CV Lab5 Notes

TA: 郑浩 (RA in SUSTech CV Lab)

Prerequisites

1. Python packages

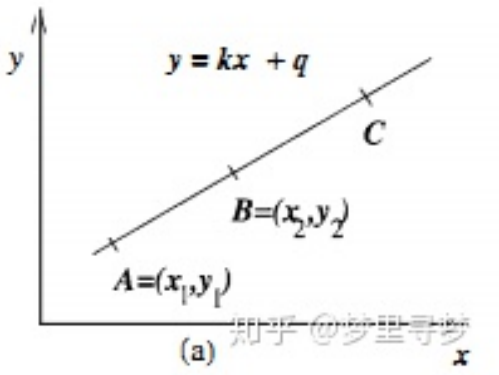
```
conda activate YOUR_ENV
conda install scikit-image pillow numpy matplotlib
```

2. The Chinese Version of Notes can be retrieved in https://zhuanlan.zhihu.com/p/203292567

Hough Transform

Hough Space in Cartesian Coordinates

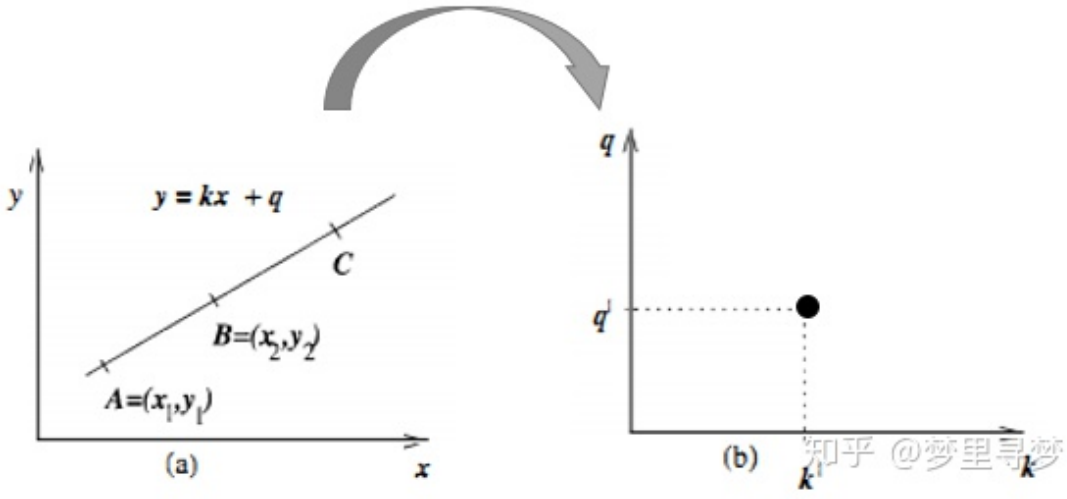
A line can be determined using $A = (x_1, y_1), B = (x_2, y_2)$ in Cartesian coordinates.



in which $y = kx + q$ can be represented using (k, q) in the parameter space:

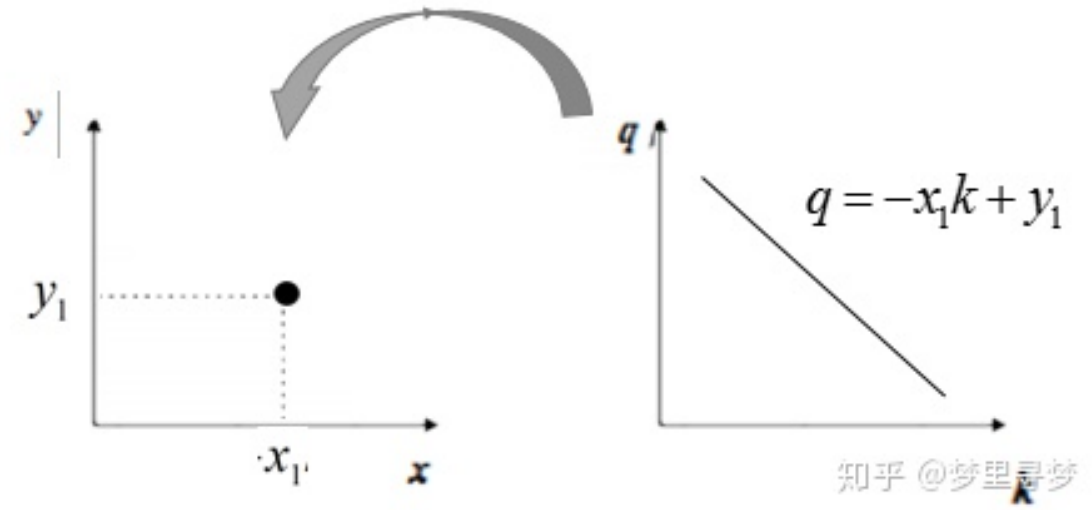
$$\begin{cases} q = -kx_1 + y_1 \\ q = -kx_2 + y_2 \end{cases} \tag{1}$$

The transformation can be visualized:

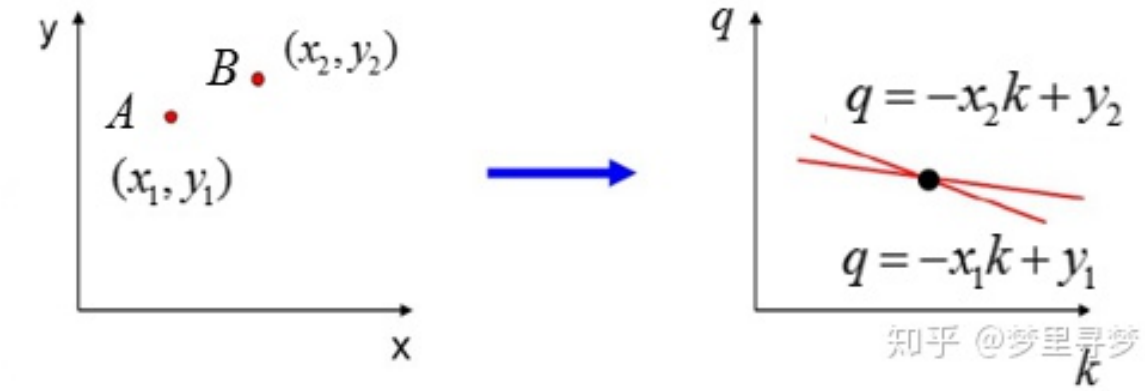


The transformed parameter space is **Hough Space**, in which a **line** in Cartesian coordinates can be represented as a **point** in Hough Space.

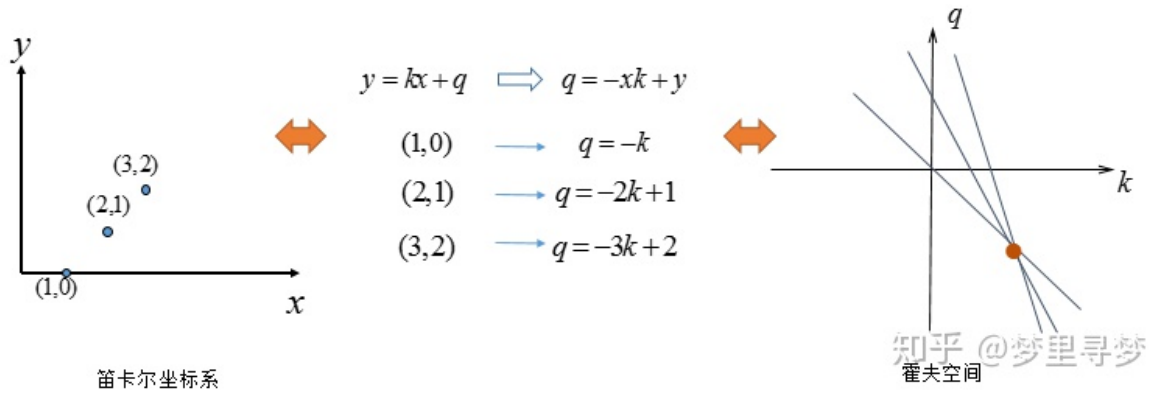
It also holds vice versa (a **point** in Cartesian coordinates can be represented as a **line** in Hough Space.):



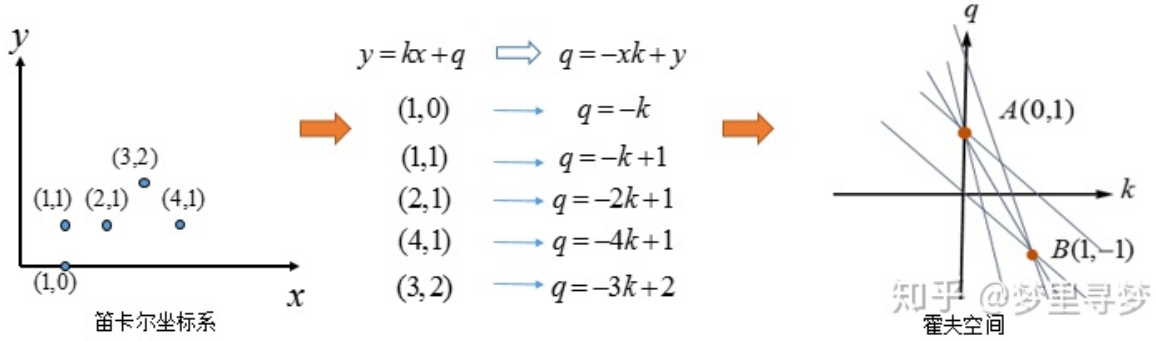
If there're two points in Cartesian coordinates:



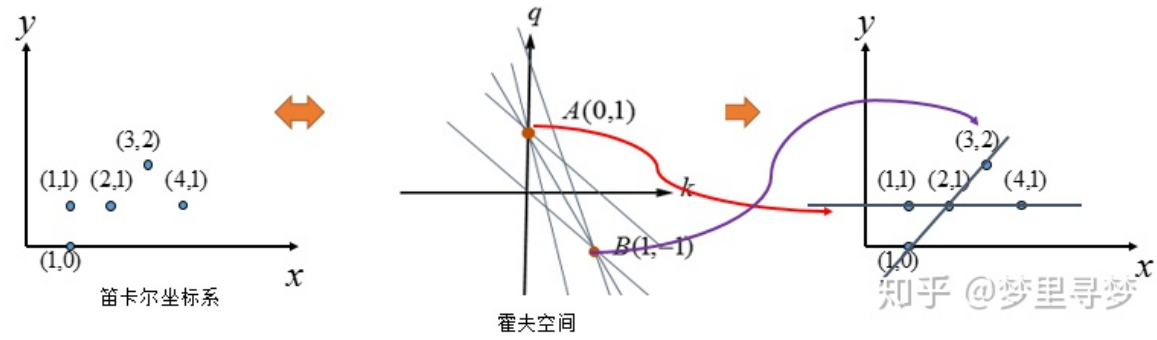
If there're three **collinear** points in Cartesian coordinates:



If there're more than one coordinates with more points:



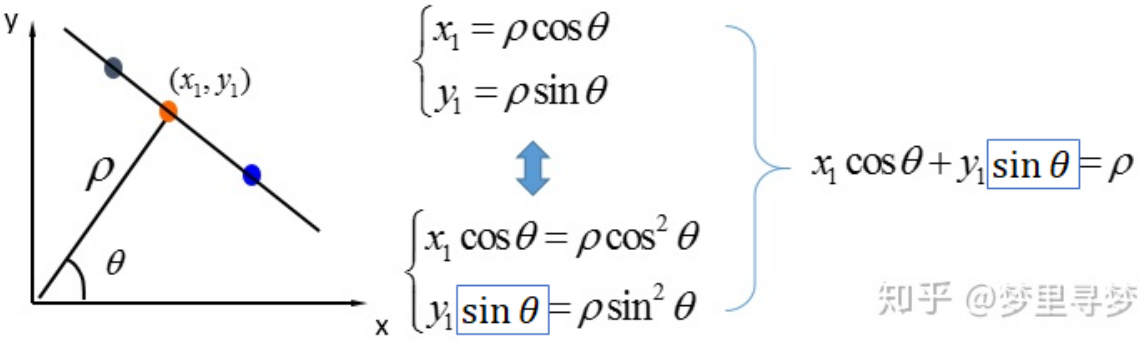
The lines can be determined using points that formed by most intersections in **Hough Space**:



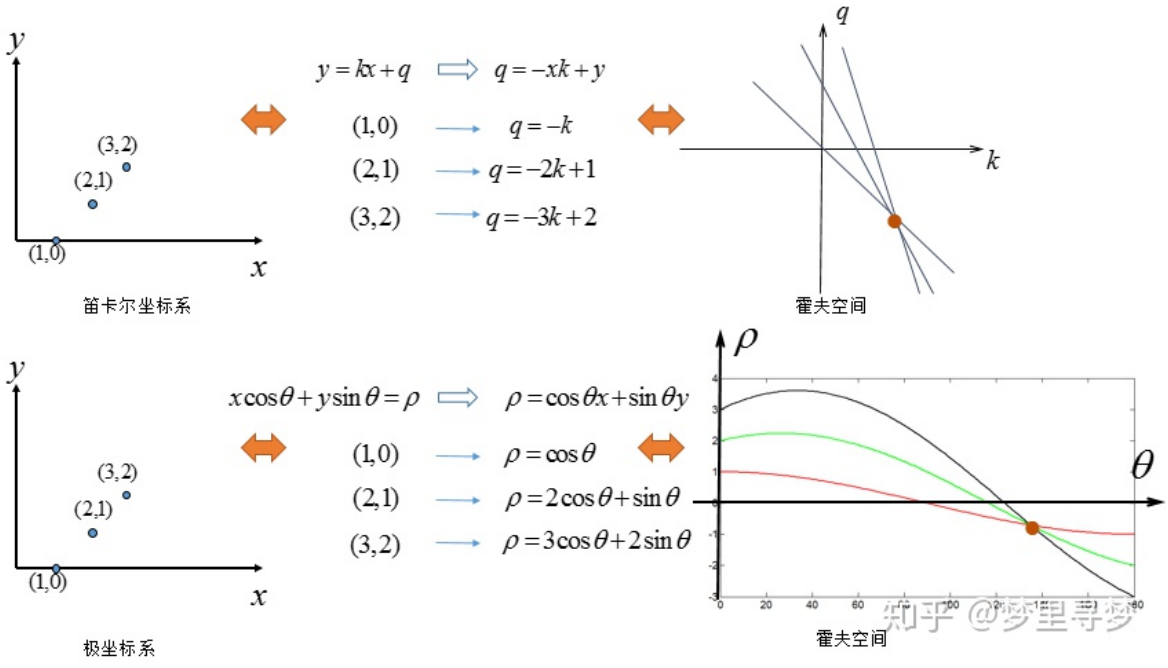
Hough Space in Polar Coordinates using [Hesse normal form](#) (Not naive Polar Coordinates!)

In Polar Coordinates, the line can be represented using (ρ, θ):

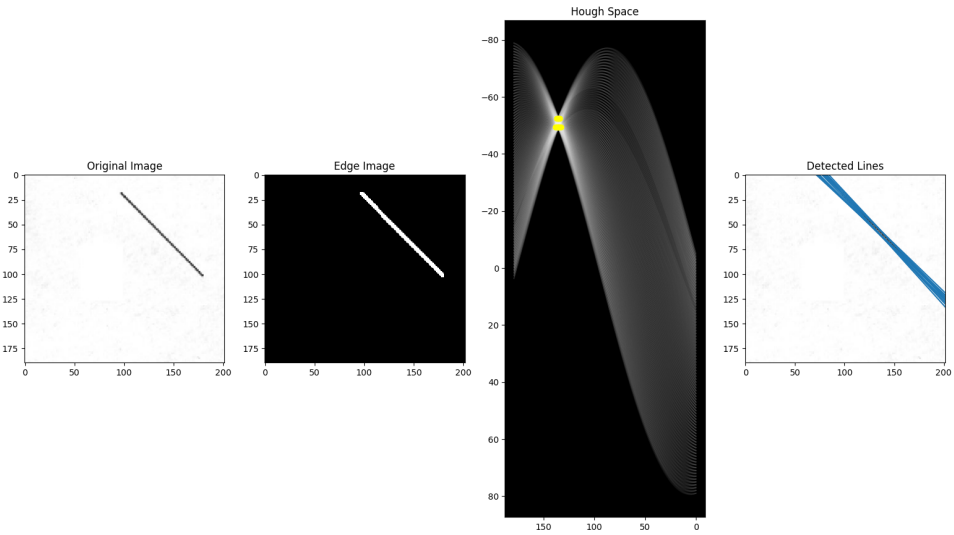
$$x \cdot cos\theta + y \cdot sin\theta = \rho \tag{3}$$



Same as the Hough Space in Cartesian coordinates, a point in Polar Coordinates can be represented by a line in Hough Space:



The whole steps of detecting lines in image:



Reference

1. https://towardsdatascience.com/lines-detection-with-hough-transform-84020b3b1549
2. https://zhuanlan.zhihu.com/p/203292567
3. https://en.wikipedia.org/wiki/Hough_transform#:~:text=The%20Hough%20transform%20is%20a,shapes%20by%20a%20voting%20procedure.