Detection of straight lines in images

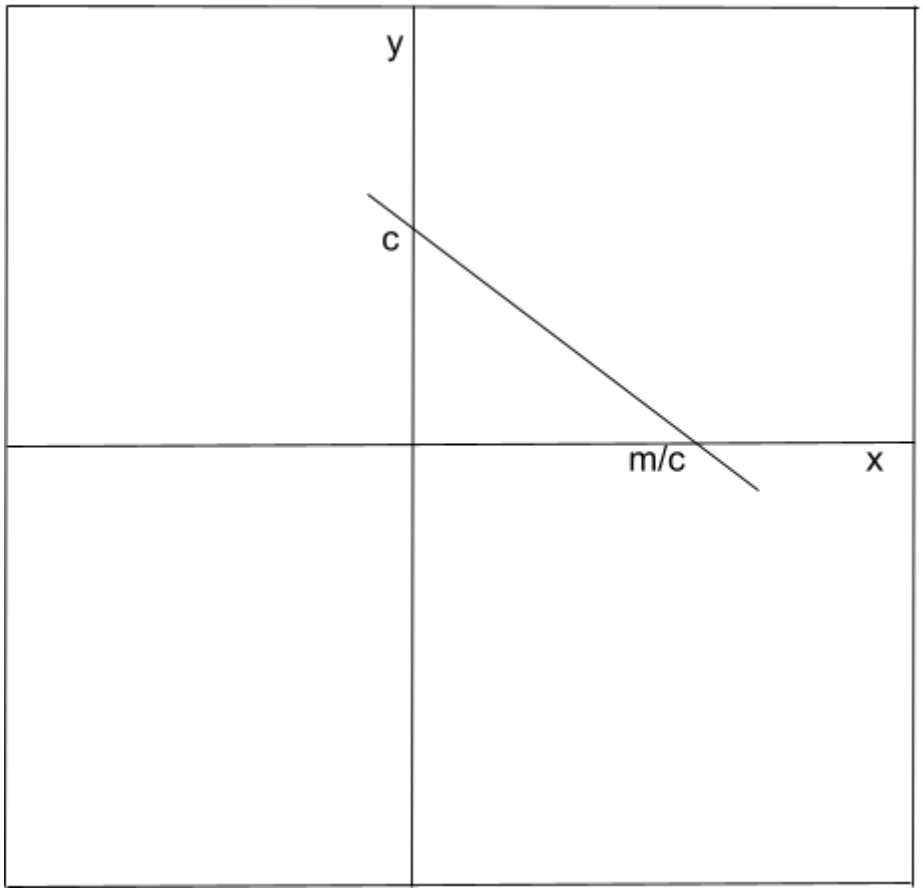
Reference :- https://www.geeksforgeeks.org/line-detection-python-opencv-houghline-method/

Drawing different shapes like lines, rectangles, ellipses etc in an image is quite a straightforward task once the analytic expression for the curve is available. One can generate a set of foreground pixels using the expression and then render it. However, detecting lines, and curves in images is a more involved task.

Analytical expression for a line in x-y plane is given by

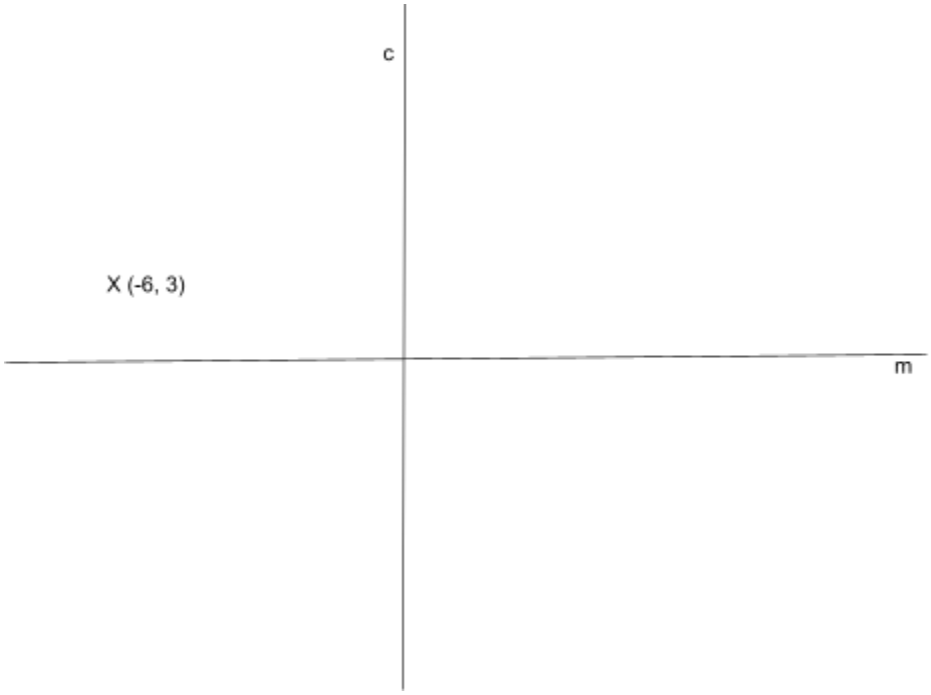
$y = mx + c$

Where *m* is the slope and *c* is y-intercept. For a given *m* and *c,* the set of all (x, y) pairs satisfying the above expression lies on the line.



Parametric Representation of Line

Slope Intercept form



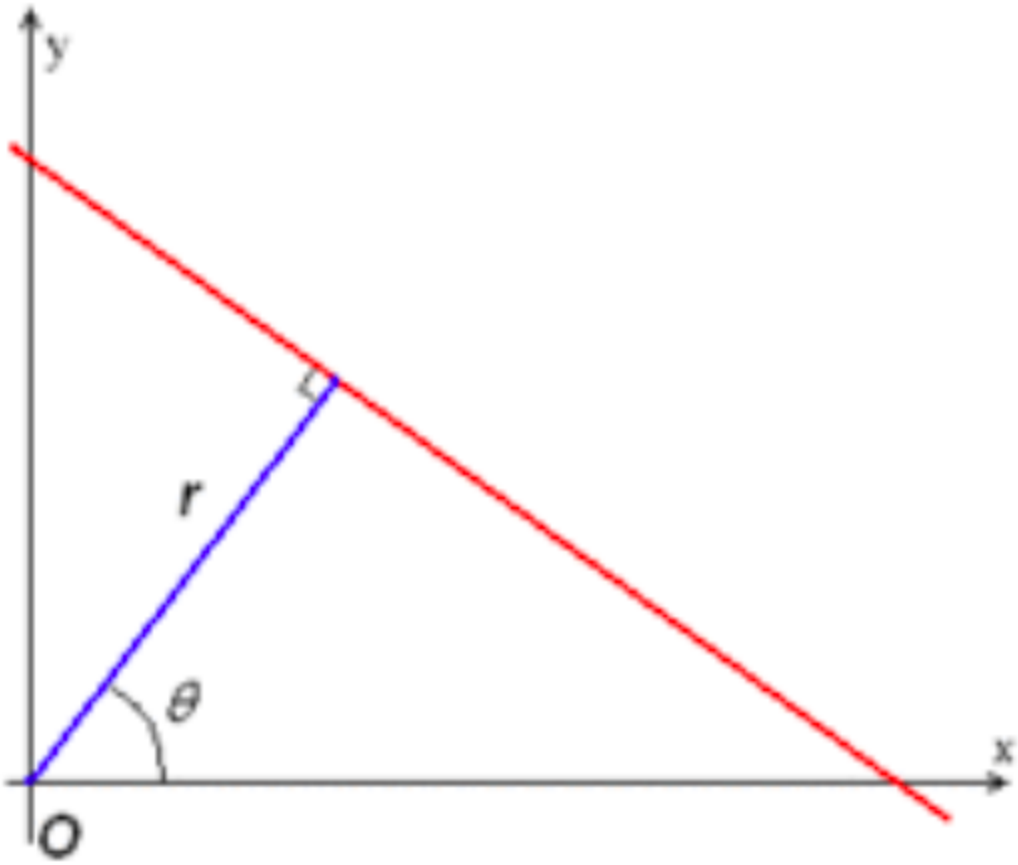
A line in (x,y) space, where x and y are horizontal and vertical coordinates in the image, corresponds to a single point in the parameter space (*m*,*c*).
(https://www.youtube.com/watch?v=p6_TXNUvQtc)

Hessel Normal form

The parameter space of y-intercept and slope (*m, c*) mentioned above has the issue that the slope m is infinity for vertical lines. Hence the Hough transform algorithm is implemented in another parameter space called Hessel normal form. The parameters in this system, r and θ are defined as below

$r = x.\cos(\theta) + y.\sin(\theta)$
where
r - Distance from origin to the closest point on the line
θ - Angle between x axis and the line from origin to closest point on the line

Just like (m,c) space, a point in (r, θ) space represents a line in (x, y) space.


y
r
θ
x
O