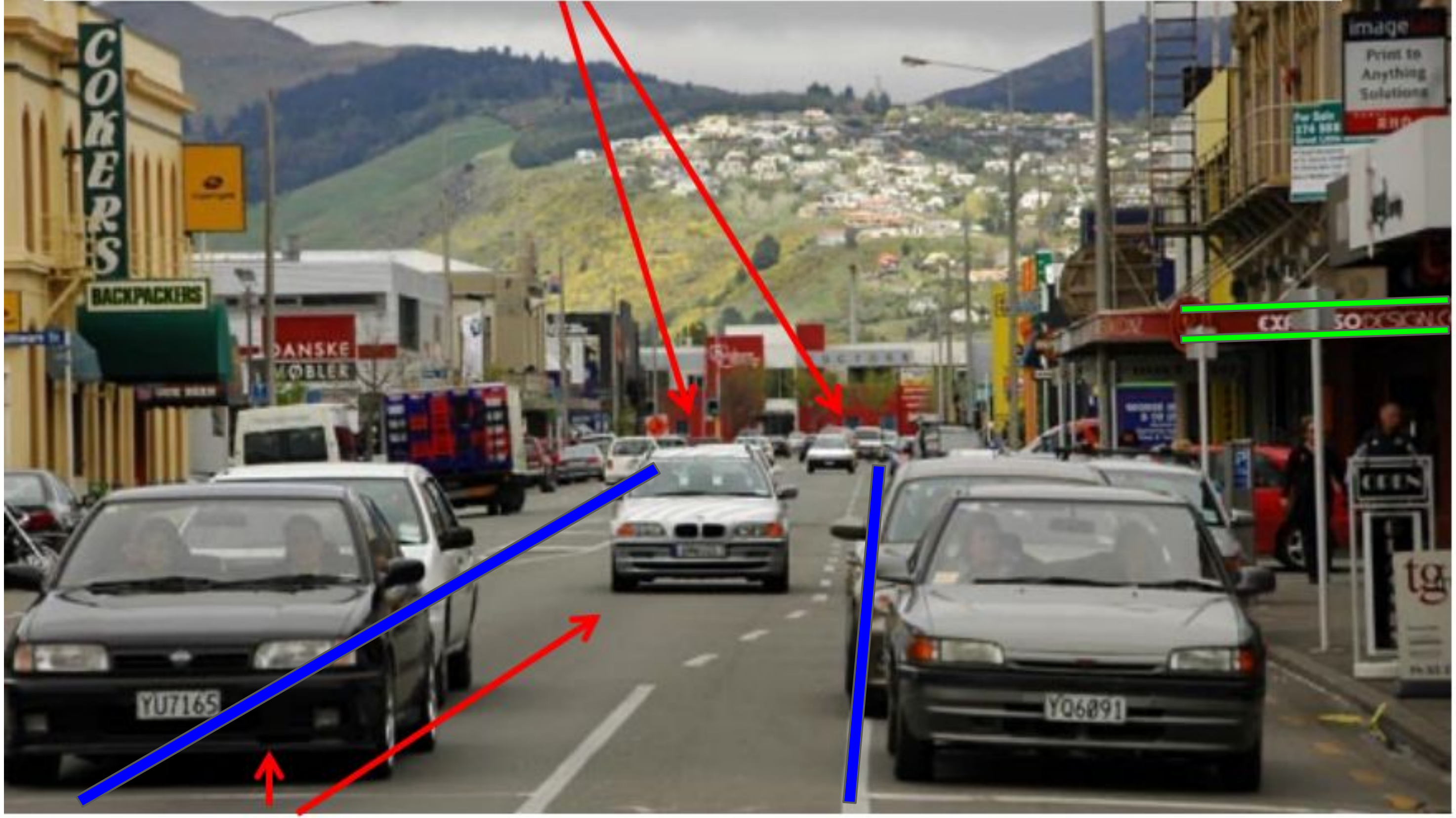
2. When calibrating a camera, the intrinsic matrix is used to correct for lens distortion.

   True. The focal lengths and optical center coordinates are scaled by distortion coefficients.

3. A square in the real world always projects as a square in the image captured by a camera.

   False. A square in the real world does not always project as a square in the image captured by a camera due to perspective projection, which can distort the shapes.

Far field: object appearance doesn't change as objects translate



Near field: object appearance changes as objects translate

Recall that the camera projection matrix is given by: $M = K [R \mid t]$

(d) (3 points) Briefly describe what the camera projection matrix is used for.

The camera projection matrix, denoted as $M = K[R \mid t]$, is used for converting coordinates from the 3D world to the 2D image plane. It incorporates both intrinsic (K) parameters, such as focal length and optical center, and extrinsic ($[R \mid t]$) parameters, which account for the camera's rotation and translation in the world.

(e) (4 points) How many degrees of freedom are in each of the following matrices/vectors?

i. 3D Rotation matrix, R — 3, Rotation angles about the three axes

ii. 3D translation vector, t — 3, Translation along the three axes

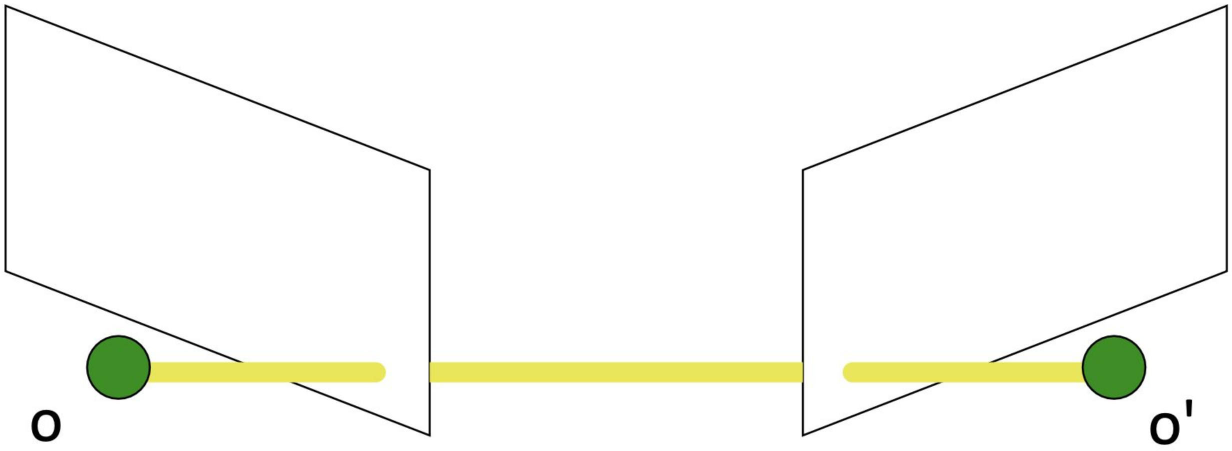
iii. Camera Intrinsic matrix, K — 5, Variables in the intrinsic matrix - focal length, skew factor and optical center

$$\begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

iv. Camera Projection matrix, M — 11, M = K[R | t]. 6+5 = 11 parameters

Epipolar Geometry

Epipolar Geometry

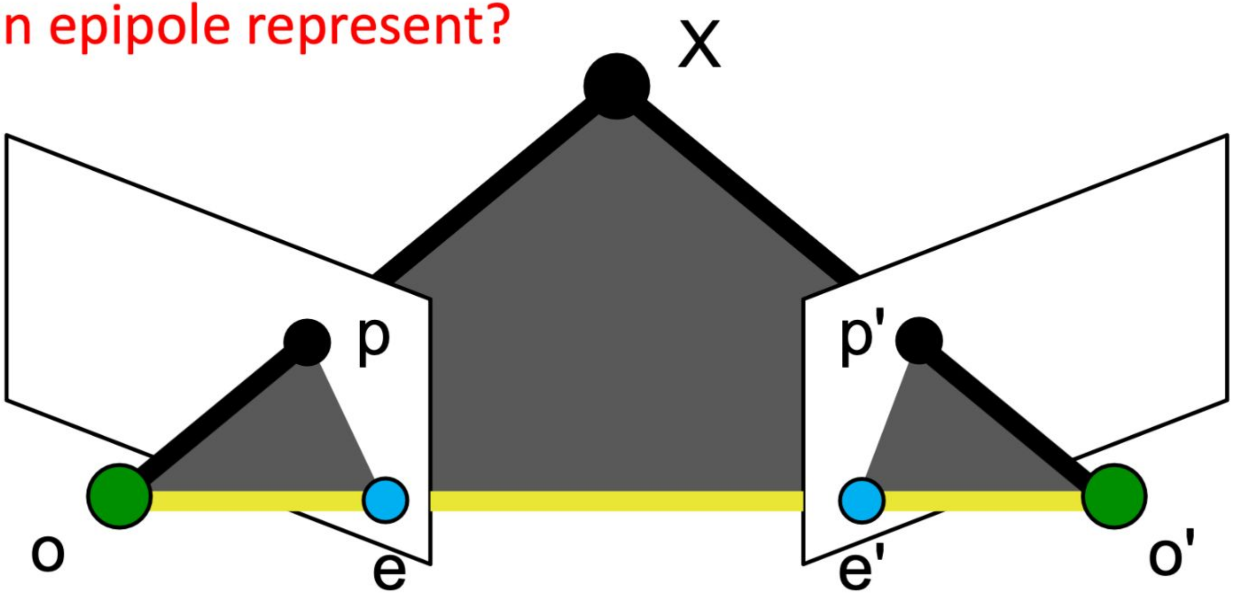


Suppose we have two cameras at origins o, o’

Baseline is the line connecting the origins

Epipolar Geometry

What does an epipole represent?

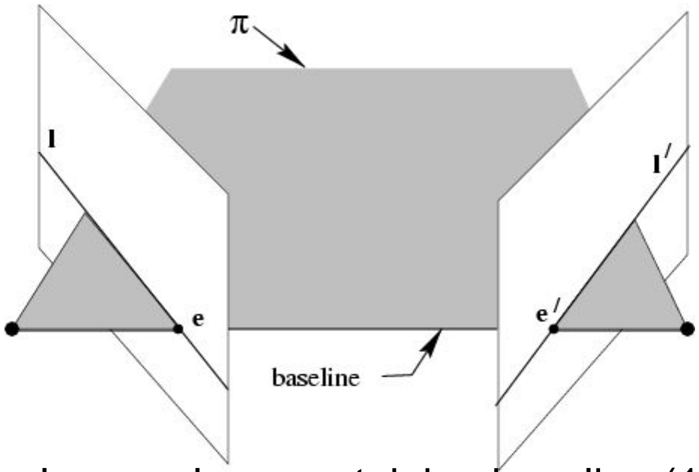


- Epipoles e, e' are where the baseline intersects the image planes
- Epipole is the projection of other camera origin in the image plane

Slide credit: David Fouhey

Epipolar Geometry

epipoles e,e’
= intersection of baseline with image plane
= projection of projection center in other image
= vanishing point of camera motion direction

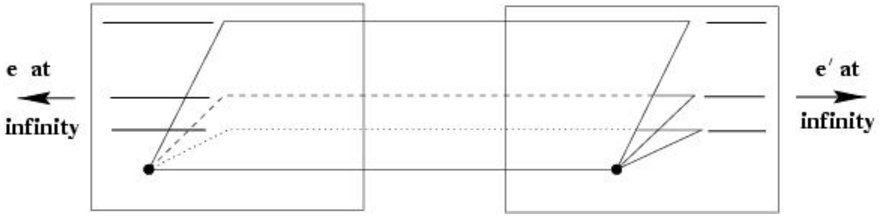


an epipolar plane = plane containing baseline (1-D family)

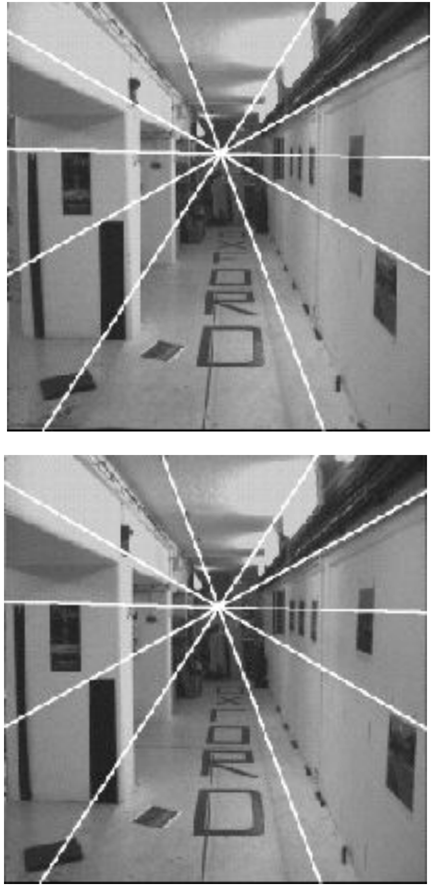
an epipolar line = intersection of epipolar plane with image
(always come in corresponding pairs)

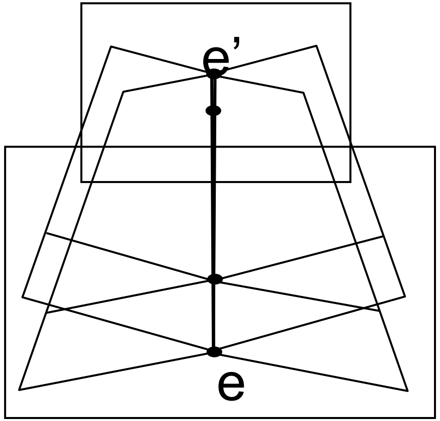
Slide Credit : Prof. Yiannis Aloimonos, UMD.

Motion parallel with image plane



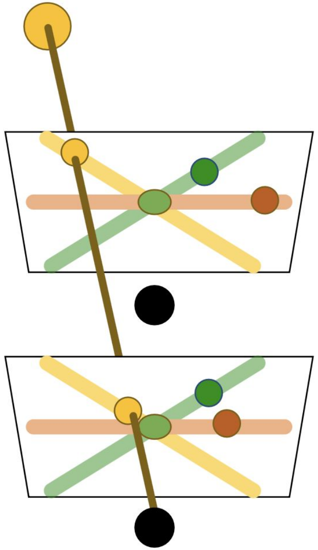
Forward Motion



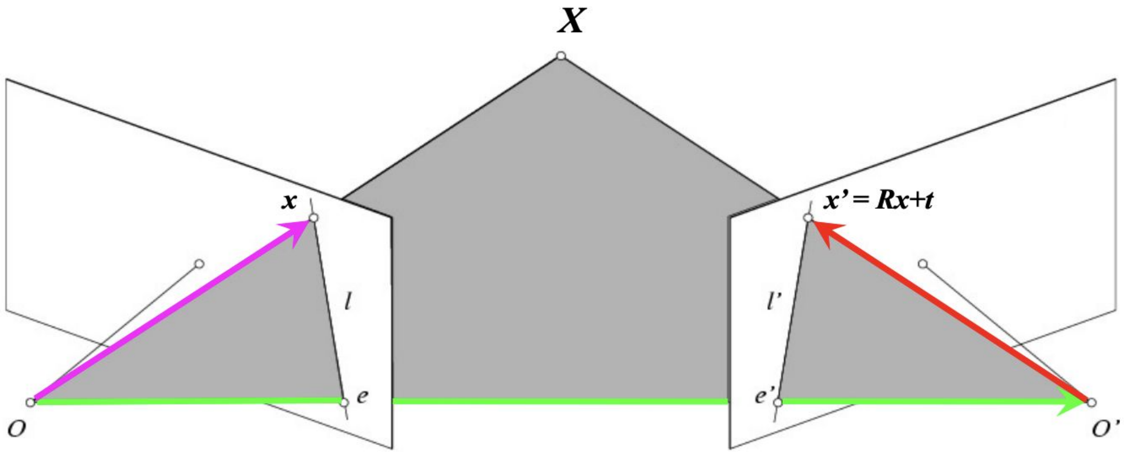


Epipole is focus of expansion / optical center of the camera.

Epipolar lines go out from the optical center



Epipolar constraint: Calibrated case



$$\boldsymbol{x}'\cdot[\boldsymbol{t}\times(\boldsymbol{R}\boldsymbol{x})]=0 \Rightarrow \boldsymbol{x}'^T\underbrace{[\boldsymbol{t}_\times]\boldsymbol{R}}_{E}\boldsymbol{x}=0 \Rightarrow \boldsymbol{x}'^T E\boldsymbol{x}=0$$

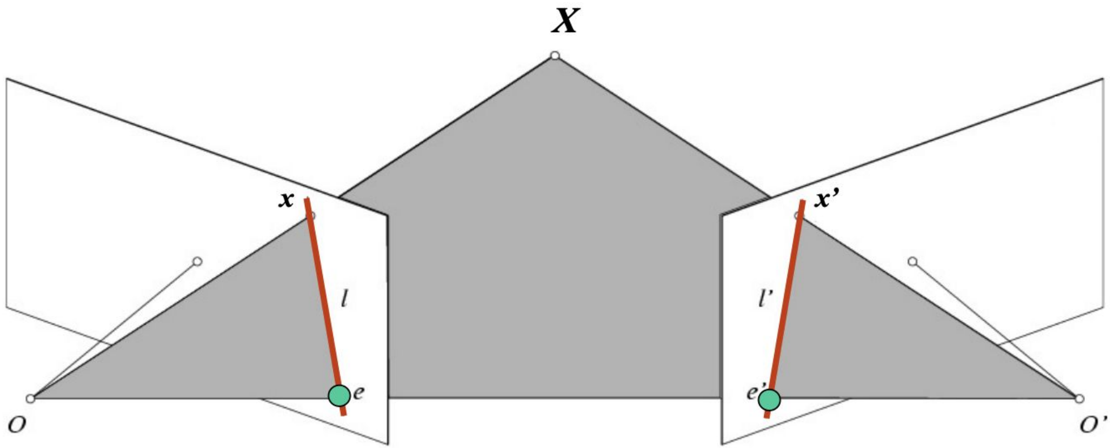
Recall: $\mathbf{a}\times\mathbf{b}=\begin{bmatrix}0 & -a_z & a_y\\ a_z & 0 & -a_x\\ -a_y & a_x & 0\end{bmatrix}\begin{bmatrix}b_x\\ b_y\\ b_z\end{bmatrix}=[\mathbf{a}_\times]\mathbf{b}$

Essential Matrix
(Longuet-Higgins, 1981)

Slide Credit : Berkeley Angjoo Kanazawa & Alexei Efros

The vectors $\boldsymbol{x}$, $\boldsymbol{t}$, and $\boldsymbol{x}'$ are coplanar

Epipolar constraint: Uncalibrated case



$$x'^T E x = 0 \quad \Rightarrow \quad \hat{x}'^T F \hat{x} = 0 \quad \text{with} \quad F = K'^{-T} E K^{-1}$$

- $F\hat{x}$ is the epipolar line associated with $\hat{x}$ ($l' = F\hat{x}$)
- $F^T\hat{x}'$ is the epipolar line associated with $\hat{x}'$ ($l = F^T\hat{x}'$)
- $F\mathbf{e} = 0$ and $F^T\mathbf{e}' = 0$
- ***F*** is singular (rank two)
- ***F*** has *seven* degrees of freedom

Slide Credit : Berkeley Angjoo Kanazawa & Alexei Efros

Essential matrix

- Encodes information of the extrinisic parameters only
- E is of rank 2
- Its two nonzero singular values are equal
- Has only 5 degrees of freedom, 3 for rotation, 2 for translation

Slide Credit : Prof. Yiannis Aloimonos, UMD.

Fundamental matrix

- Encodes information of the intrinsic and extrinisic parameters
- F is of rank 2
- Has 7 degrees of freedom. There are 9 elements, but scaling is not significant and det F = 0

(a) (2 points) Which of the following quantities are required to be known in order to **solve for** (not define) the Essential matrix using the epipolar constraint?
select all that apply.

- ☐ R (rotation)
- ☐ t (translation)
- ■ K (instrincs)
- ☐ 3D to 2D correspondences
- ■ 2D correspondences

(b) (2 points) Which of the quantities needed to solve for the Essential matrix (i.e., selected above) are unknown when solving for the Fundamental matrix?

Solution: K

(c) (3 points) What is the dimension of the Essential / Fundamental matrices?

Solution: 3x3

(d) (3 points) What is the rank of the Essential / Fundamental matrices?

Solution: Rank 2

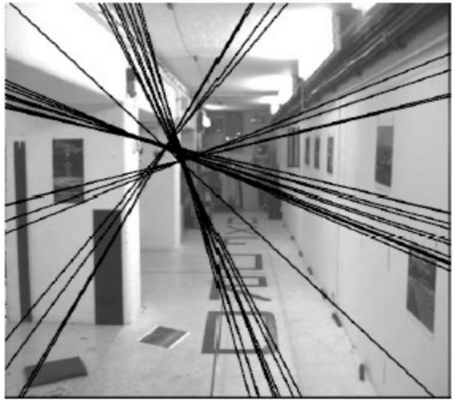
The eight-point algorithm

$$\mathbf{x} = (u, v, 1)^T, \quad \mathbf{x}' = (u', v', 1)$$

$$\begin{bmatrix} u' & v' & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0 \quad \Rightarrow \quad \begin{bmatrix} u'u & u'v & u' & v'u & v'v & v' & u & v & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

Solve homogeneous linear system using eight or more matches

Enforce rank-2 constraint (take SVD of ***F*** and throw out the smallest singular value)



Slide Credit : Berkeley Angjoo Kanazawa & Alexei Efros

Eight Point Algorithm

The input is formed by m point correspondences, m >= 8

- Construct the m x 9 matrix A
- Find the SVD of A: $A = UDV^T$
- The entries of F are the components of the column of V corresponding to the least s.v.
- Enforce the rank 2 constraint by performing SVD on F and setting a singular value to zero.

(e) (4 points) When estimating the fundamental matrix F using the 8-point algorithm, you compute the SVD (Singular value decomposition) of the estimated F matrix, as shown below.

$$F = U \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} V^T$$

Does this estimated F give the correct result?

☐ Yes ■ No

If not, what steps would you take to fix it?

Solution: It does not give the correct result. We know F needs to be singular/rank 2. How do we force F to be singular. Open it up with SVD, zero a singular value,

$$\hat{F} = U \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

Image rectification

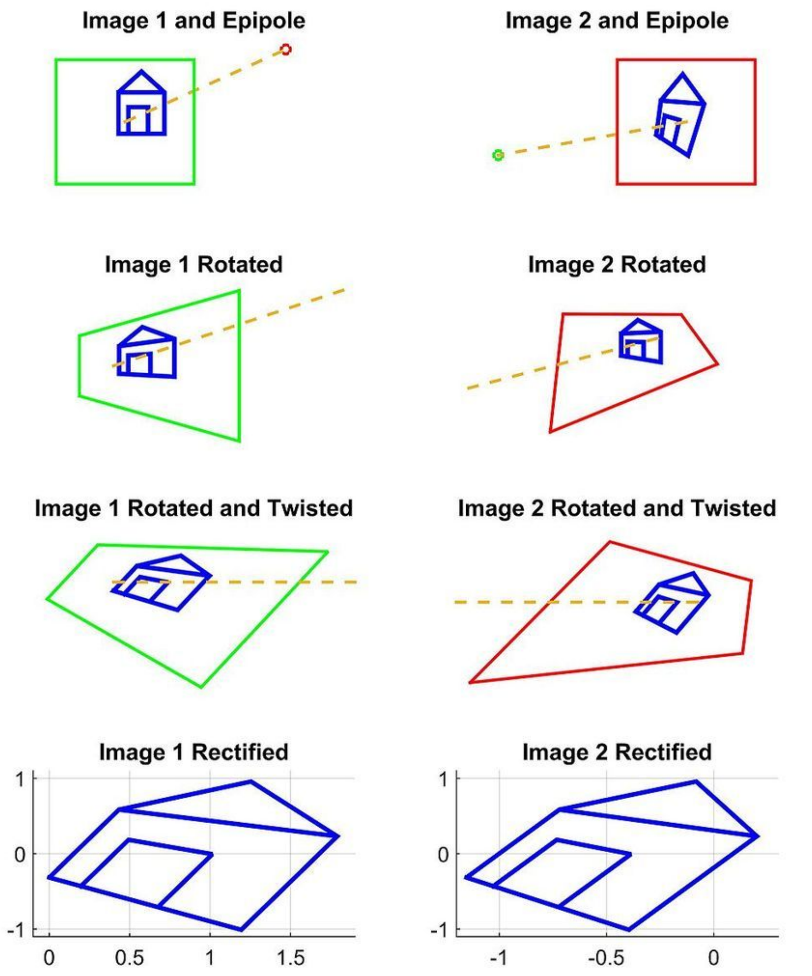
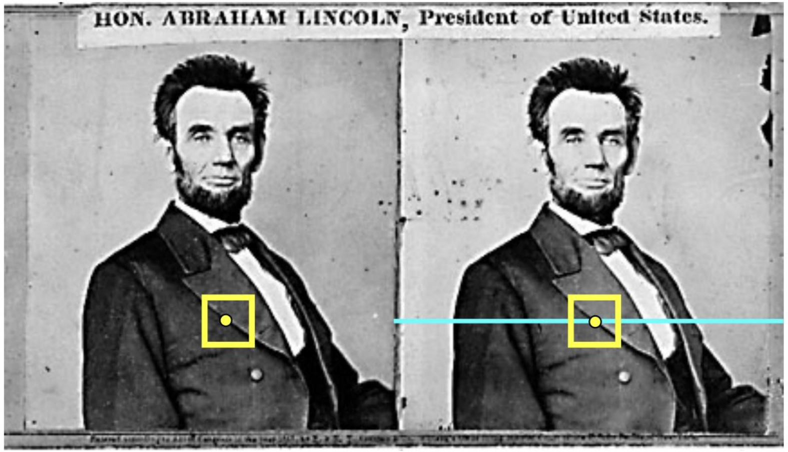


Image rectification warps both images such that they appear as if they have been taken with only a horizontal displacement and as a consequence all epipolar lines are horizontal, which slightly simplifies the stereo matching process.

Source : [Wikipedia]

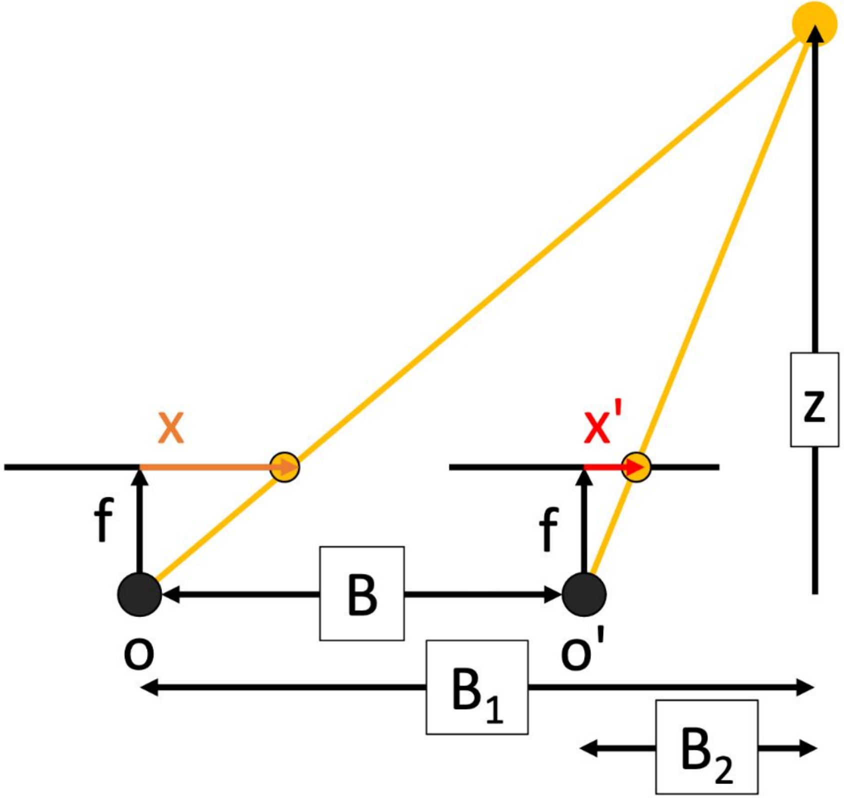
Basic stereo matching algorithm



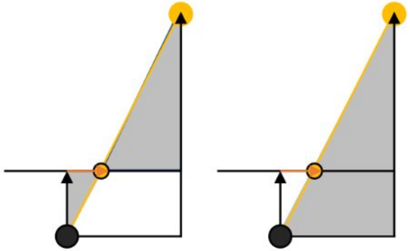
If necessary, rectify the two stereo images to transform epipolar lines into scanlines

- For each pixel x in the first image
    - Find corresponding epipolar scanline in the right image
    - Search the scanline and pick the best match x'
    - Triangulate matches to get the depth information

Triangulation: Depth from disparity

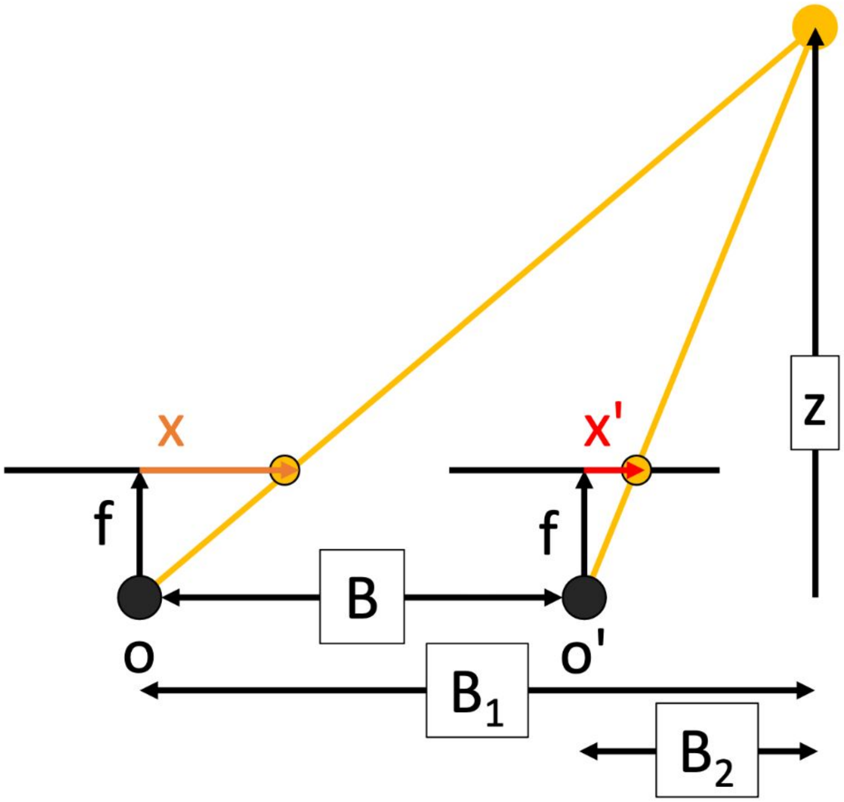


$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{x'}{f} = \frac{B_2}{z}$$



By similar triangles

Triangulation: Depth from disparity



$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{x'}{f} = \frac{B_2}{z}$$

Subtract them

$$\frac{x - x'}{f} = \frac{B_1 - B_2}{z}$$

$$\underbrace{x - x'}_{\text{Disparity}} = \frac{fB}{z}$$

(f) (3 points) Why would one choose to use image rectification before searching for correspondences across stereo images?

> **Solution:** Epipolar lines become horizontal scanlines, which is easier to search between.

(g) (3 points) Write the steps in basic stereo matching algorithm.

> **Solution:** If necessary, rectify the two stereo images to transform epipolar lines into scanlines • For each pixel x in the first image • Find corresponding epipolar scanline in the right image • Search the scanline and pick the best match x' • Triangulate matches to get the depth information

(h) (2 points) In stereo vision, what does the term "baseline" refer to?
select all that apply.

- ☐ The distance between the principal point and the image plane.
- ■ The distance between the optical centers of two cameras.
- ☐ The distance between the camera and the object being viewed.
- ☐ The distance between the lenses and the image sensor.
- ☐ The distance between the camera lens and the camera body.

Pointers

- Please remember mentor's name (write their initials on top of the exam sheet). Mentor list for reference: Google Sheets
- No one is allowed to leave the exam hall during the exam duration before submission.
- Be on time, reach by 2 p.m.
- Can only bring pen and white sheets for calculation.
- Adhere to the academic policy of integrity. Any type of cheating will not be tolerated.
- One double sided formula sheet allowed.

Questions?

Best of luck!