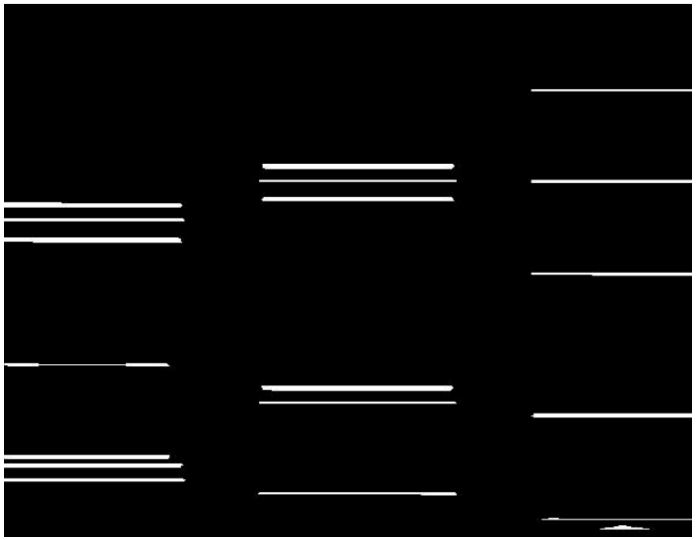
Feature detection and matching
- Lines

Objectives

- Learn the basics of Line Detection
- Learn the most popular line detectors: Hough transform.
- Learn the application of convolution-based techniques in line detection
- Applications of line detection in image processing

What is Line detection?

- Line detection is an algorithm that takes a collection of n edge points and finds all the lines on which these edge points lie.
- Two types of techniques:
  - The Hough transform
  - Convolution-based techniques



Line detection Steps

- The lane detection pipeline follows these steps:
  - Pre-process image using grayscale and Gaussian blur.
  - Apply canny edge detection to the image.
  - Apply masking region to the image.
  - Apply Hough transform to the image.
  - Extrapolate the lines found in the Hough transform to construct the left and right lane lines.

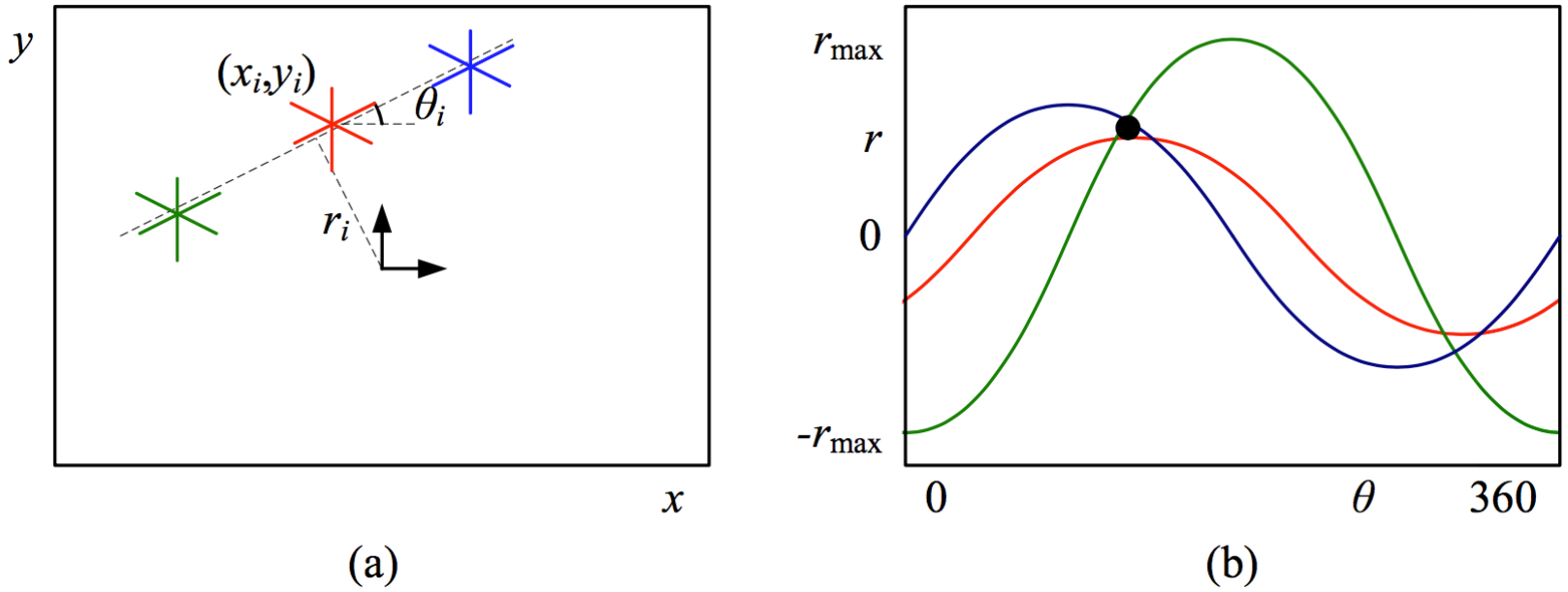
Line detection
- Hough transform

FPT Education
FPT UNIVERSITY

- It is used to recognize complex lines in photographs.
- For the Hough Transform algorithm, it is crucial to perform edge detection first to produce an edge image which will then be used as input into the algorithm.
- The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.
- Now day, the Hough transform has been extended to identifying positions of arbitrary shapes, most commonly circles or ellipses.

Line detection - Hough transform

- Each edge point votes for all possible lines passing through it, and lines corresponding to high accumulator or bin values are examined for potential line fits

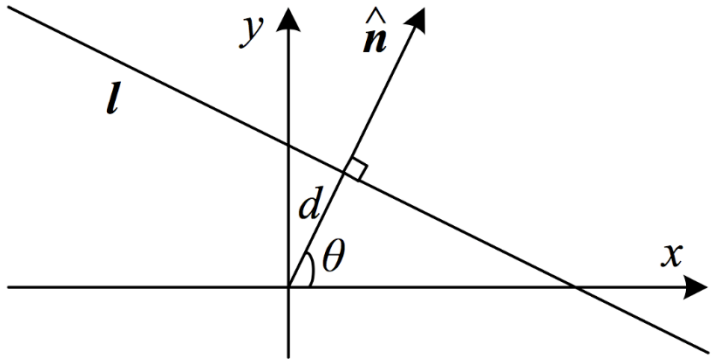


Line detection - Hough transform

- Before we can vote for line hypotheses, we must first choose a suitable representation
- The normal-distance (n̂ , d) parameterization for a line.

  - To obtain a minimal two-parameter representation for lines, we convert the normal vector into an angle

$$\theta = \tan^{-1} n_y / n_x$$

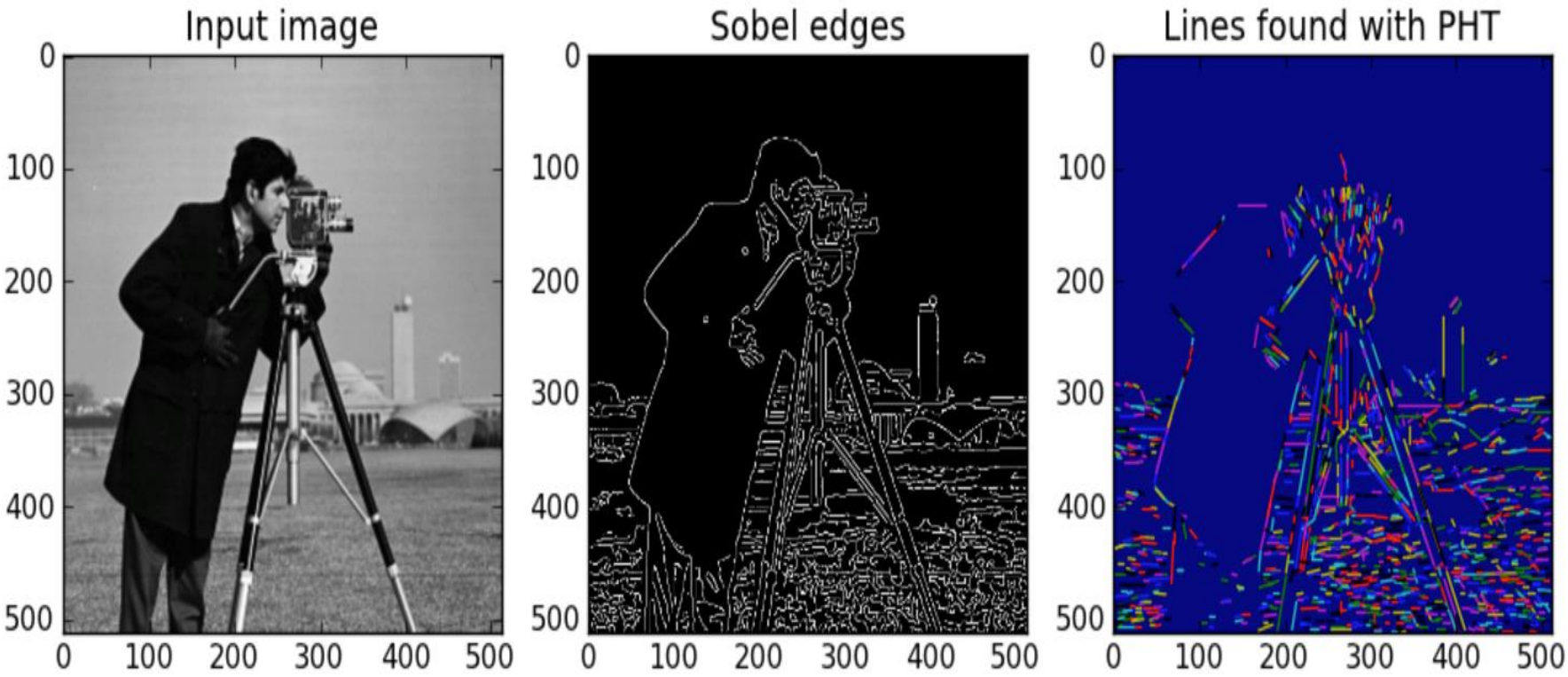


Line detection - Hough transform

procedure *Hough*$(\{(x, y, \theta)\})$:

1. Clear the accumulator array.
2. For each detected edgel at location (x, y) and orientation $\theta = \tan^{-1} n_y/n_x$, compute the value of
$$d = x\, n_x + y\, n_y$$
and increment the accumulator corresponding to (θ, d).
3. Find the peaks in the accumulator corresponding to lines.
4. Optionally re-fit the lines to the constituent edgels.

Line detection
- Hough transform



Line detection - Convolution based technique

FPT Education
FPT UNIVERSITY

- A convolution-based technique that produces an image description of the thin lines in an input image
- In a convolution-based technique, the line detector operator consists of a convolution mask tuned to detect the presence of lines of a particular width n and a θ orientation
- Four convolution masks
  - detect horizontal lines
  - detect vertical lines
  - detect oblique (+45 degrees) lines
  - detect oblique (–45 degrees) lines

Line detection
- Convolution based technique

Horizontal mask(R1)

-1	-1	-1
2	2	2
-1	-1	-1

Vertical (R3)

-1	2	-1
-1	2	-1
-1	2	-1

Oblique (+45 degrees)(R2)

-1	-1	2
-1	2	-1
2	-1	-1

Oblique (−45 degrees)(R4)

2	-1	-1
-1	2	-1
-1	-1	2

Line detection - Convolution based technique

- These masks above are tuned for light lines against a dark background, and would give a big negative response to dark lines against a light background.
- If you are only interested in detecting dark lines against a light background, then you should negate the mask values.
- Alternatively, you might be interested in either kind of line, in which case, you could take the absolute value of the convolution output.

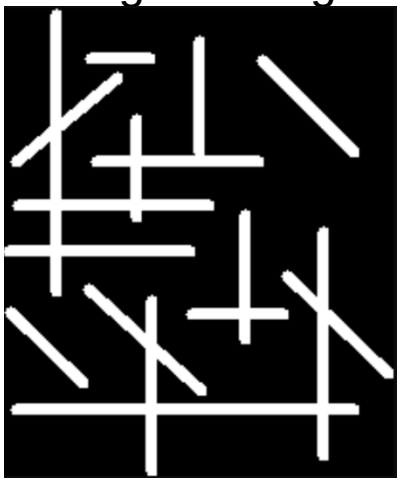
Line detection - Convolution based technique

FPT Education
FPT UNIVERSITY

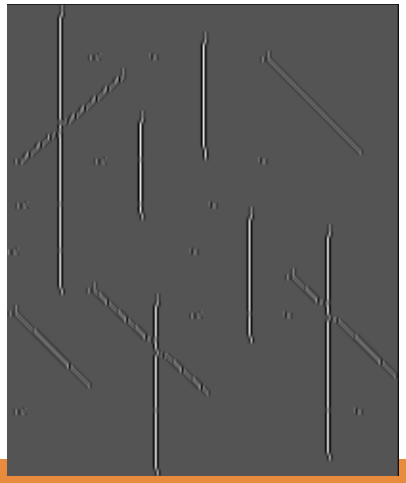
- If Ri denotes the response of kernel i, we can apply each of these kernels across an image, and for any particular point, if Ri>Rj for all i # j that point is more likely to contain a line whose orientation (and width) corresponds to that of kernel i.
- One usually thresholds Ri to eliminate weak lines corresponding to edges and other features with intensity gradients that have a different scale than the desired line width.
- In order to find complete lines, one must join together line fragments, e.g., with an edge tracking operator.

Line detection - Convolution based technique

Original image



vertical kernel



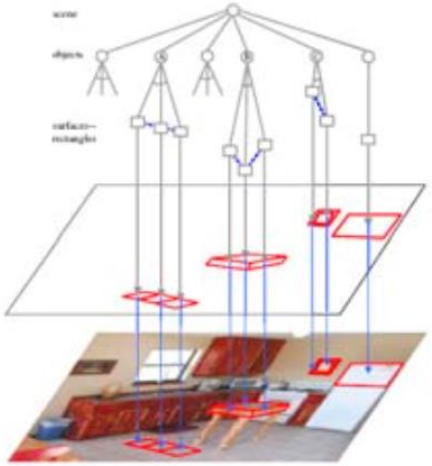
the oblique 45 degree

the oblique 135 degree

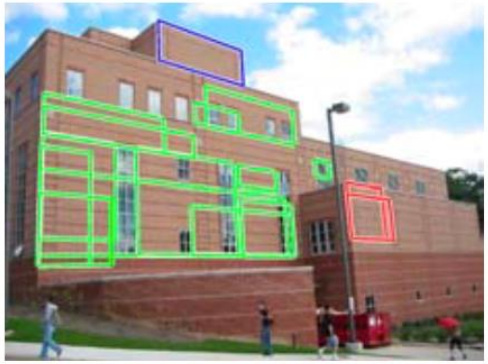
Application: Rectangle detection

(a)

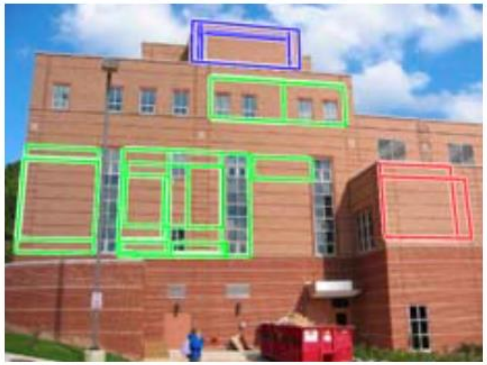
(b)



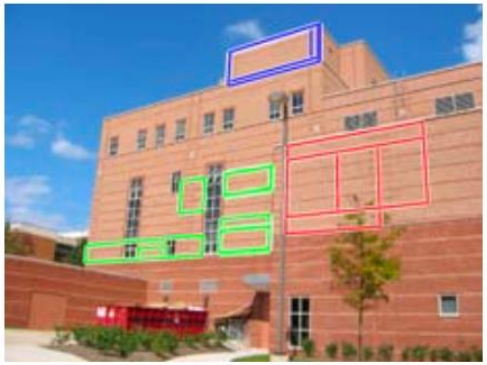
(c)



(d)



(e)



(f)

Summary

- Learn the basics of Line Detection
- Learn the most popular line detectors: Hough transform.
- Learn the application of convolution-based techniques in line detection
- Applications of line detection in image processing