Hough Transform for Curve Detection

Soumya Sara John
SC17M048
M.Tech (DSP)
IIST,Thiruvanathapuram

Table of Contents

Introduction

- Curve tracing and Curve detection are important operations in computer vision
- Useful in object detection,tracking,etc.
- Involves tracing the curve completely from the image and classifying the shape

Classical Hough Transform

- Applicable when little is known about the location of the boundary
- Shape of the curve can be defined using some parametric equation
- Using the parametric equation, the centre of the curve can be found out by voting and the curve can be traced based on this centre

Straight Line Detection

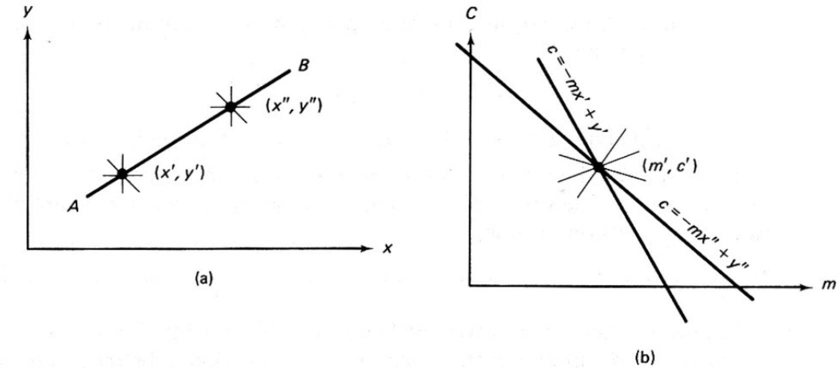

Fig 1: Straight Line in (a)Cartesian Space (b) Parameter (m,c) space

Straight Line Detection

- Equation of line (fig 1(a)):

$$y' = mx' + c$$

$$y'' = mx'' + c$$

- In parameter space(fig 1(b)):

$$c = y' - mx'$$

$$c = y'' - mx''$$

Straight Line Detection- Algorithm

1. Quantify parameter space between the proper maximum and minimum values for c and m
2. Form accumulator array A(c,m), initialised to all zeroes
3. Find the gradient of the image
4. For each (x,y) with gradient magnitude greater than some threshold,
    4.1 For each m,calculate $c = mx + y$
    4.2 $A(c, m) = A(c, m) + 1$
5. Find the index of the local maxima of A to get a and m

Straight Line Detection

- $c = \infty$ when the line is perpendicular
- Hence, use (r, θ) space or polar space

$$xcos\theta + ysin\theta = r$$

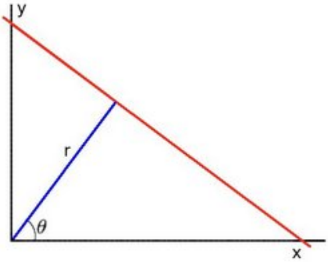

Fig 2: Polar space

Straight Line Detection- Algorithm

1. Quantify parameter space between the proper maximum and minimum values for r and θ
2. Form accumulator array A(r,θ), initialised to all zeroes
3. Find the gradient of the image
4. For each (x,y) with gradient magnitude greater than some threshold,
   4.1 For the corresponding θ,calculate $r = xcos\theta + ysin\theta$
   4.2 $A(r, \theta) = A(r, \theta) + 1$
5. Find the index of the local maxima of A to get r and θ

Circle Detection

- Three parameters present - (a,b,r)
- Equation of a circle with centre (a,b) and radius r

$$(x-a)^2 + (y-b)^2 = r^2$$

- Then the parameters are:

$$a = x - \sqrt{r^2 - (y-b)^2}$$

$$b = y - \sqrt{r^2 - (x-a)^2}$$

- Computationally complex

Circle Detection

- Using gradient information also
- $$x = a + rsin\phi => a = x - rsin\phi$$
  $$y = b + rcos\phi => b = y - rcos\phi$$

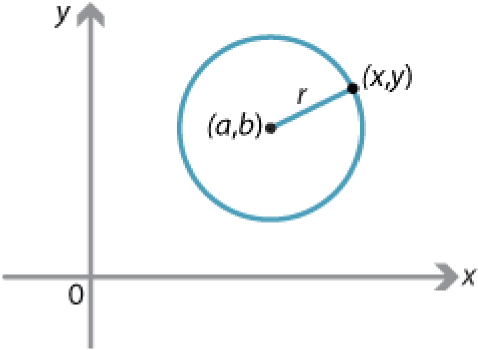

Fig 3: Circle

Circle Detection- Algorithm

1. Quantify parameter space between the proper maximum and minimum values for r,a and b
2. Form accumulator array A(a,b,r), initialized to all zeroes
3. Find the gradient of the image
4. For each (x,y) with gradient magnitude greater than some threshold,
   4.1 For the corresponding gradient angle ϕ and for each r,calculate $a = x - rsin\phi$ and $b = y - rcos\phi$
   4.2 $A(a, b, r) = A(a, b, r) + 1$
5. Find the index of the local maxima of A to get a,b,r

Ellipse Detection

- Using gradient information
- Equation of an ellipse with centre (x_0, y_0), major axis a and minor axis b:

$$\frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1$$

- $$x = x_0 \pm \frac{a}{\sqrt{1 + \frac{b^2}{a^2 tan^2 \phi}}}$$

  $$y = y_0 \pm \frac{b}{\sqrt{1 + \frac{a^2 tan^2 \phi}{b^2}}}$$

- 4 parameters present

Ellipse Detection- Algorithm

1. Quantify parameter space between the proper maximum and minimum values for a , b,x_0, y_0
2. Form accumulator array A(a, b, x_0, y_0), initialized to all zeroes
3. Find the gradient of the image
4. For each (x,y) with gradient magnitude greater than some threshold,
    4.1 For the corresponding gradient angle ϕ and for each a,b ,calculate $x_0 and y_0$
    4.2 A$(a, b, x_0, y_0) =$ A$(a, b, x_0, y_0) + 1$
5. Find the index of the local maxima of A to get a, b, x_0, y_0

Generalized Hough Transform

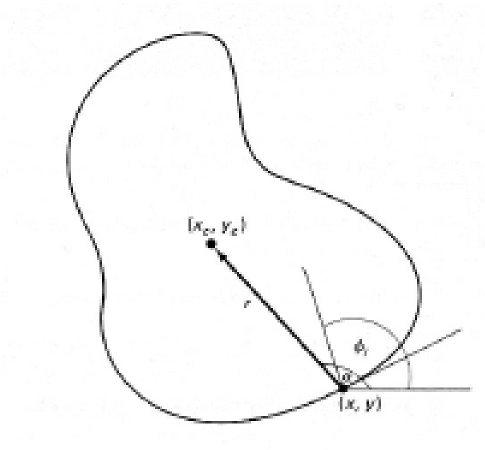

Fig 3: Random shape used to form R-table

Generalized Hough Transform

- Introduced by Dana.H.Ballard in 1981
- Choose a reference image for the required shape
- Choose a reference point (x_c, y_c), usually the centroid
- For each edge point (x_i, y_i) , find the Eucleidean distance R from (x_c, y_c) and the angle α
- Update these values as a 2-D array $r_i^j = (R, \alpha)$ corresponding to the gradient ϕ_j
- Do this for all the edge points to create the R-table

R-table

ϕ_1	$\mathbf{r}_1^1, \mathbf{r}_2^1, \ldots, \mathbf{r}_{n_1}^1$
ϕ_2	$\mathbf{r}_1^2, \mathbf{r}_2^2, \ldots, \mathbf{r}_{n_2}^2$
$\vdots$	$\vdots$
ϕ_m	$\mathbf{r}_1^m, \mathbf{r}_2^m, \ldots, \mathbf{r}_{n_m}^m$

Fig 5: R-table

GHT-Algorithm

1. Make the R-table for the shape to be located
2. Form an accumulator array $A(x_{cmin} : x_{cmax}, y_{cmin} : y_{cmax})$ initialized to all zeros
3. For each edge point :
    3.1 Compute $\phi((x_i, y_i))$
    3.2 Calculate
    $$x_c = x + r(\phi)cos(\alpha(\phi))$$
    $$y_c = y + r(\phi)sin(\alpha(\phi))$$
    3.3 Increment $A(x_c, y_c) = A(x_c, y_c) + 1$
4. Find the index of the local maxima of A

GHT

- To include scaling by S and rotation by θ of the shape,just make the following changes:

$$x_c = x + Sr(\phi)cos(\alpha(\phi) + \theta)$$

$$y_c = y + Sr(\phi)sin(\alpha(\phi) + \theta)$$

- $A(x_c, y_c, S, \theta) = A(x_c, y_c, S, \theta) + 1$

HT to detect circle

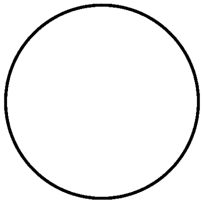

Figure: Original image

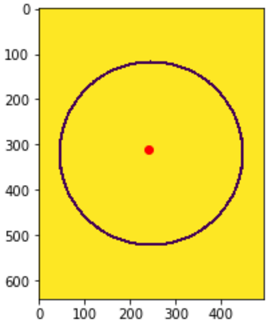

Figure: Centre of circle detected

Fig 6: Circle Detection

GHT to detect square

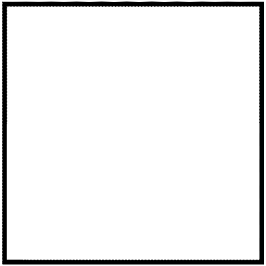

Figure: Original image

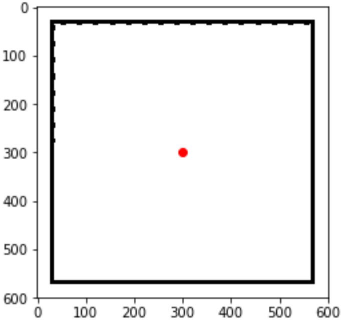

Figure: Centre of square detected

Fig 7: Square Detection

Advantages and Disadvantages

- Advantages:
  - Robust to partial deformation of shapes as maximum voting is used
  - Tolerant to low variance noise
  - Can find multiple occurences of the same shape in the image
- Disadvantages
  - Requires lot of storage for complex shapes

Conclusion

- A method for object recognition involving shape recognition
- Effective in detecting even complex shapes
- Can be used for tracking , object detection , segmentation , etc.

Thank you!