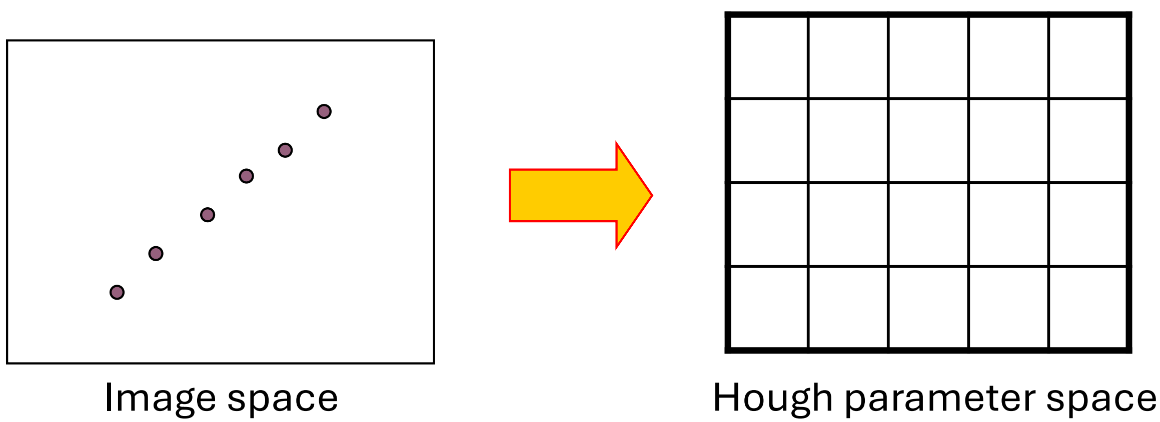
The Hough transformation

D.A. Forsyth

University of Illinois at Urbana-Champaign

Hough transform

- Possibly the earliest voting scheme – but still useful!
  - Discretize parameter space into bins
  - For each feature point in the image, put a vote in every bin in the parameter space that could have generated this point
  - Find bins that have the most votes



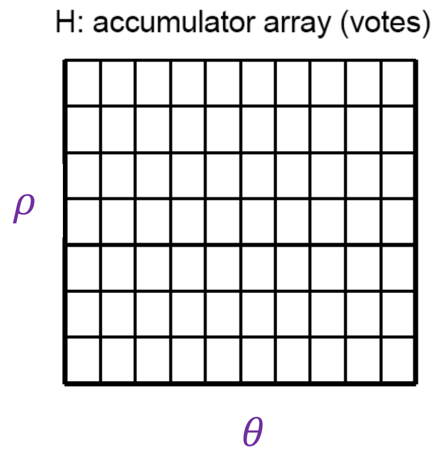
P.V.C. Hough, *Machine Analysis of Bubble Chamber Pictures*, Proc. Int. Conf. High Energy Accelerators and Instrumentation, 1959

Algorithm outline

- Initialize accumulator H to all zeros
- For each feature point (x, y)
  - For $\theta = 0$ to 180

$$\rho = x \cos\theta + y \sin\theta$$

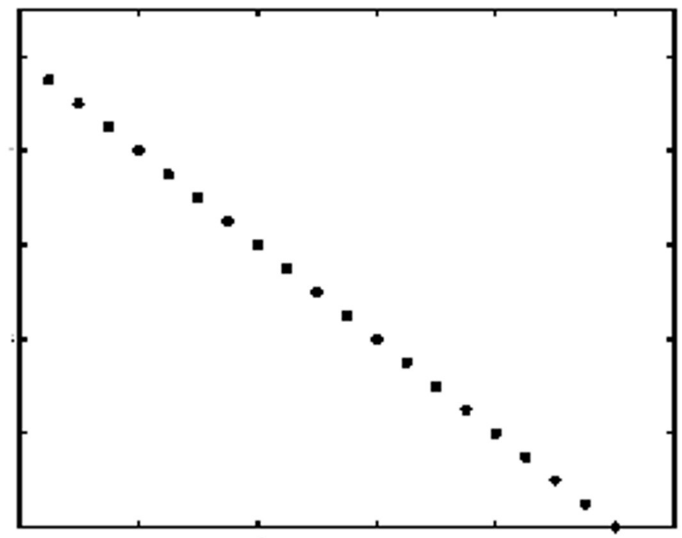
$$H(\theta, \rho) \mathrel{+}= 1$$



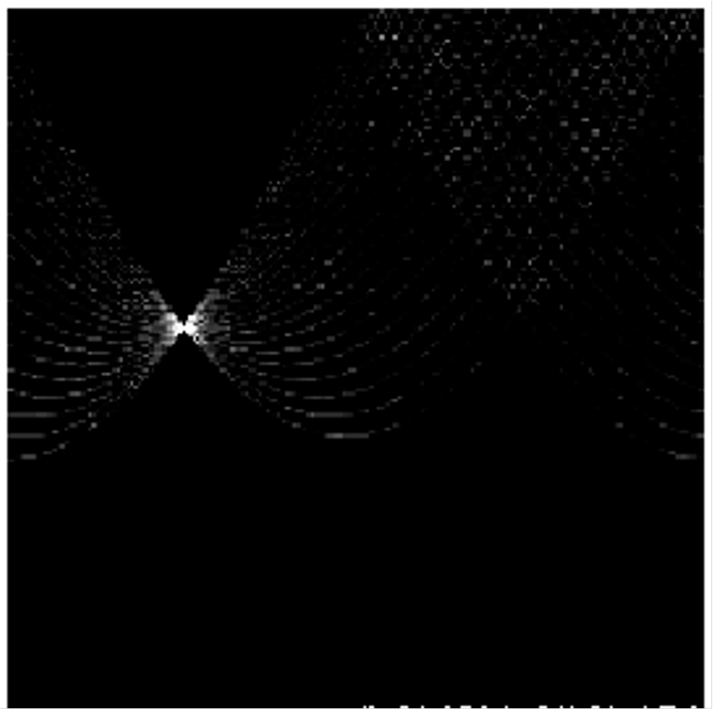
- Analyze array
  - Find (θ, ρ) where $H(\theta, \rho)$ is a local maximum
  - The detected line in the image is given by

$$\rho = x \cos\theta + y \sin\theta$$

Basic illustration



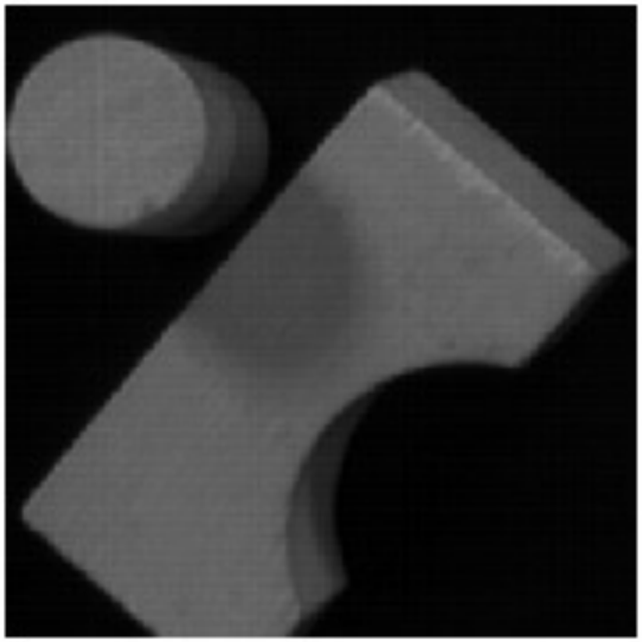
features

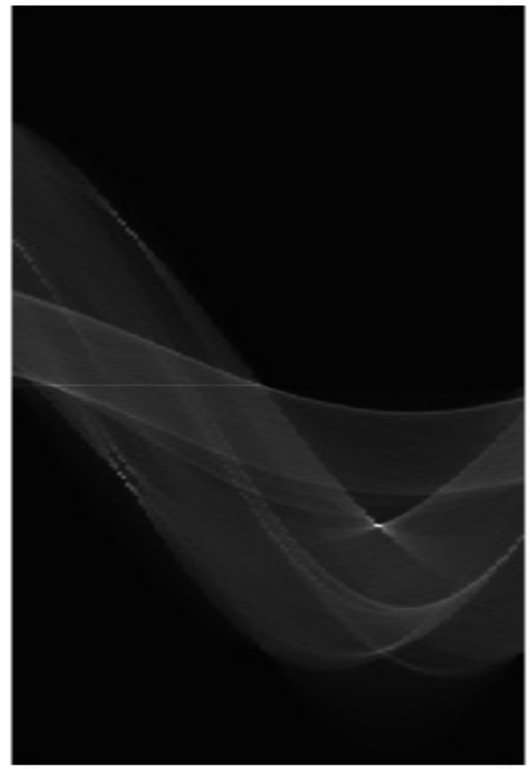


votes

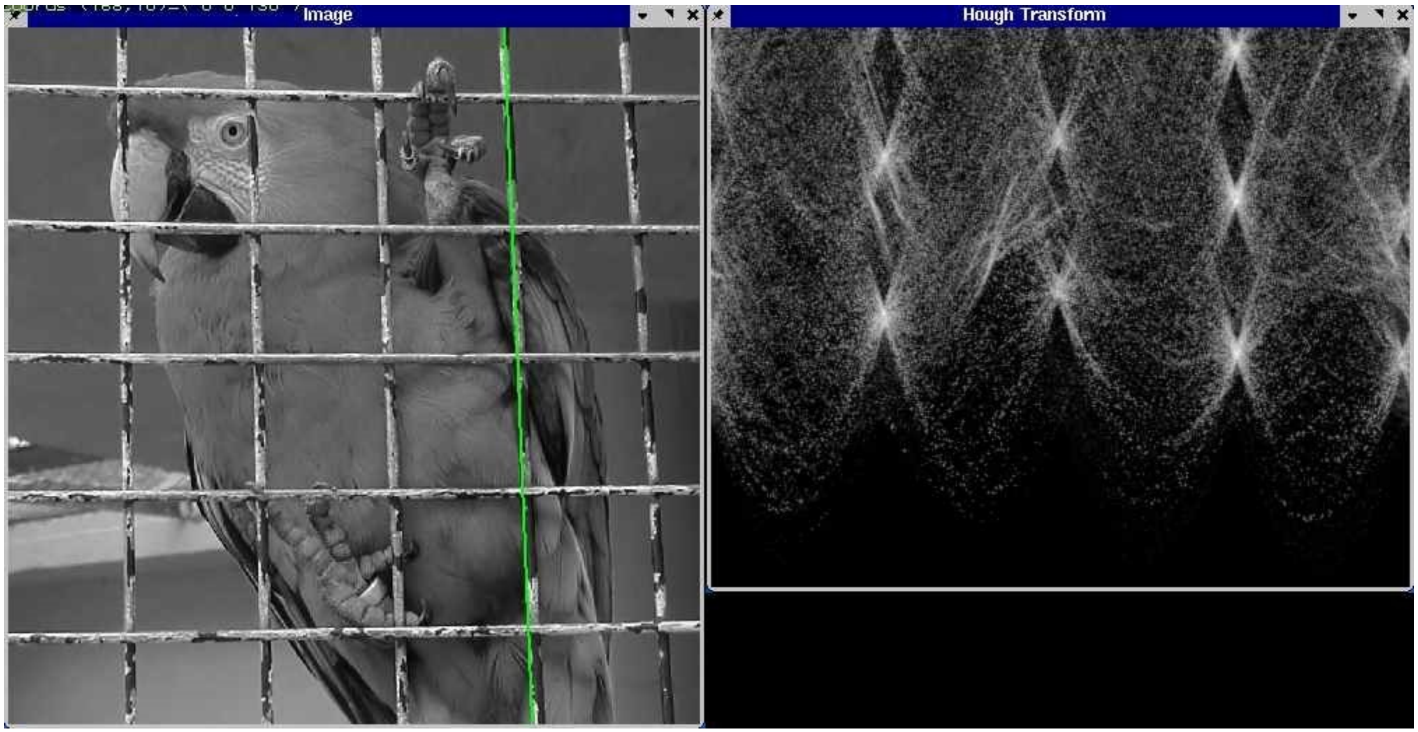
Hough transform demo

Several lines





A more complicated image

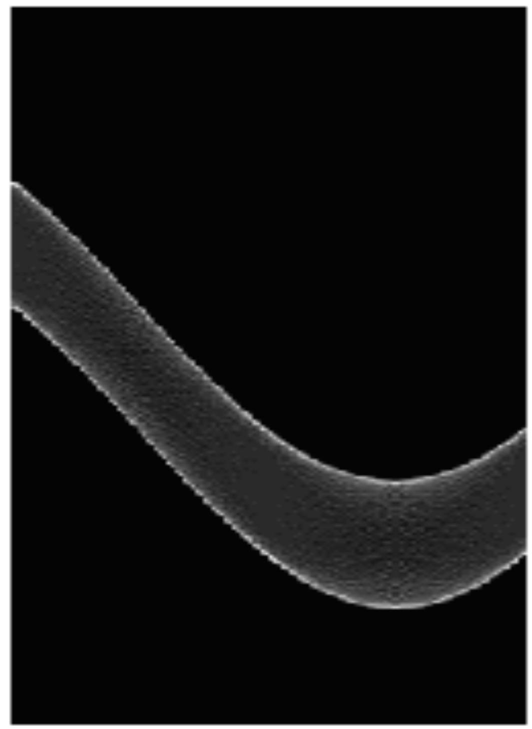


Source

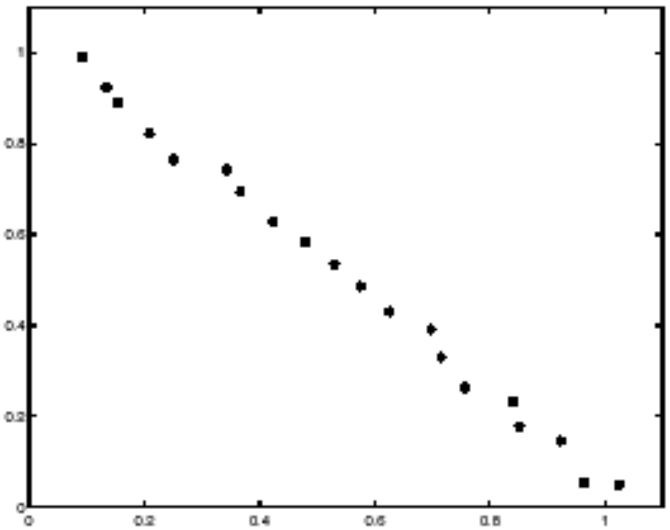
Other shapes

Square

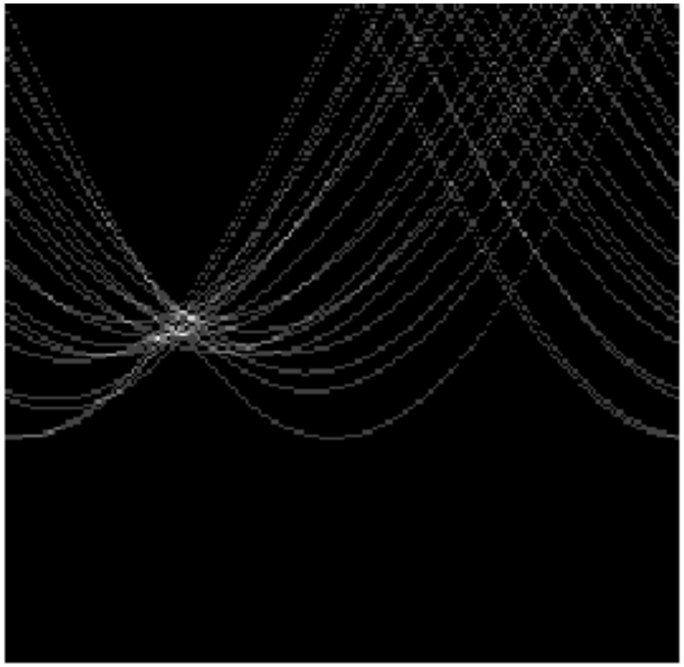
Circle



Effect of noise



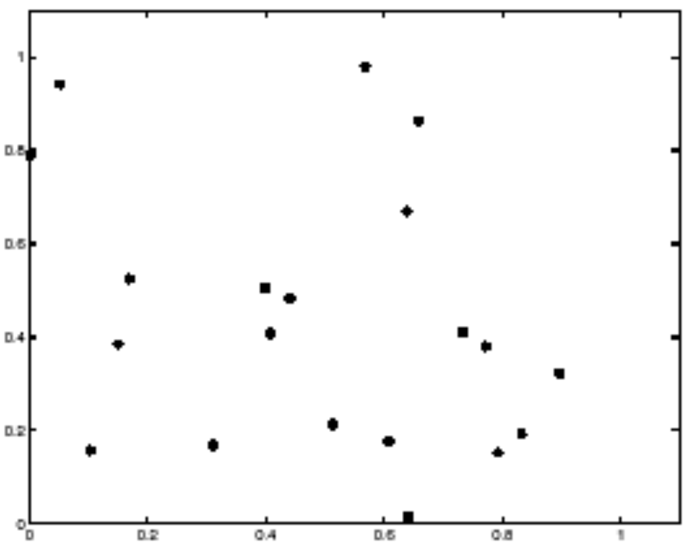
features



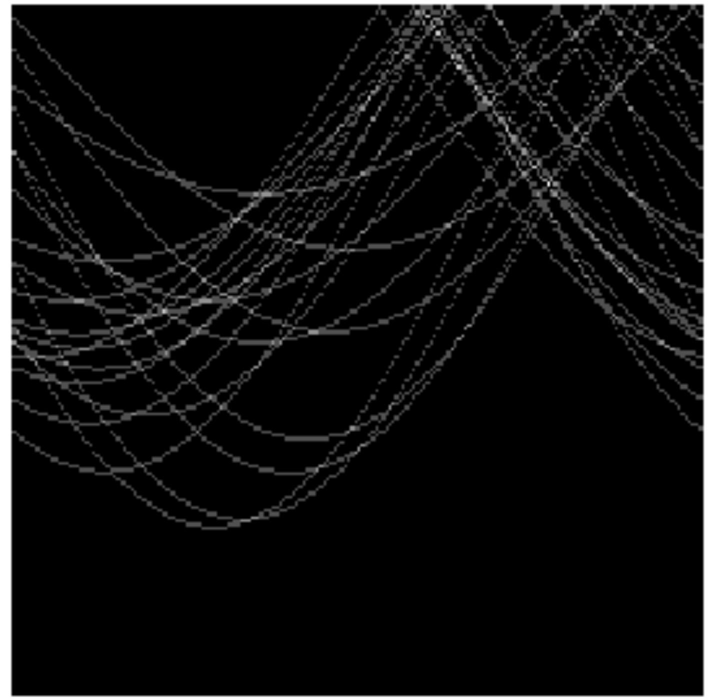
votes

- Peak gets fuzzy and hard to locate

Effect of outliers



features



votes

- Uniform noise can lead to spurious peaks in the array

Dealing with noise

- How to choose a good grid discretization?
  - **Too coarse:** large votes obtained when too many different lines correspond to a single bucket
  - **Too fine:** miss lines because some points that are not exactly collinear cast votes for different buckets
- Increment neighboring bins (smoothing in accumulator array)
- Try to get rid of irrelevant features
  - E.g., take only edge points with significant gradient magnitude

Hough transform: Pros and cons

- Pros
  - Can deal with non-locality and occlusion
  - Can detect multiple instances of a model
  - Some robustness to noise: noise points unlikely to contribute consistently to any single bin
  - Leads to a surprisingly general strategy for shape localization (more on this next)
- Cons
  - Complexity increases exponentially with the number of model parameters – in practice, not used beyond three or four dimensions
  - Non-target shapes can produce spurious peaks in parameter space
  - It’s hard to pick a good grid size

Things to think about…

13.6. Imagine you wish to fit circles of fixed, known radius with a hough transform. What is the dimension of the accumulator array?

13.7. Imagine you wish to fit ellipses with a hough transform. What is the dimension of the accumulator array?

13.8. Imagine you wish to fit spheres with a hough transform. What is the dimension of the accumulator array?

13.9. You wish to fit curves with a hough transform. The accumulator array is d dimensional. You want to have n bins along each axis. How many bins are there in total? What problems might occur if d is big?