Computer Vision

Lecture 3: Hough Transform

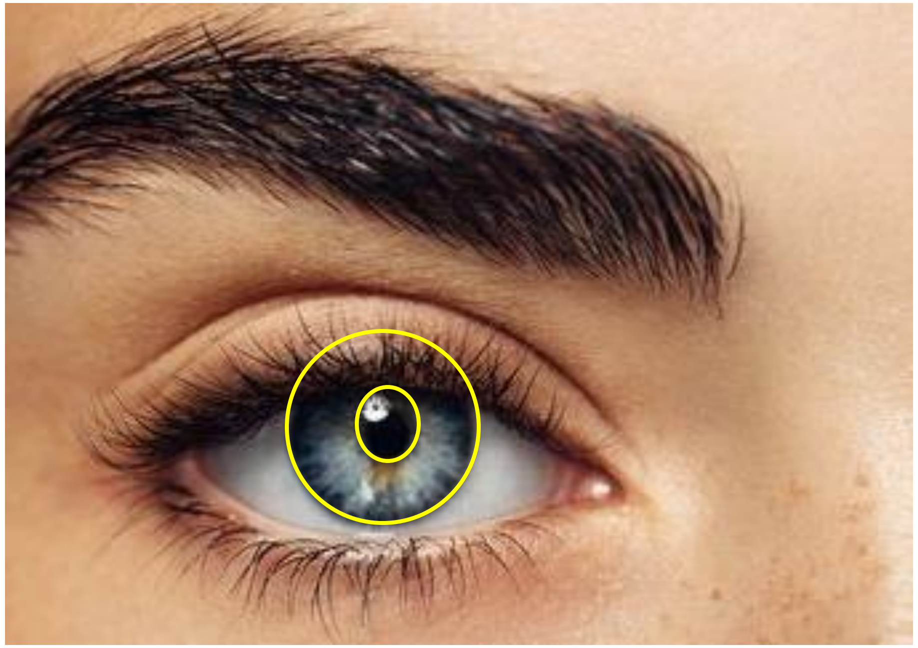
30.10.2024

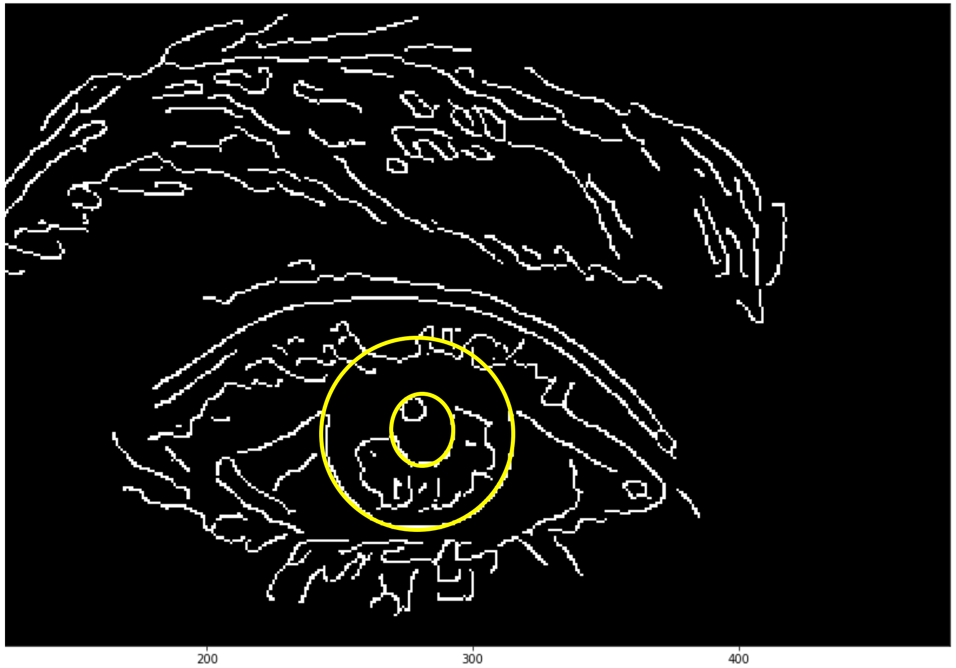
Manuel Heurich, Tim Landgraf

List of Topics

1.	Introduction (Color)	Computer Vision Fundamentals
2.	Edge Detectors	
3.	**Hough Transform**	Conventional Computer Vision
4.	Histogram-based Detection and Tracking	
5.	Optic flow	
6.	SIFT / SURF	
7.	Introduction to Neural Information Processing	Deep Learning (Supervised)
8.	Convolutional Neural Networks	
9.	Image Classification, Object Detection	
10.	Semantic Segmentation	
11.	Pose Estimation	
12.	Style Transfer	
13.	Recurrent Neural Networks, Image Captioning	
14.	Generative Models	Unsupervised Learning
15.	Unsupervised feature extraction	

Last class: edge detection





Today: fit objects to edge image

The Hough Transform

Cited by 6141

United States Patent Office

3,069,654
Patented Dec. 18, 1962

1

3,069,654
METHOD AND MEANS FOR RECOGNIZING COMPLEX PATTERNS
Paul V. C. Hough, Ann Arbor, Mich., assignor to the United States of America as represented by the United States Atomic Energy Commission
Filed Mar. 25, 1960, Ser. No. 17,715
6 Claims. (Cl. 340—146.3)

2

of the point on the line segment from the horizontal midline 109 of the framelet 108.

(3) Each line in the transformed plane is made to have an intercept with the horizontal midline 101 of the picture 100 equal to the horizontal coordinate of its respective point on the line segment in framelet 108.

Thus, for a given reference point 110 on line segment 102 a line 110A is drawn in the plane transform 102A. The reference point 110 is approximately midway between

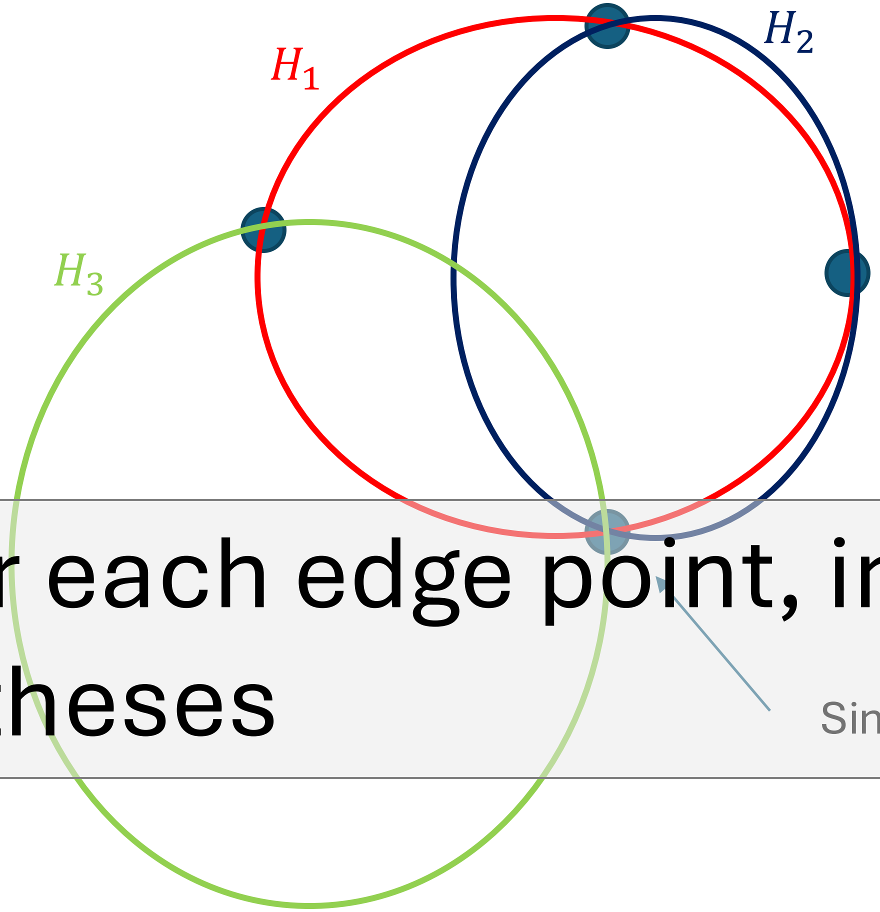
Duda & Hart propose an extension to the idea (1972) - this is the one we used since then

Cited by 8903

The Hough Transform

- The „vanilla“ HT **robustly** finds simple shapes in edge images
  - robust = even with **missing edge pieces** or noise
  - usually used for lines, circles, ellipses
  - can be used for any **parametric shape model**
- The **Generalized Hough Transform**
  - finds *any* shape, given a template

Let ● be an edge pixel. Which ellipse hypothesis is more likely?



edge „score“

$H_1 : 4$

$H_2 : 3$

$H_3 : 2$

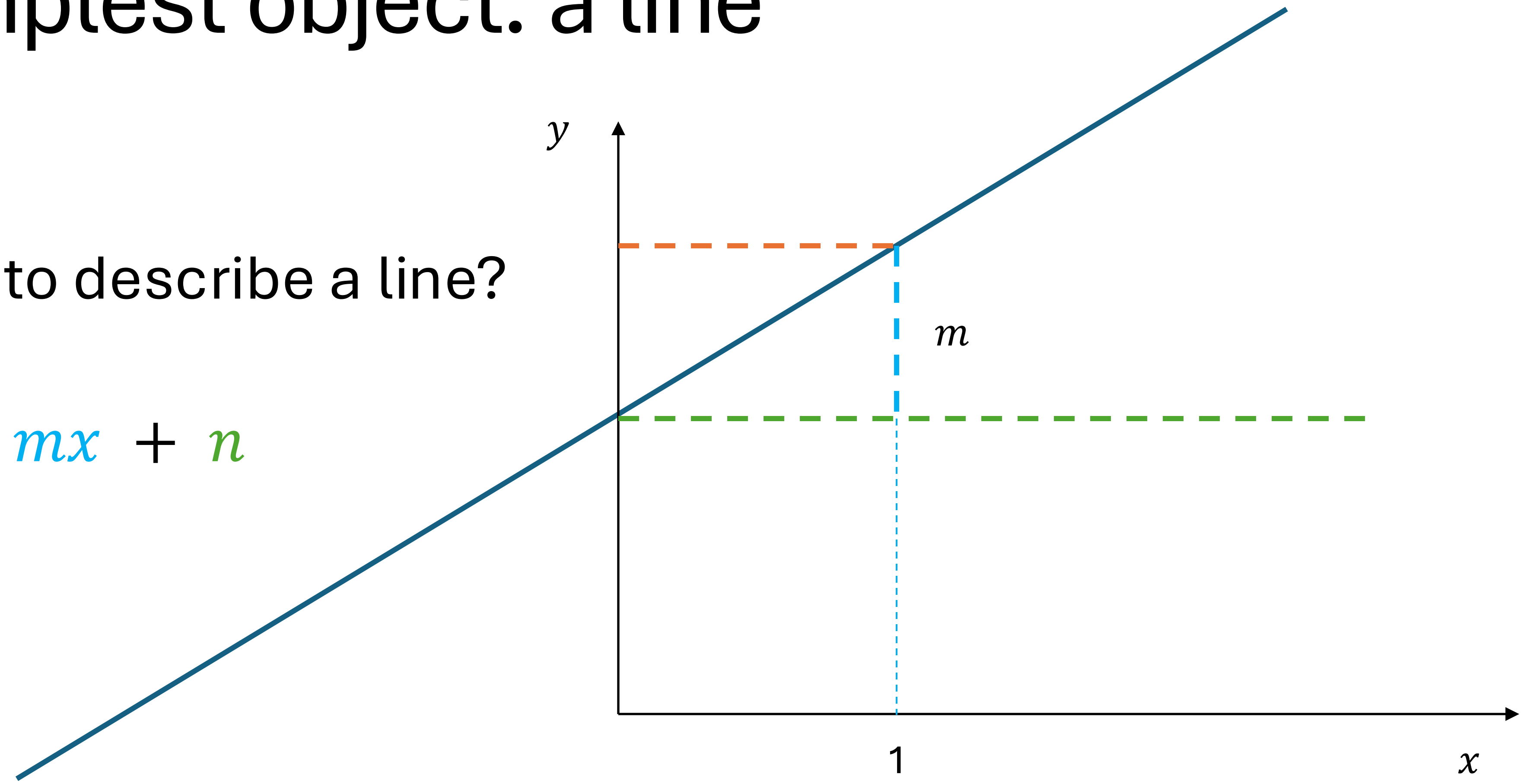
IDEA: For each edge point, increment (a discrete set of) hypotheses

Single edge pixels support multiple hypotheses!

Simplest object: a line

How to describe a line?

$$y = mx + n$$

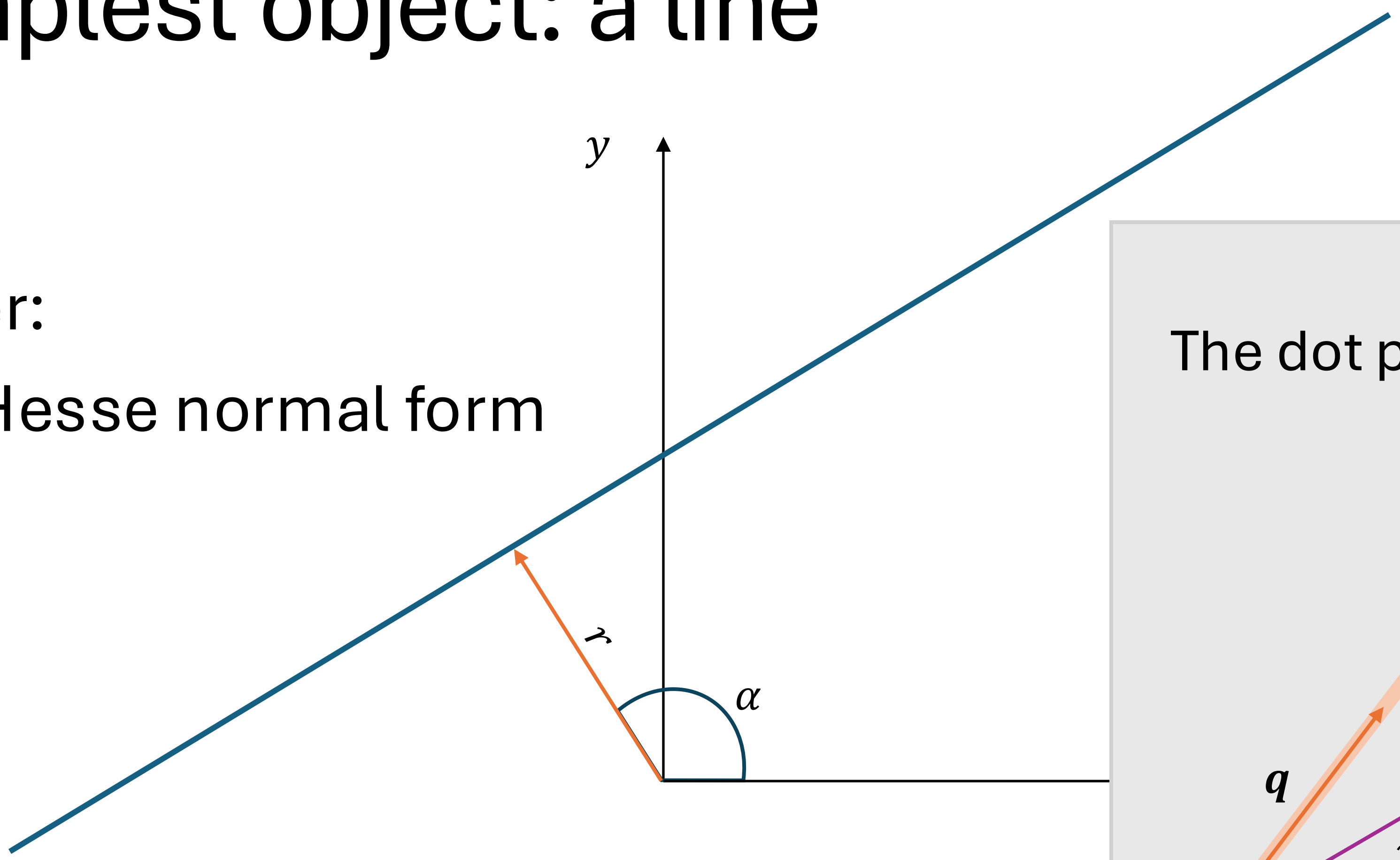


Q: what are valid parameters for a vertical line?

Simplest object: a line

Better:

use Hesse normal form



$$r = x\cos(\alpha) + y\sin(\alpha)$$

The dot product: **BFF**!

$$\boldsymbol{x}^T\boldsymbol{q} = s$$

with $|q| = \mathbf{1}$

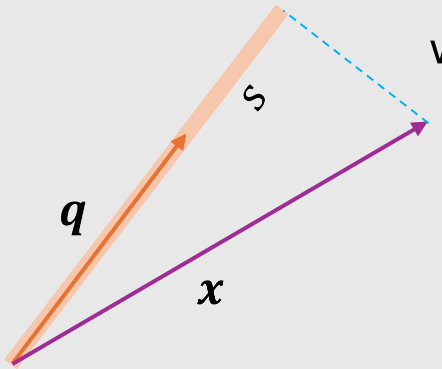
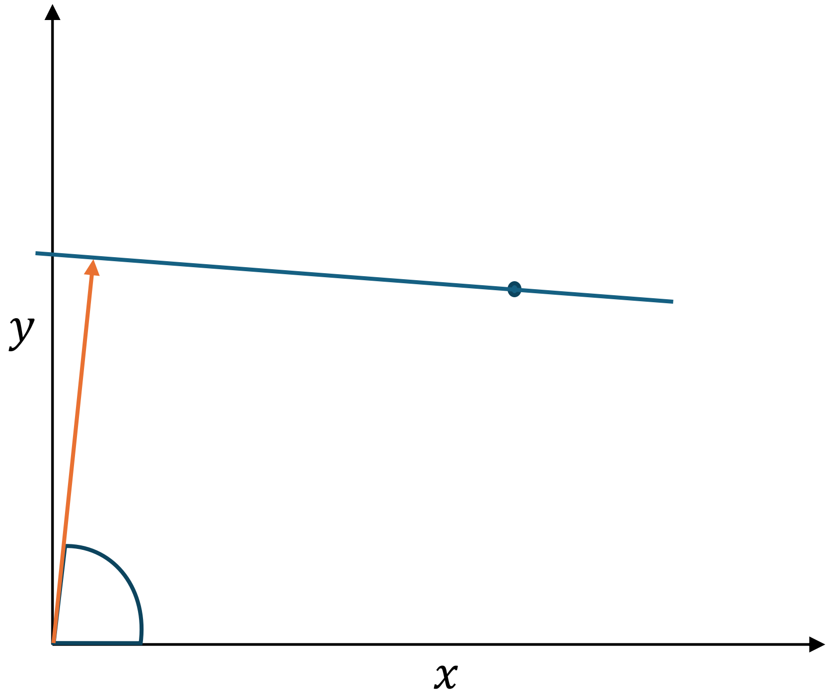


Image space and parameter space

Image



Parameter space

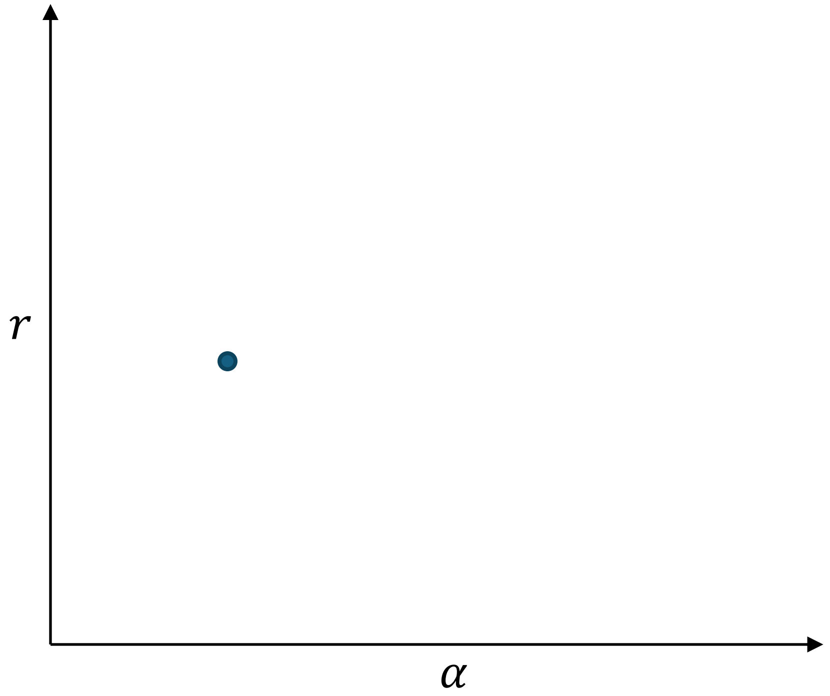
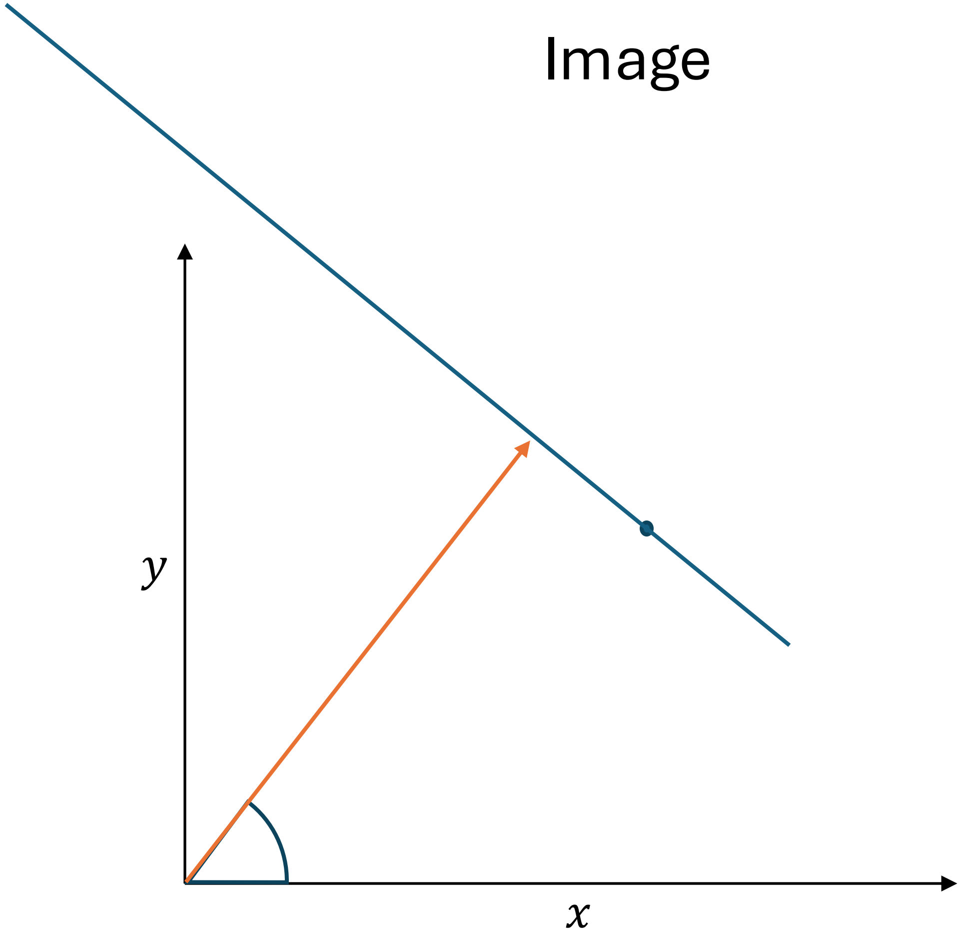


Image space and parameter space

Image



Parameter space

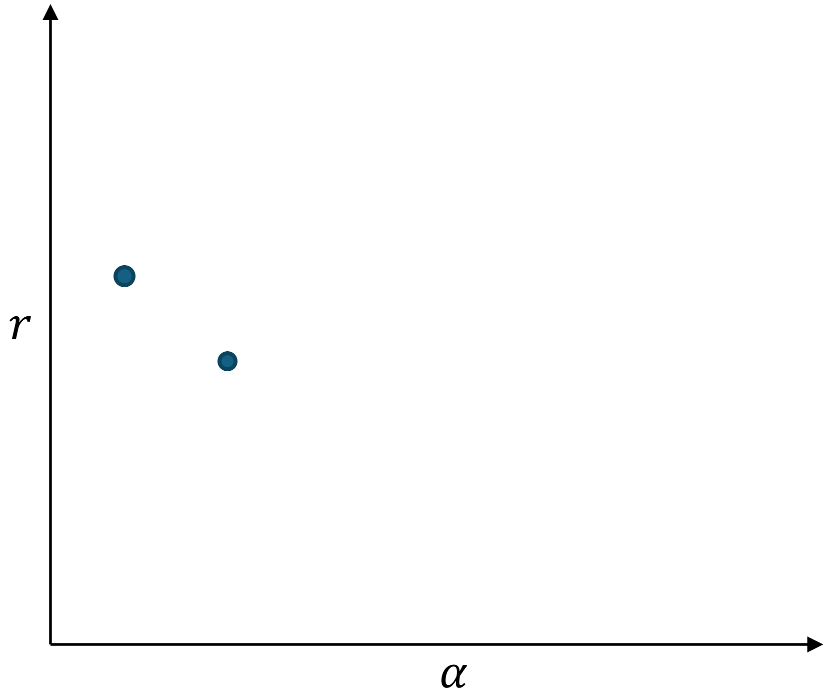
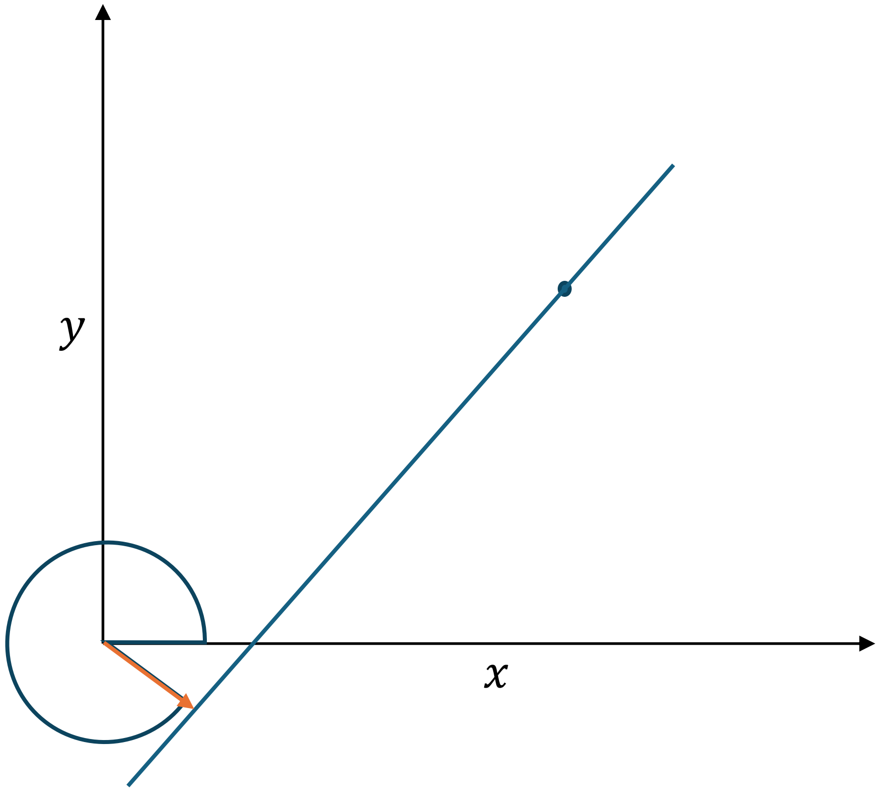


Image space and parameter space

Image



Parameter space

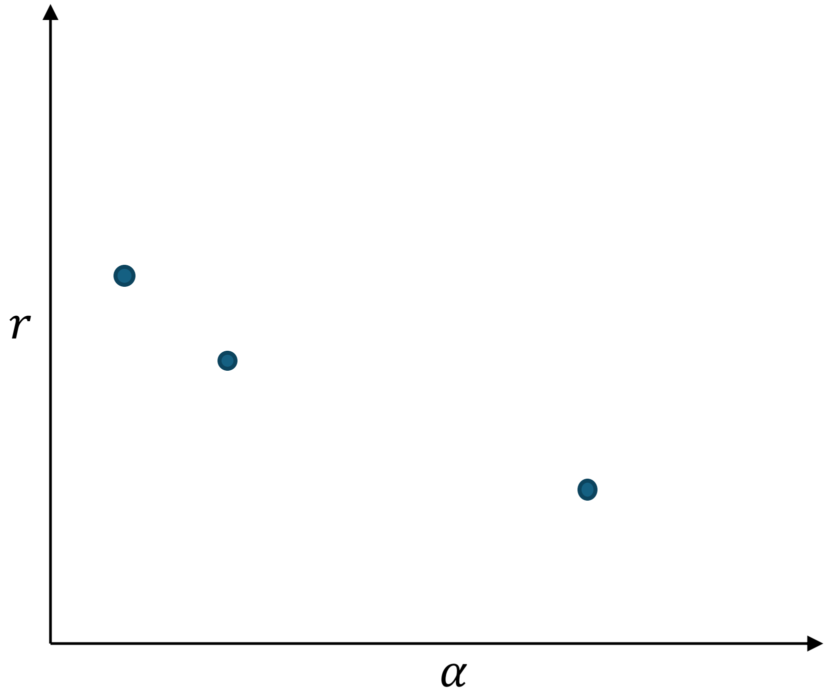
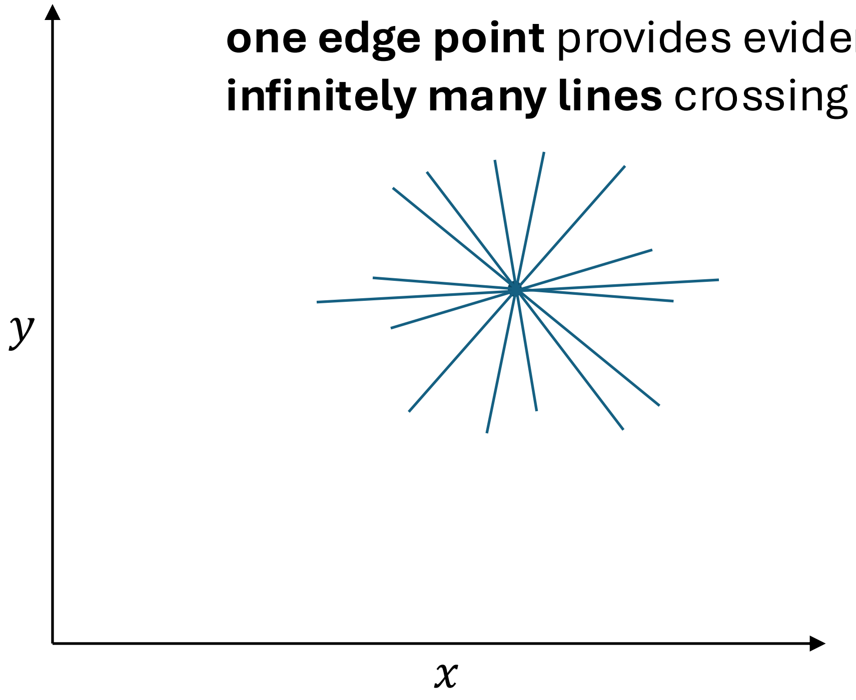


Image space and parameter space

Image

one edge point provides evidence for **infinitely many lines** crossing that point



Parameter space

each of these lines is a **point in parameter space**, hence: **one edge point** in the image yields **one curve** in parameter space

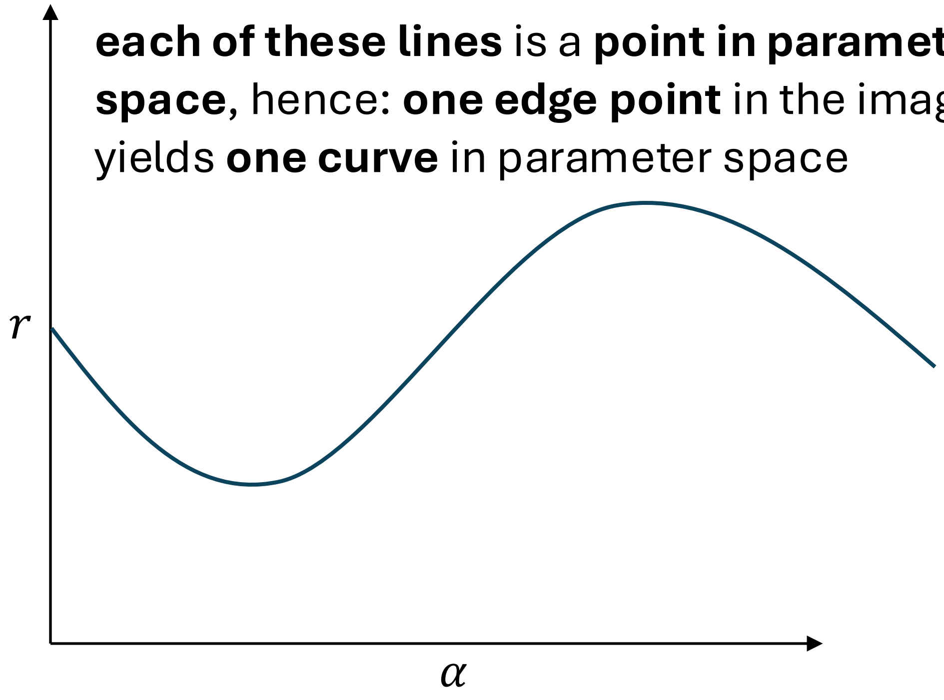
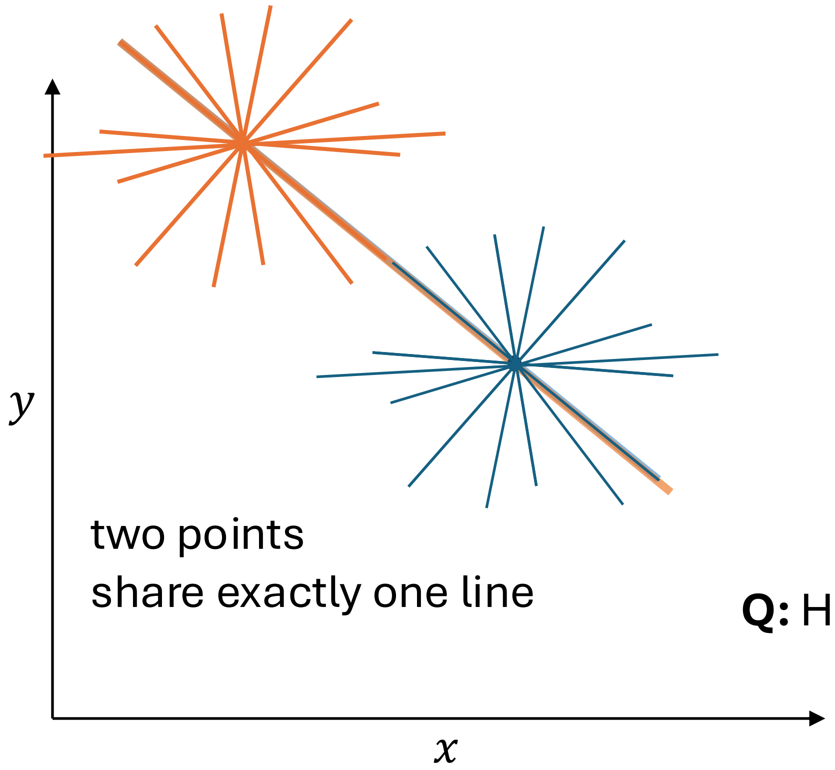
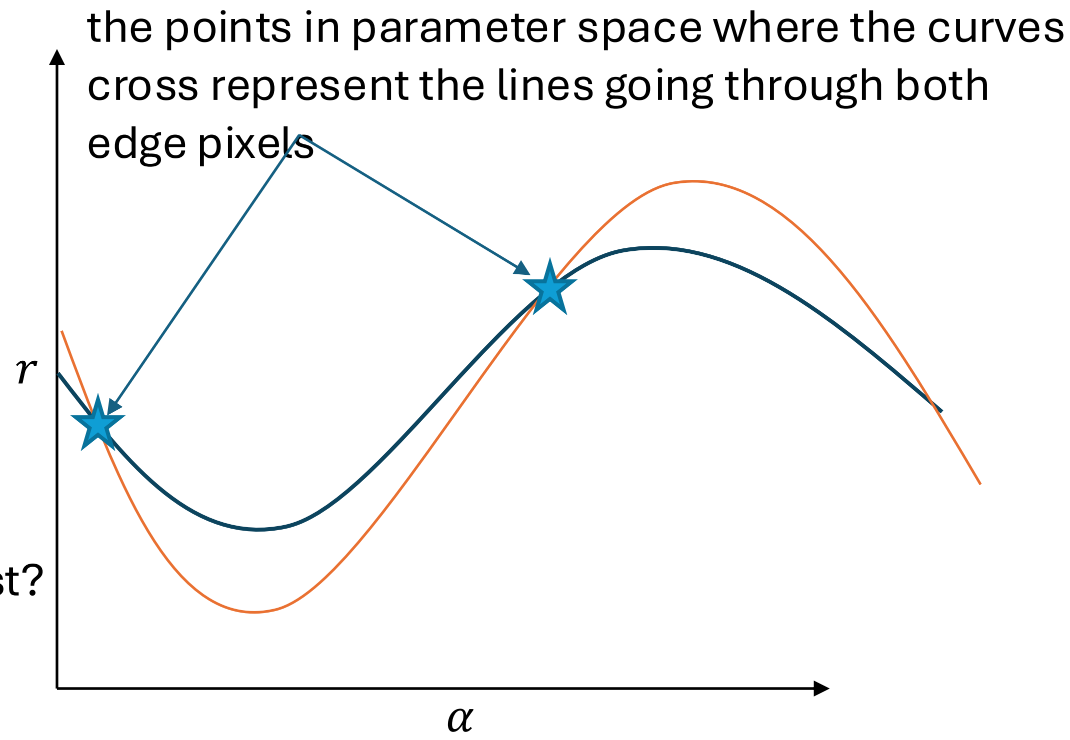


Image space and parameter space

Image



Parameter space

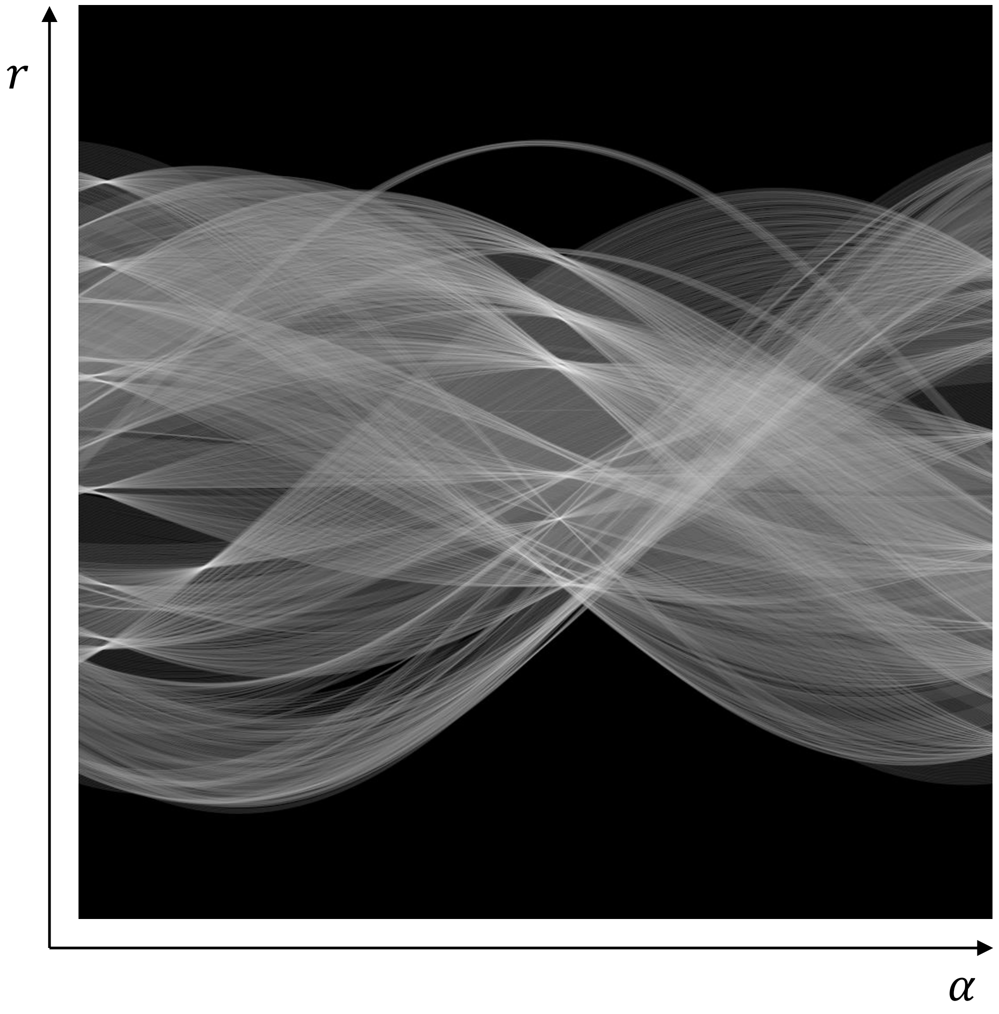


Q: How many solutions exist?

The „accumulator“

```
for each edge pixel at x,y
        for a = 0 : pi
                r = x * cos(a) + y * sin(a)
                A(a,r)++
```

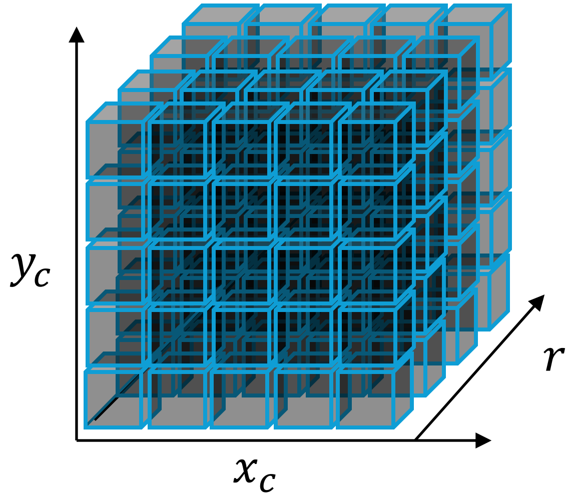
→ find indices of local maxima in A

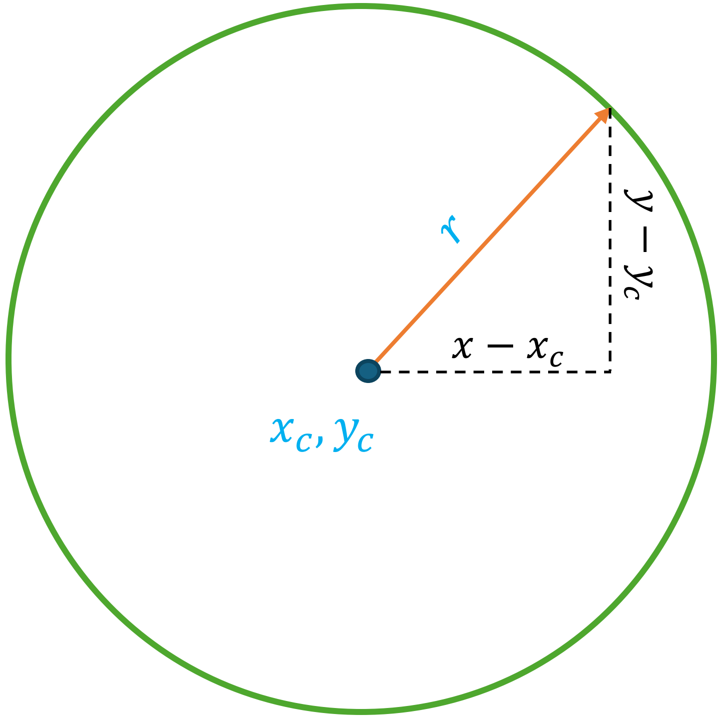


The Hough Transform for Circles

Circle equation:

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

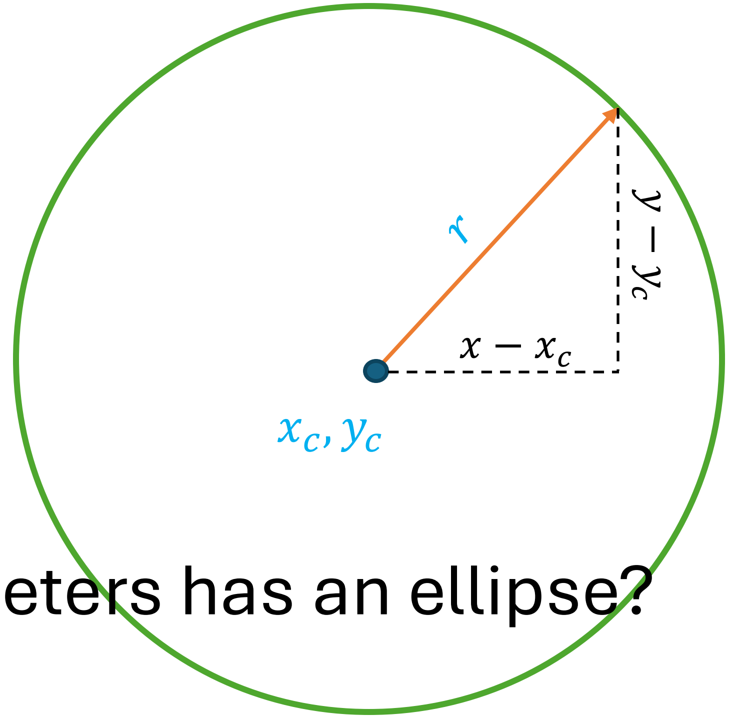


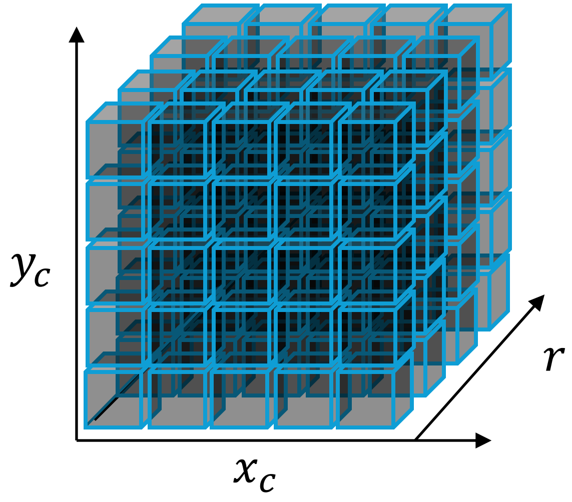


The Hough Transform for Circles

Circle equation:

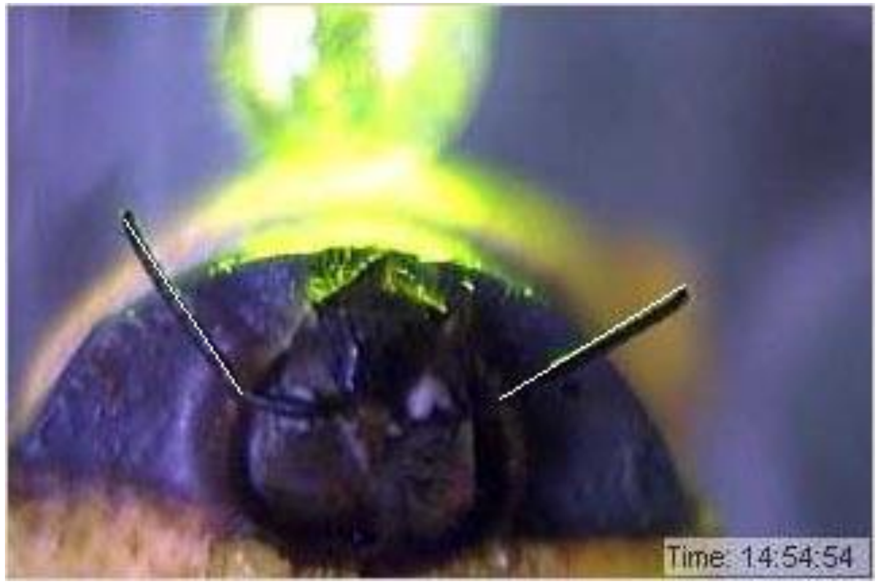
$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

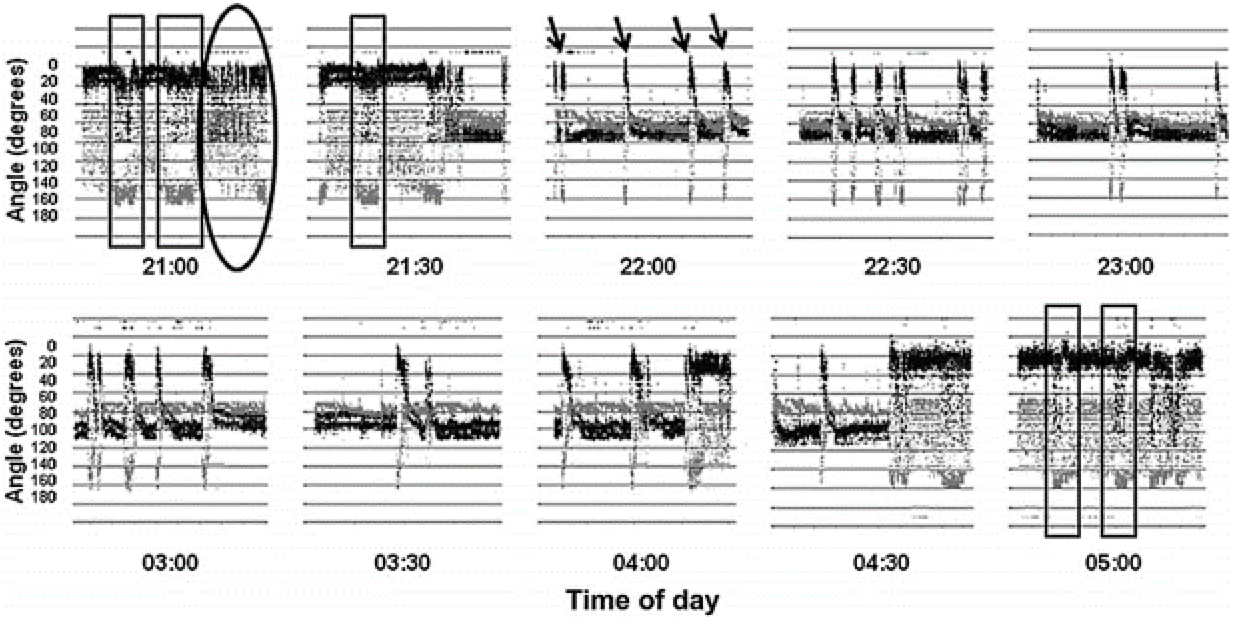




Q: How many parameters has an ellipse?

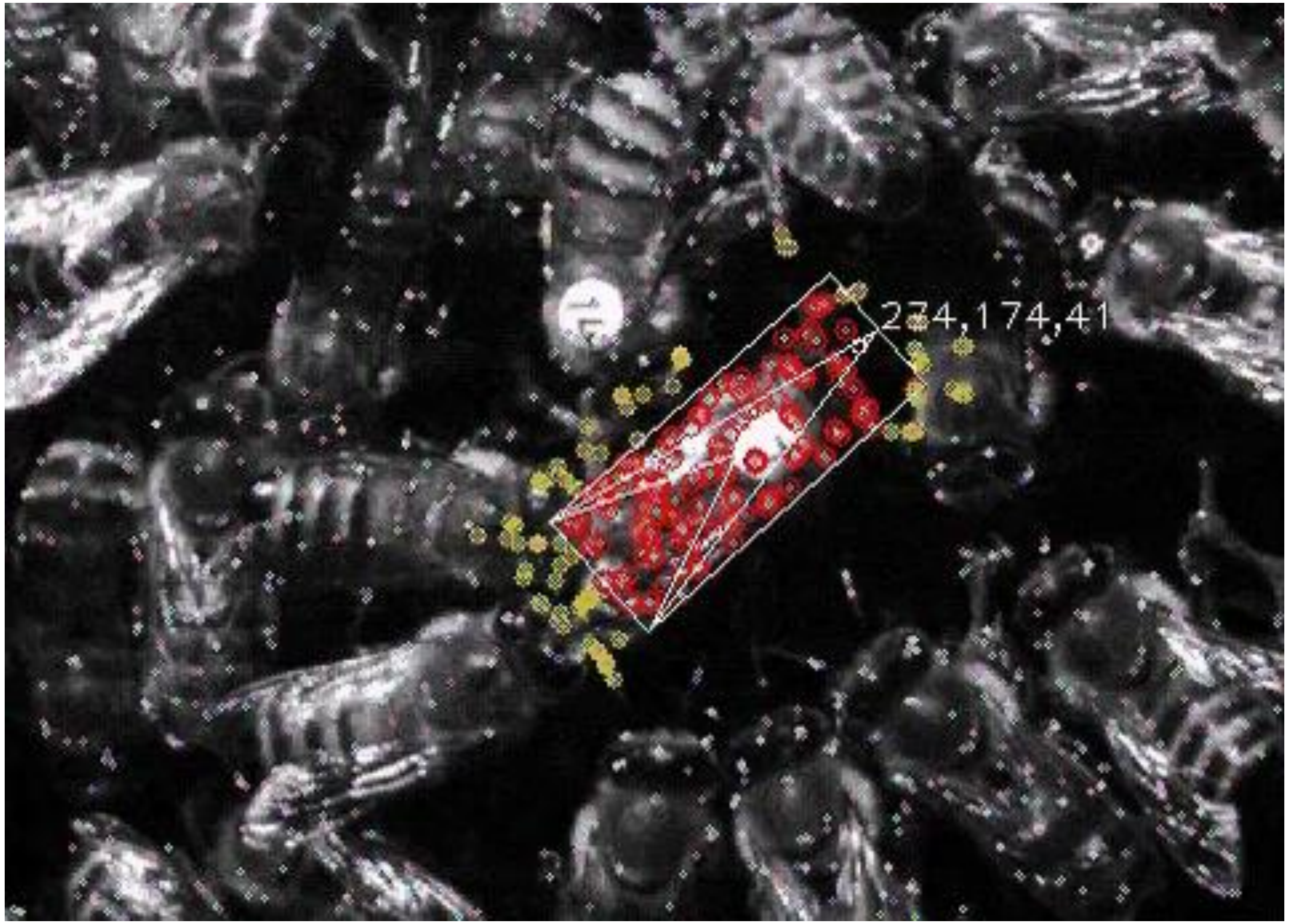
HT for tracking antennal behavior





Hussaini, S.A., Bogusch, L., Landgraf, T. and Menzel, R., 2009. Sleep deprivation affects extinction but not acquisition memory in honeybees. *Learning & memory*, *16*(11), pp.698-705.

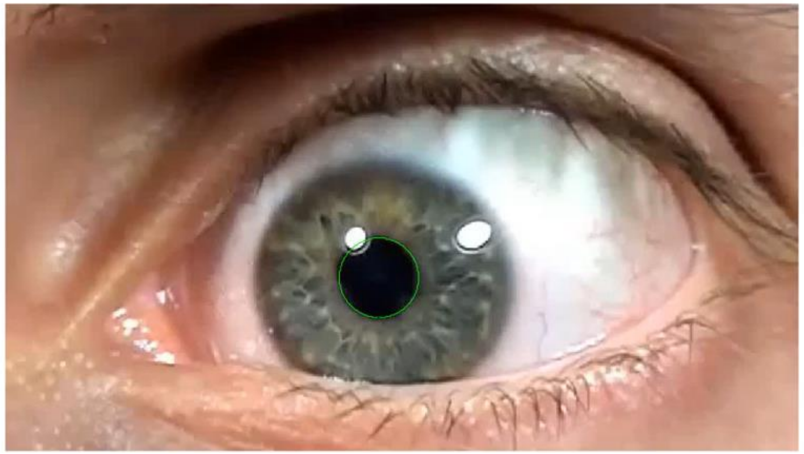
HT for finding rigid transform parameters

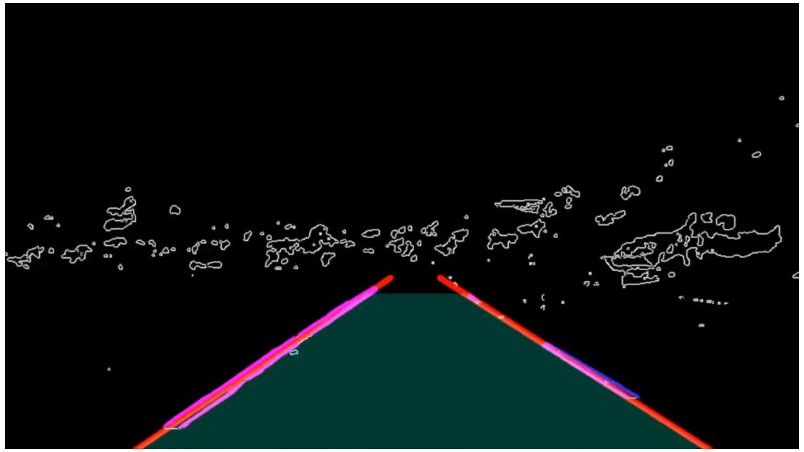


Landgraf, T. and Rojas, R., 2007. Tracking honey bee dances from sparse optical flow fields.

Ex. 3.1: detect and track lanes or eyes with HT

- implement the HT for lines OR circles
- in the assignment notebook you‘ll find links to two image sequences, use your line finder for the lane data, the circle finder for the eye data





https://www.youtube.com/watch?v=rWTTTGRN6jw

https://www.youtube.com/watch?v=WsWCAi7tkv8

The Generalized Hough Transform

Pattern Recognition Vol. 13, No. 2, pp. 111–122, 1981.
Printed in Great Britain.

0031-3203/81/020111-12 $02.00/0
Pergamon Press Ltd.

GENERALIZING THE HOUGH TRANSFORM TO DETECT ARBITRARY SHAPES*

D. H. BALLARD

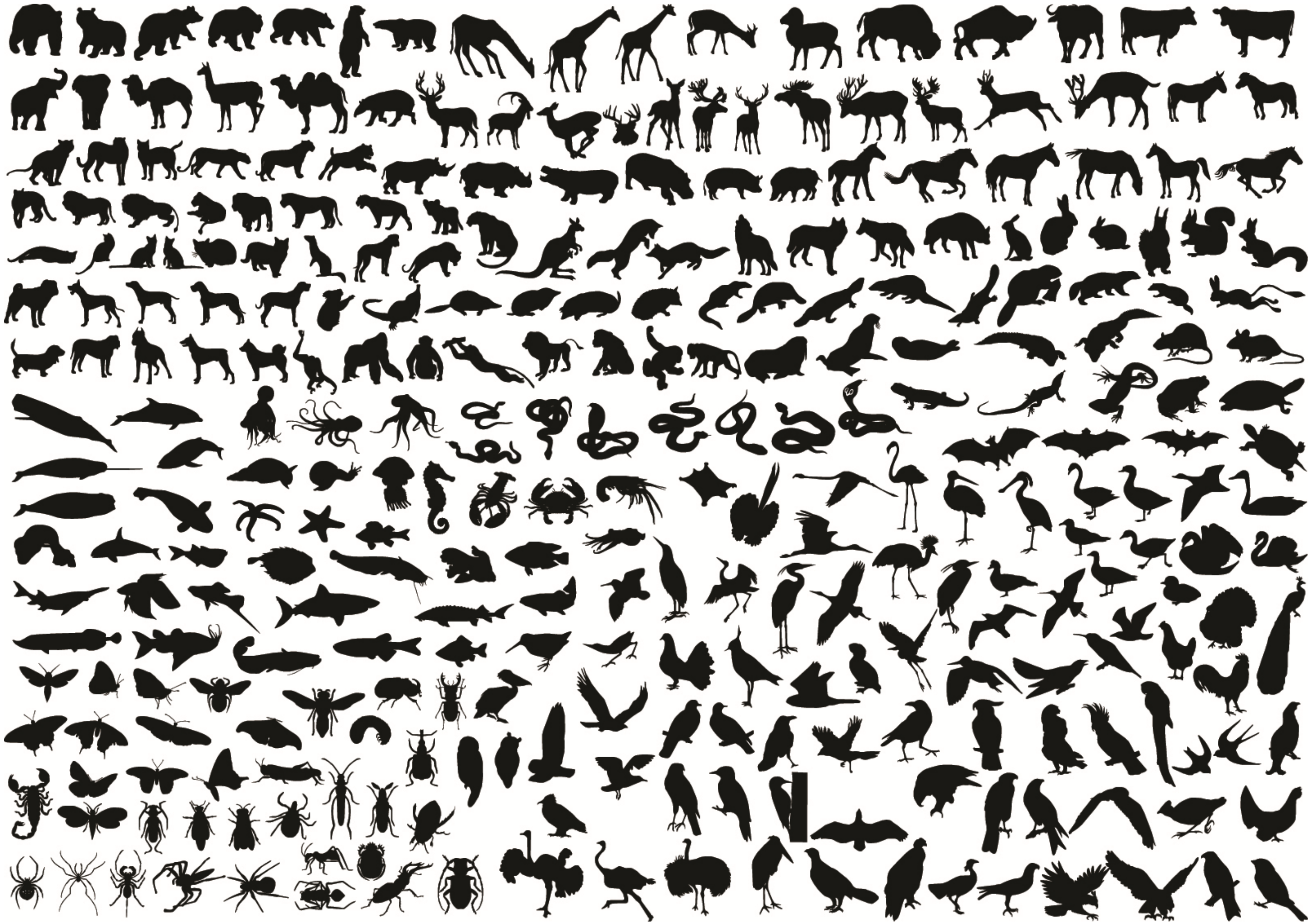
Computer Science Department, University of Rochester, Rochester, NY 14627, U.S.A.

(*Received* 10 *October* 1979; *in revised form* 9 *September* 1980; *received for publication* 23 *September* 1980)

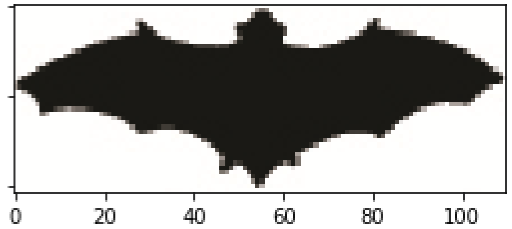
Abstract—The Hough transform is a method for detecting curves by exploiting the duality between points on a curve and parameters of that curve. The initial work showed how to detect both analytic curves[1,2] and non-analytic curves,[3] but these methods were restricted to binary edge images. This work was generalized to the detection of some analytic curves in grey level images, specifically lines,[4] circles[5] and parabolas.[6] The line detection case is the best known of these and has been ingeniously exploited in several applications.[7,8,9]

We show how the boundaries of an *arbitrary* non-analytic shape can be used to construct a mapping between image space and Hough transform space. Such a mapping can be exploited to detect instances of that particular shape in an image. Furthermore, variations in the shape such as rotations, scale changes or figure–ground reversals correspond to straightforward transformations of this mapping. However, the most remarkable property is that such mappings can be composed to build mappings for complex shapes from the mappings of simpler component shapes. This makes the generalized Hough transform a kind of universal transform which can be used to find arbitrarily complex shapes.

Image processing　Hough transform　Shape recognition　Pattern recognition
Parallel algorithms



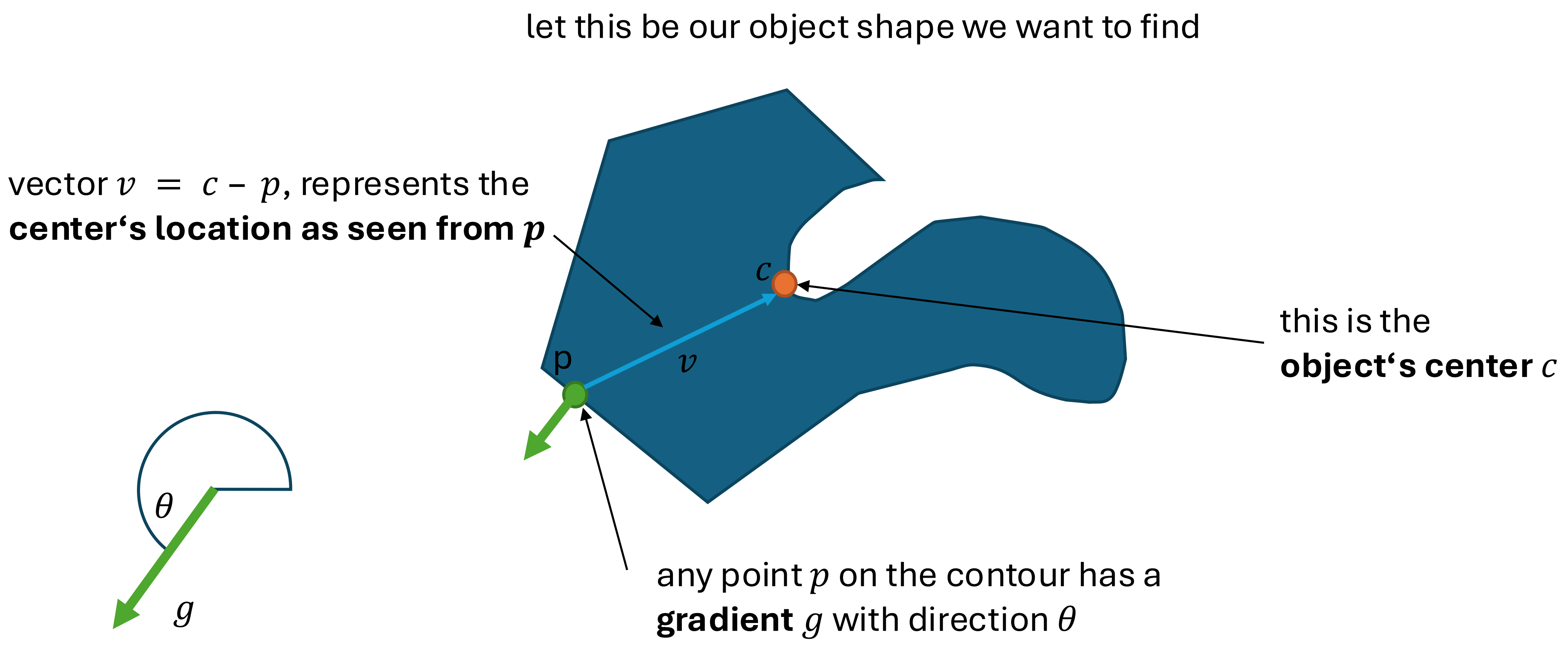
Find the bat!



template

haystack

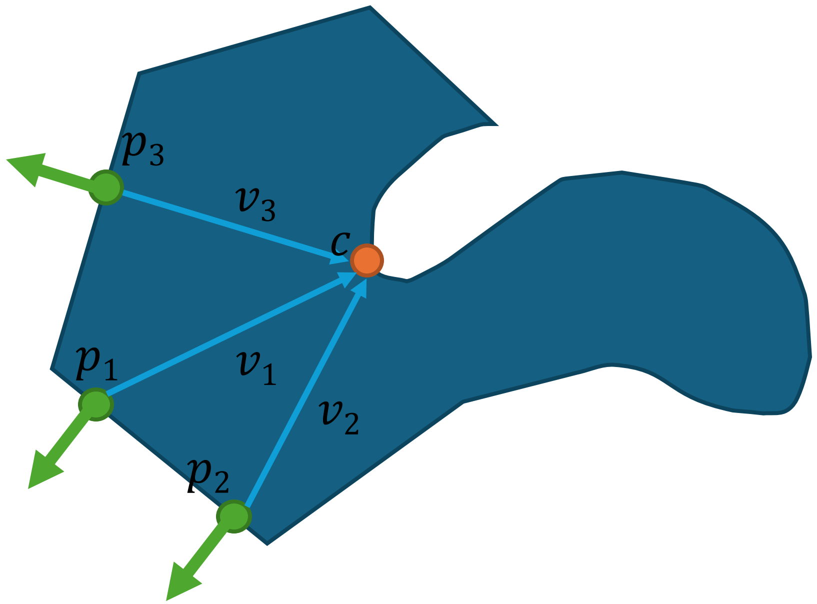
Preprocessing the Template



Preprocessing the Template

R-Table

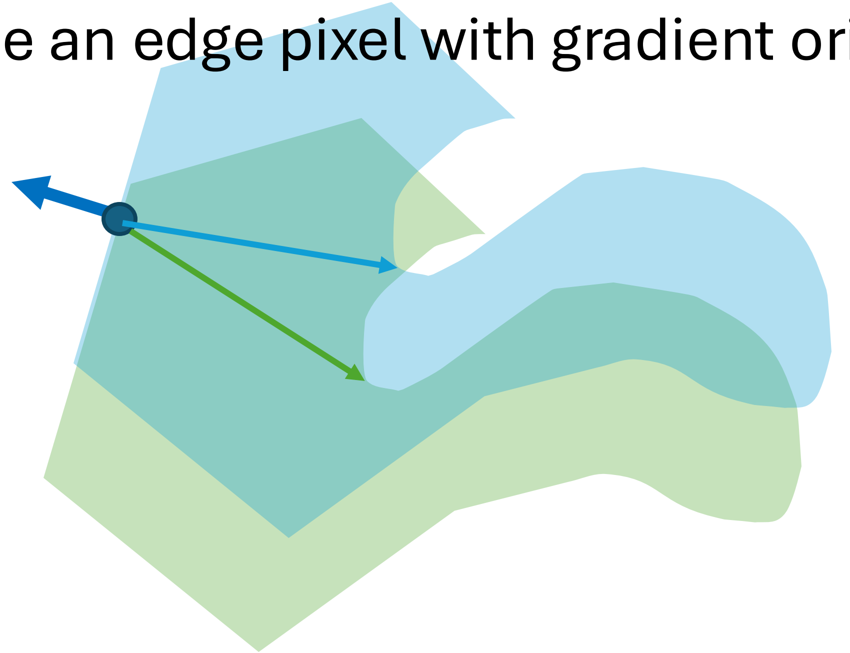
θ	v
0°	v_6
20°	v_1, v_2
40°	v_4
60°	v_3, v_5
80°	…



For all edge pixels p_i in the template: append v_i to $R(\theta_i)$!

Localizing the object

let p_A be an edge pixel with gradient orientation θ

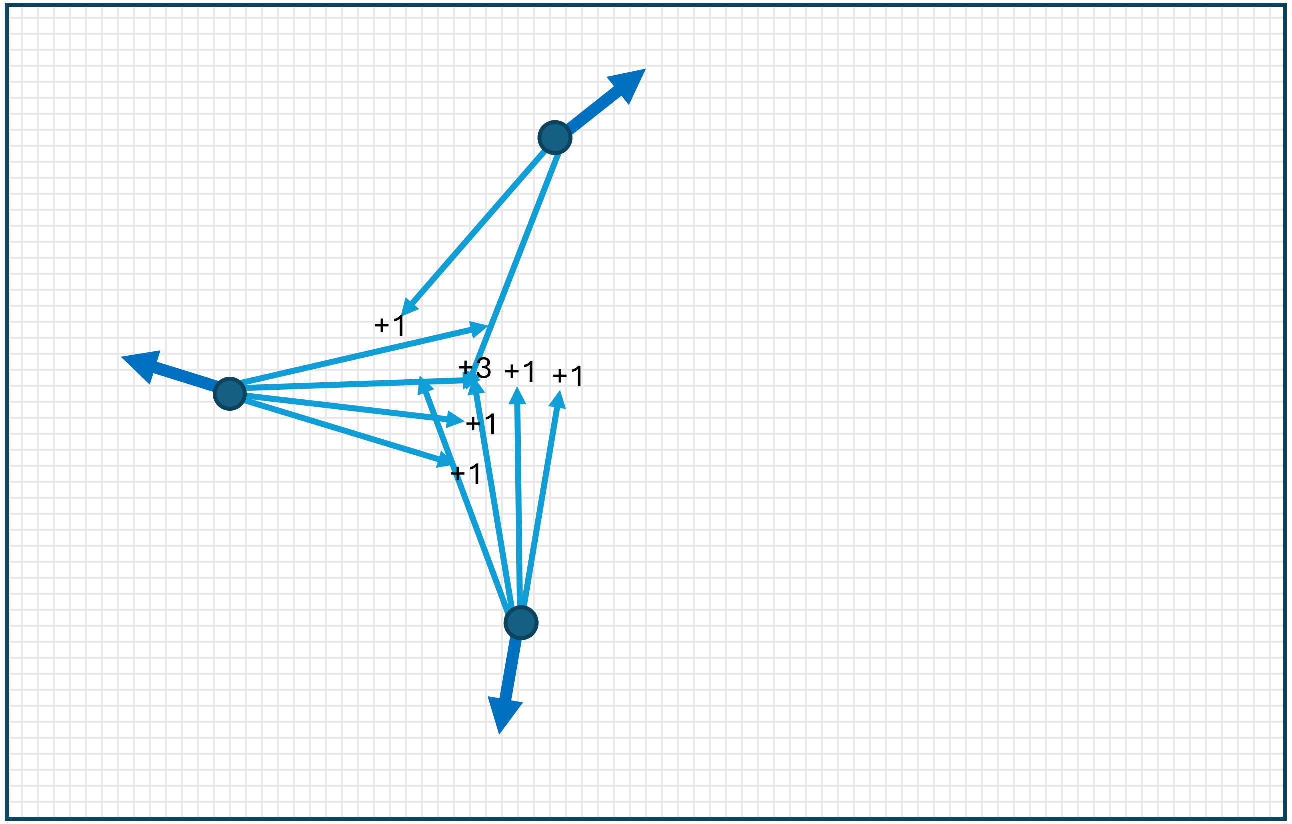


This pixel supports different hypotheses,
it may be any of the ones **with the same gradient**

Localizing the object

θ	v
0°	v_6
20°	v_1, v_2
40°	v_4
60°	v_3, v_5
80°	…

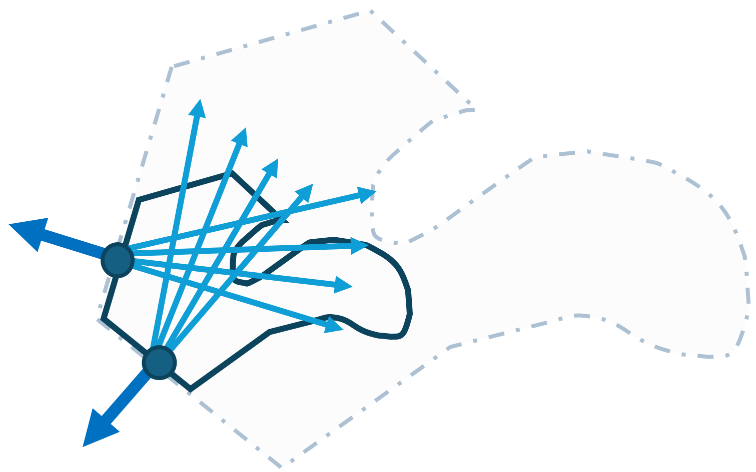
Accumulator



For each edge pixel:
look up the corresponding list of vectors and increment the locations $p + vi$

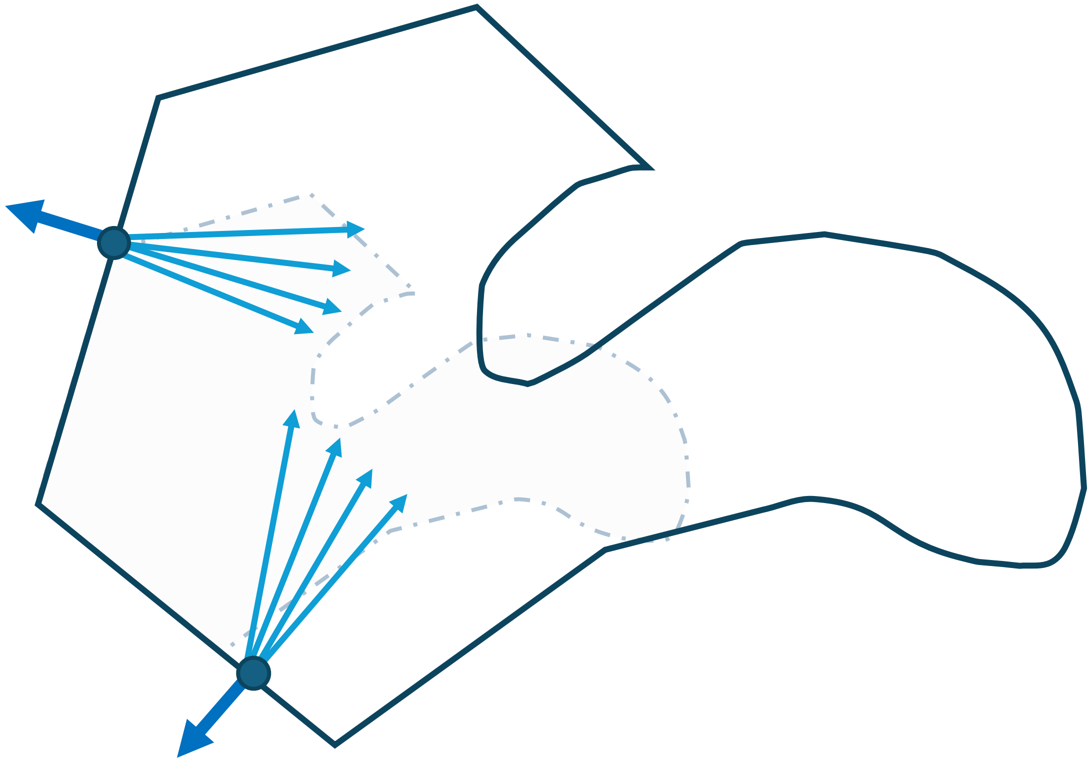
Adding scale-invariance

The algorithm fails when objects appear in different sizes!



No (or wrong) overlap of vectors!

Adding scale-invariance

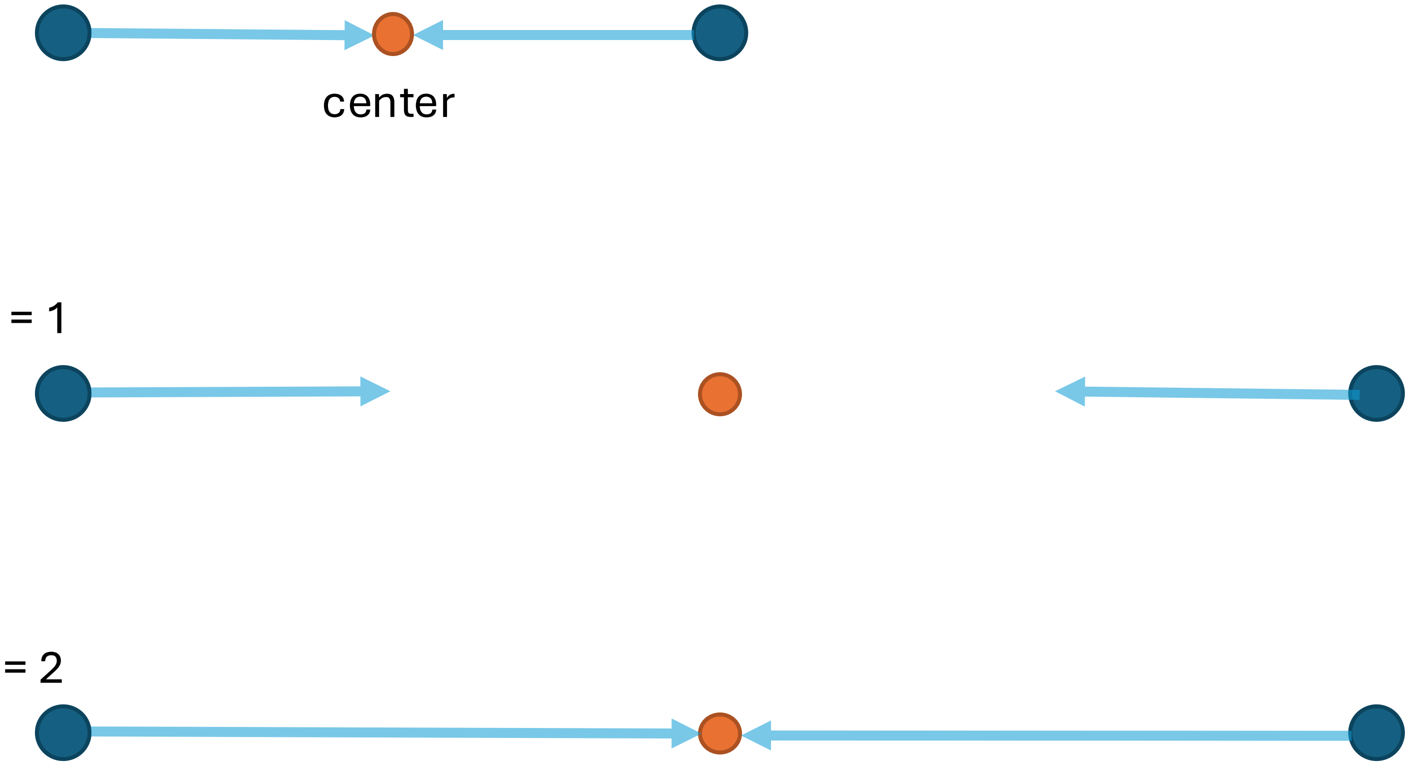


No (or wrong) overlap of vectors!

Adding scale-invariance

Scale: just scale v and introduce a new dimension to the accumulator

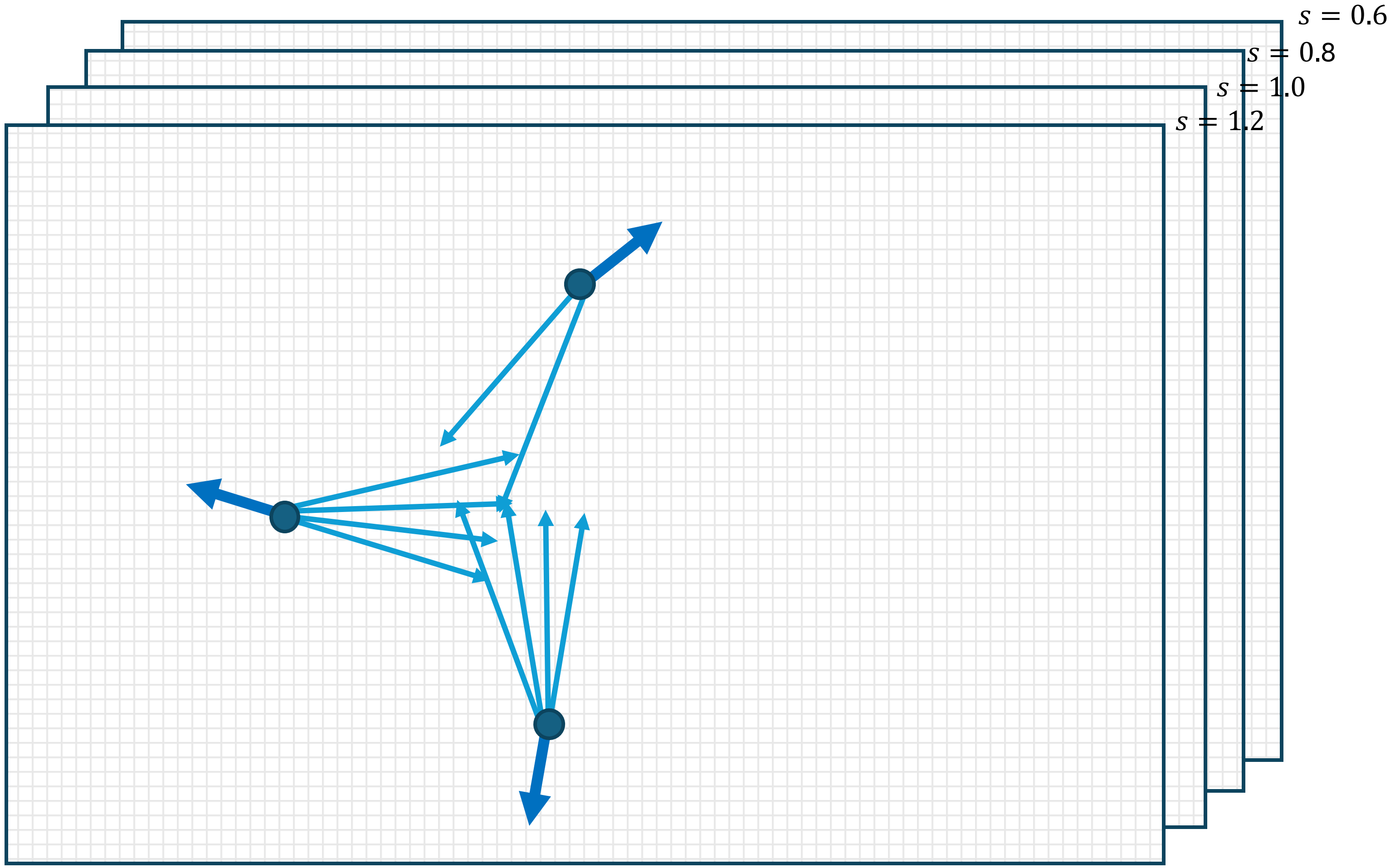
template



object size: 2x
accumulator at scale = 1

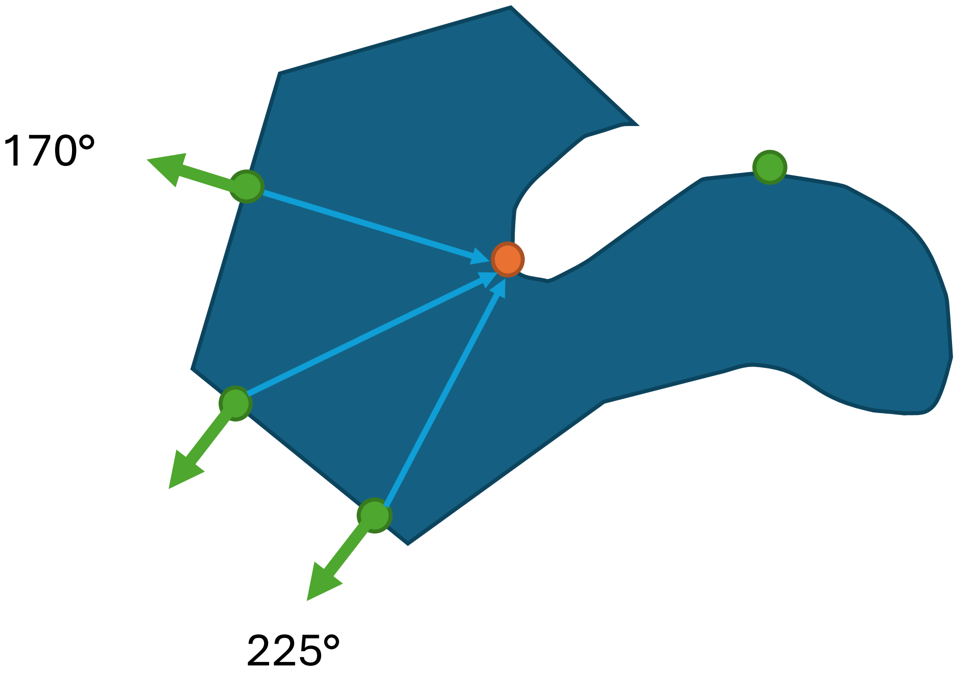
object size: 2x
accumulator at scale = 2

Adding scale-invariance

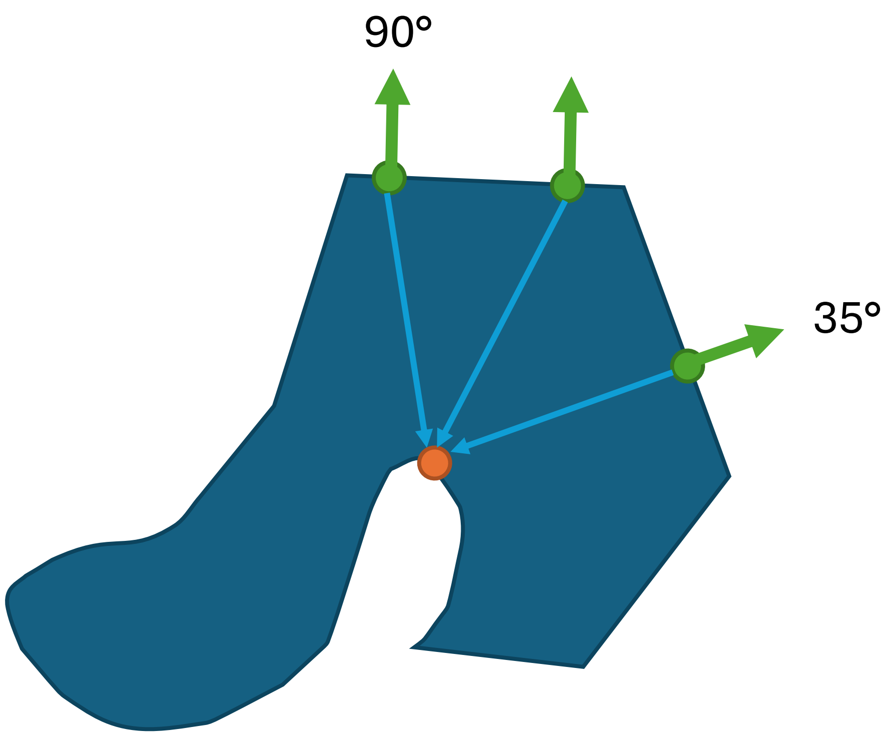


Accumulator is now 3-D!

Adding rotation-invariance

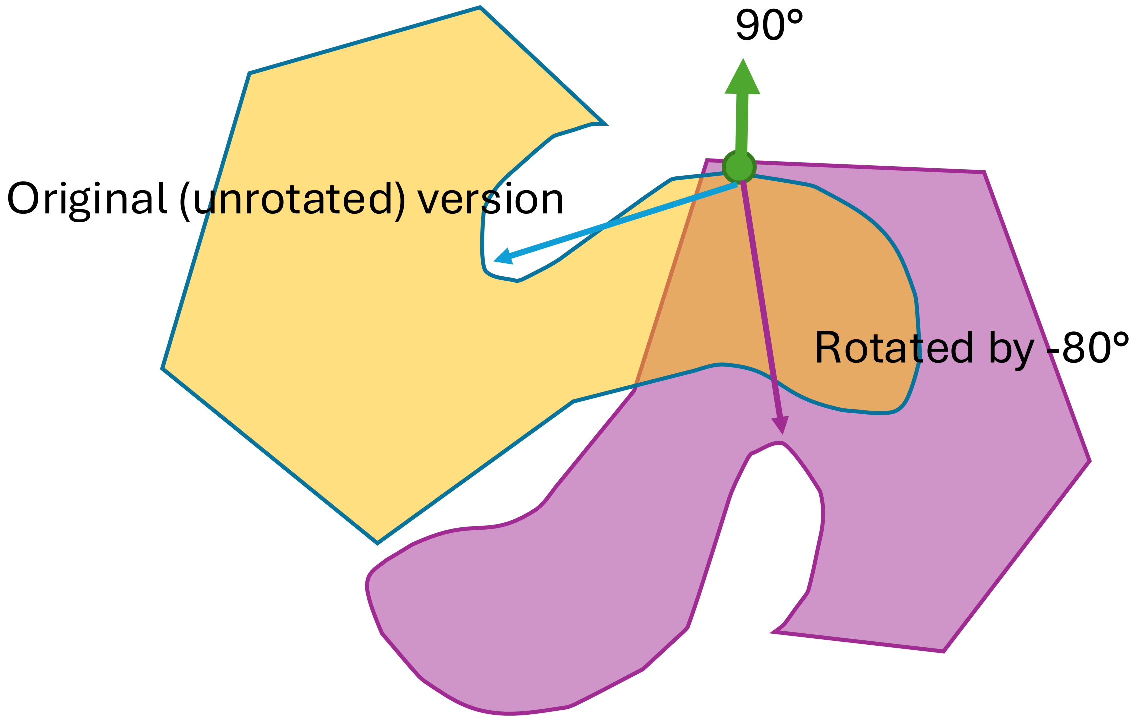


Original (unrotated) version



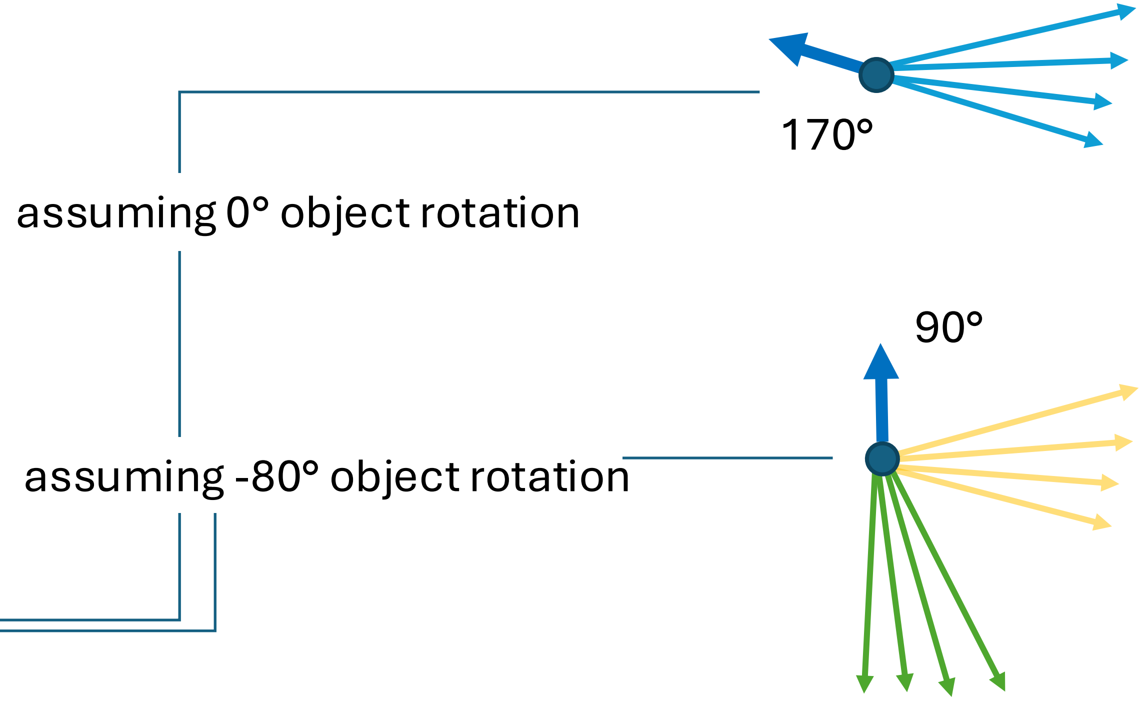
Rotated version

Adding rotation-invariance



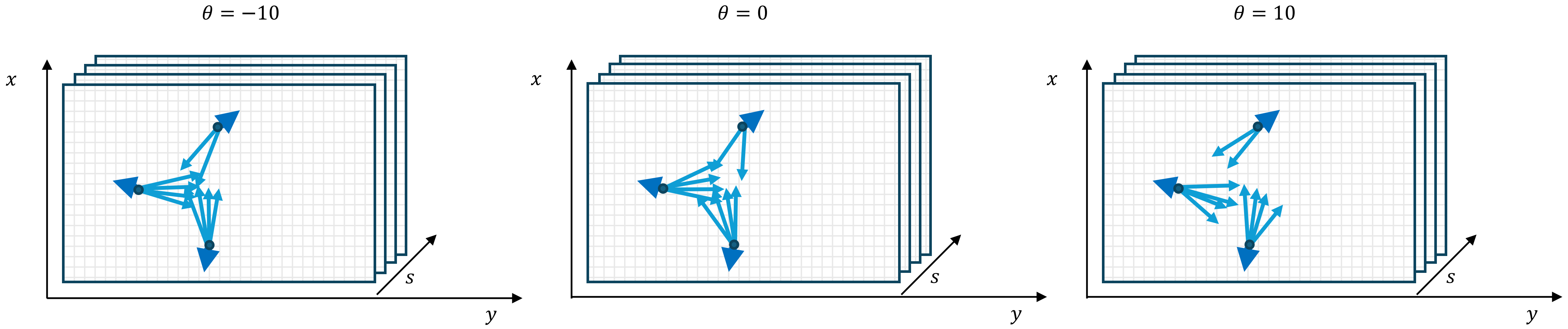
Adding rotation-invariance

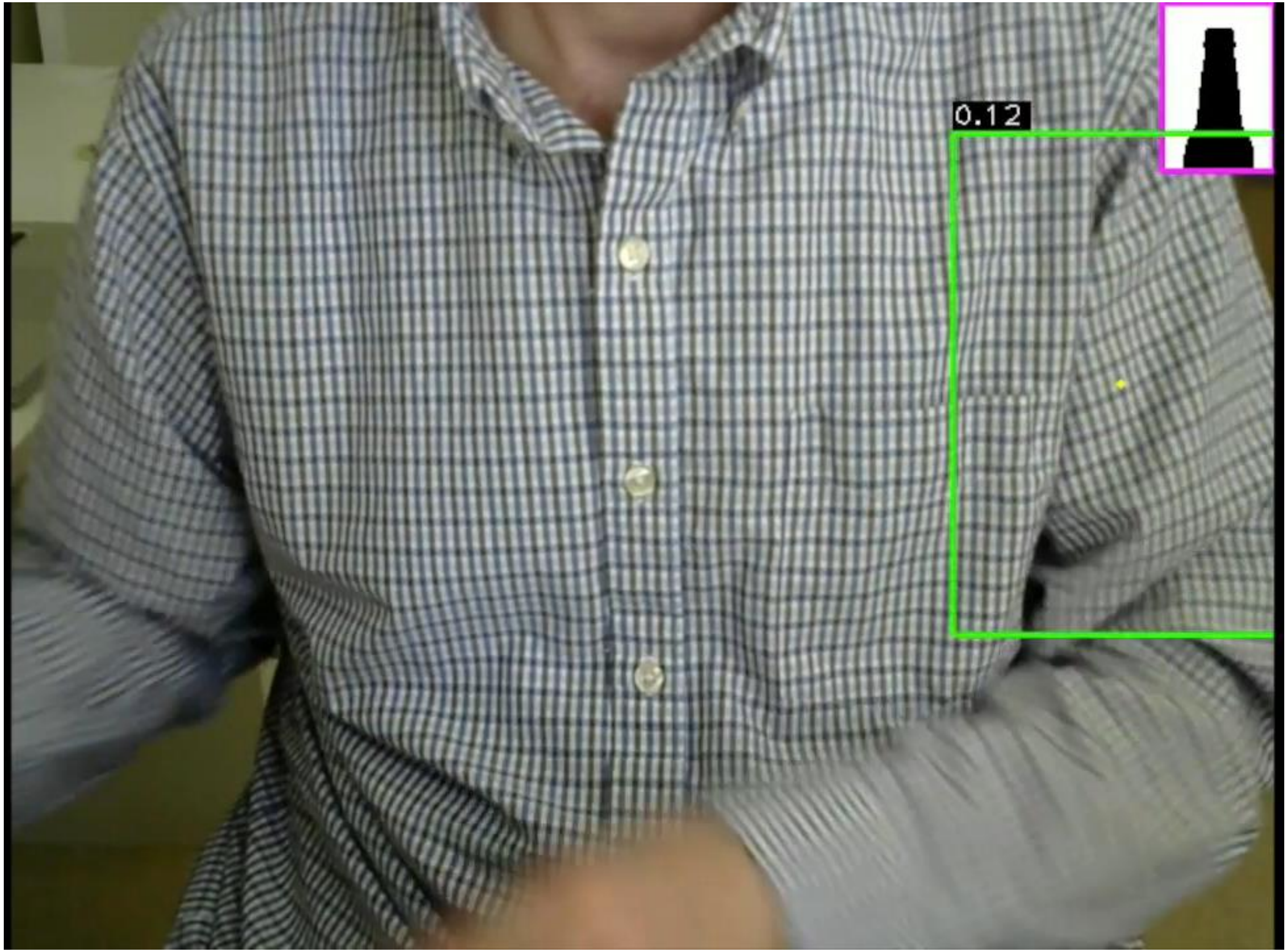
θ	v
80°	
90°	
100°	
…	
160°	
170°	
180°	
190°	



Adding rotation-invariance

Accumulator is now 4-D!

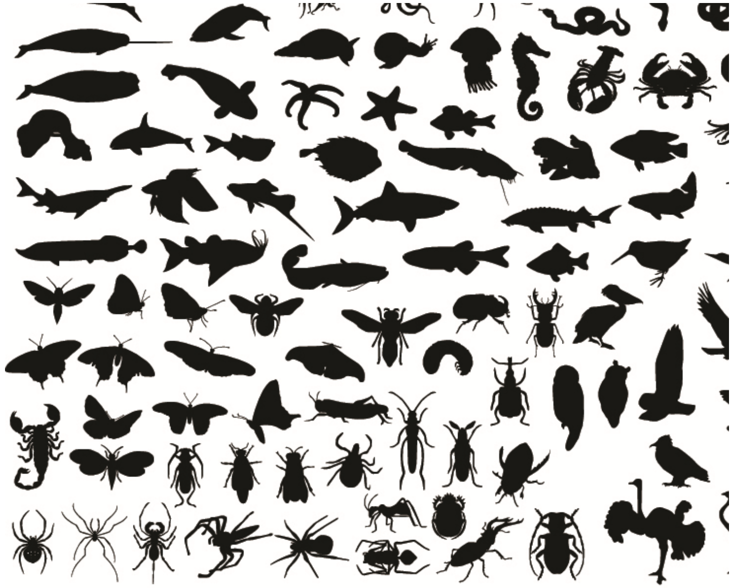




https://www.youtube.com/watch?v=heNQ9mr__L8

Ex. 3.2: find complex shapes with GHT

- implement the GHT
- test it on the animals dataset
- test it on the europe dataset



Summary

- Classical Hough Transform
  - find shapes for which we have an analytic, parametric form
  - the more parameters, the more dimensions of the accumulator
    - →needs more time and space
  - very robust!
- Generalized Hough Transform
  - find more complex shapes, no need for parametric model
  - at least 2-D, at max 4-D accumulator