



ELABORAZIONE DELLE IMMAGINI

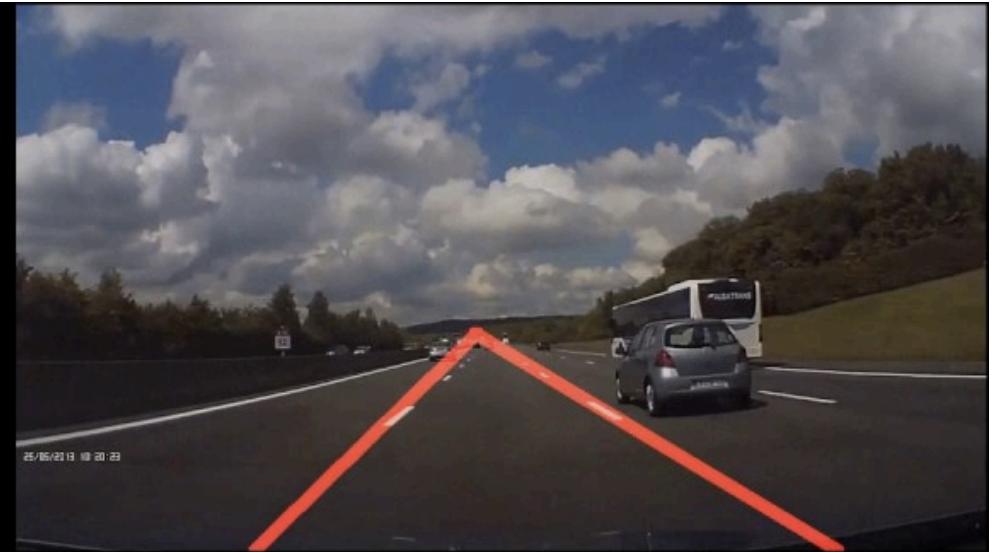
SEGMENTAZIONE DELLE IMMAGINI: TRASFORMATA DI HOUGH

LA TRASFORMATA DI HOUGH

- È una tecnica che permette di **riconoscere** particolari **configurazioni** di **punti** presenti **nell'immagine**
 - **rette**
 - **cerchi**
 - altre forme
- In generale questa tecnica si **applica** se la **forma** cercata può essere **espressa** tramite una **funzione** nota che fa uso di un insieme di **parametri**
- L'insieme dei **parametri definisce** una particolare **istanza** della forma
- Viene definito un **operatore globale** perchè **non opera** su un **intorno** di ogni pixel ma analizza le sue **proprietà globali** (**rispetto** agli **altri pixel** dell'immagine)



ESEMPI

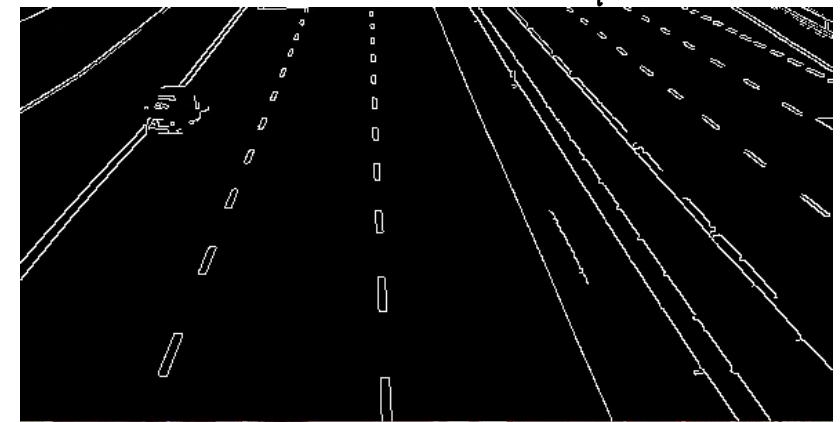


Elim Parte II – Prof. A. Ferone

ESEMPI



ESEMPI



Elim Parte II – Prof. A. Ferone

Hough

La Trasformata di Hough non riesce a restituire un intervallo, ma individua + intervalli
↓
non individua il cerchio, ma tutti quelli presenti

TRASFORMATA DI HOUGH PER RETTE

- Se si assume come **rappresentazione** della **retta** la forma canonica $y=mx+b$, qualunque retta è completamente specificata dal valore dei **parametri (m,b)**
 - $b = \text{intercetta}$
 - $m = \text{coeff angolare}$
- Se si assume un tipo di **rappresentazione** diversa, es. la forma **normale** di Hesse, $\rho = x \cos \theta + y \sin \theta$, la retta è completamente specificata dal valore dei **parametri (ρ, θ)**
- Dati dei **punti dell'immagine** si vogliono **trovare i parametri** delle **rette** su cui giacciono

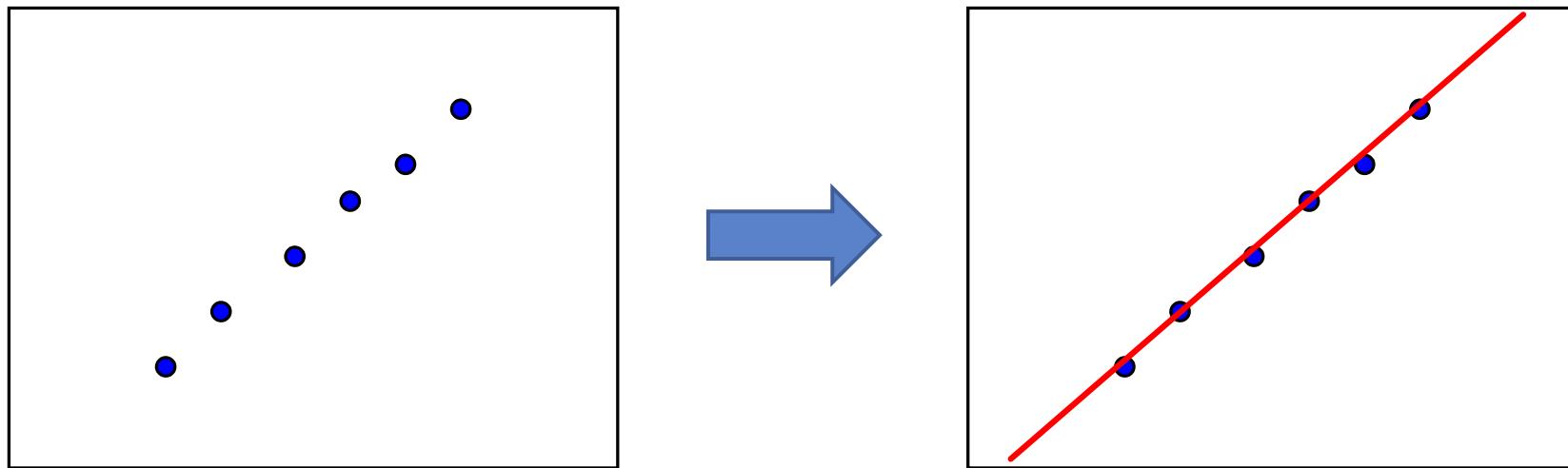
raglio trovare i parametri delle rette su cui giacciono i pixel di edge

Elim Parte II – Prof. A. Ferone

Noi partiamo prima dai pixel di edge e poi devo capire se si trovano su una retta, dopo trovare i parametri

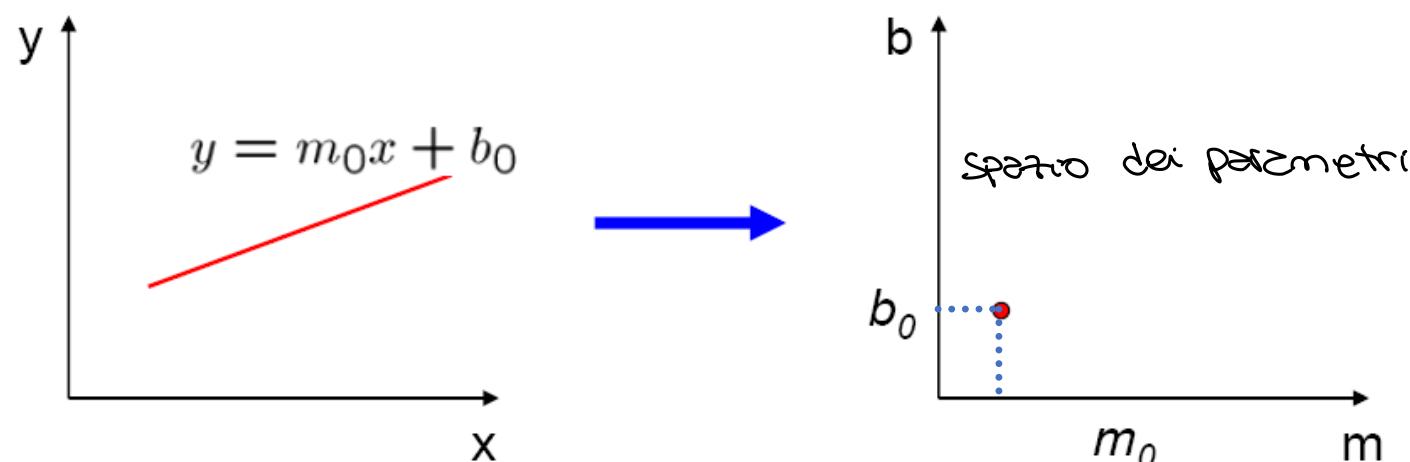
RETTE

i pixel non si dispongono esattamente lungo la retta
quindi non rispettano tutti esattamente l'equazione
della retta
↓
presenta di ERRORI



SPAZIO DEI PARAMETRI

- Fissata la **forma** di interesse (retta) e la sua **rappresentazione**, è possibile considerare una **trasformazione** dallo spazio dell'**immagine** allo spazio dei **parametri** (m, b)
- Nello **spazio dei parametri**, una particolare **istanza** di **retta** viene **rappresentata** da un **punto** di coordinate (m_0, b_0)



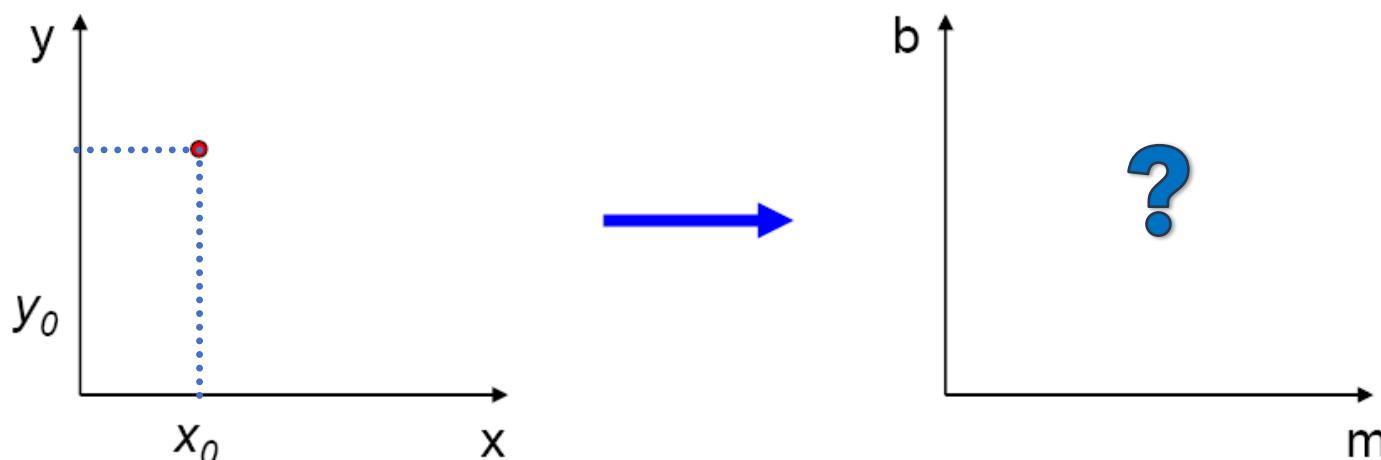
Elim Parte II – Prof. A. Ferone

Voglio trovare i parametri m_0 e b_0

In effetti se riesco a trasformo un punto nello spazio dei parametri: so dove è la retta

SPAZIO DEI PARAMETRI

- A cosa **corrisponde** un **punto** dello spazio **immagine** nello spazio dei **parametri**?
- Riscriviamo l'equazione della retta ($y=mx+b$) come **$b=-xm+y$**



Poiché in un punto passano infinite rette e ogni retta è individuata dai parametri (m, b) allora avremo tanti punti nello spazio dei parametri:

Elim Parte II – Prof. A. Ferone

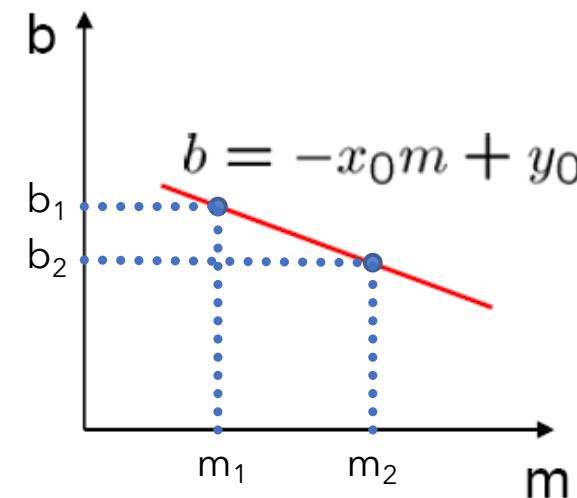
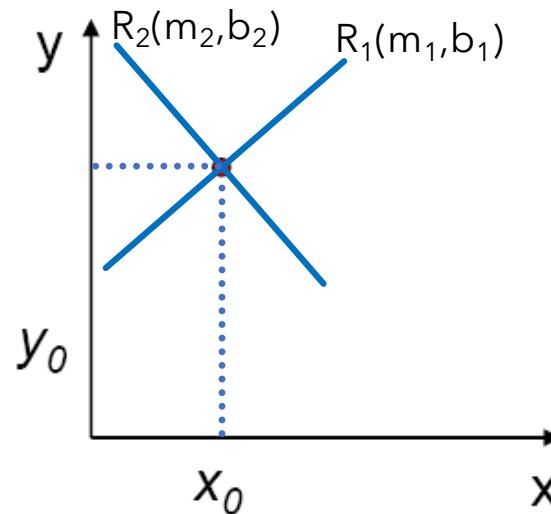
quante sono le rette

↳ le rette si intersecano tutte nello stesso punto di edge (x, y)

Quindi gli infiniti punti sono rappresentati nello spazio dei parametri: le due rette di coefficienti x_0, y_0 che sono i nuovi parametri

SPAZIO DEI PARAMETRI

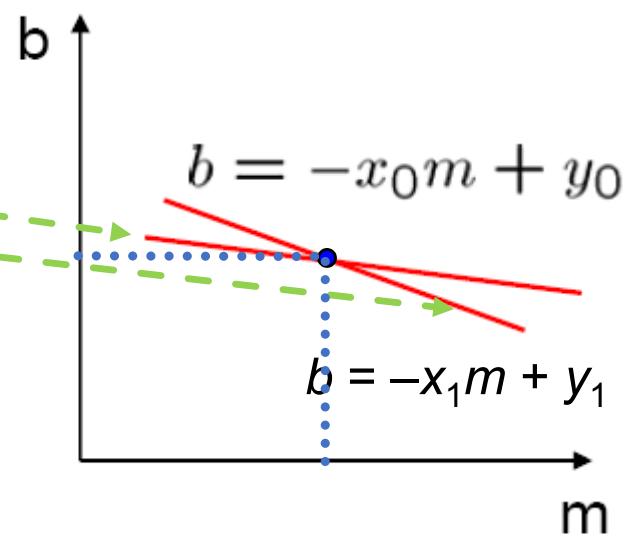
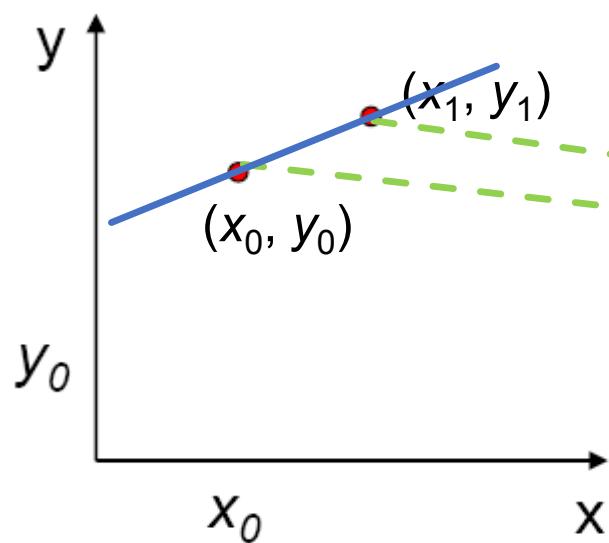
- Per un **punto** passano **infinite rette**, ognuna **identificata** da una coppia di parametri
- Un **punto** nello spazio **immagine** corrisponde ad una **retta** nello spazio dei **parametri**



Elim Parte II – Prof. A. Ferone

SPAZIO DEI PARAMETRI

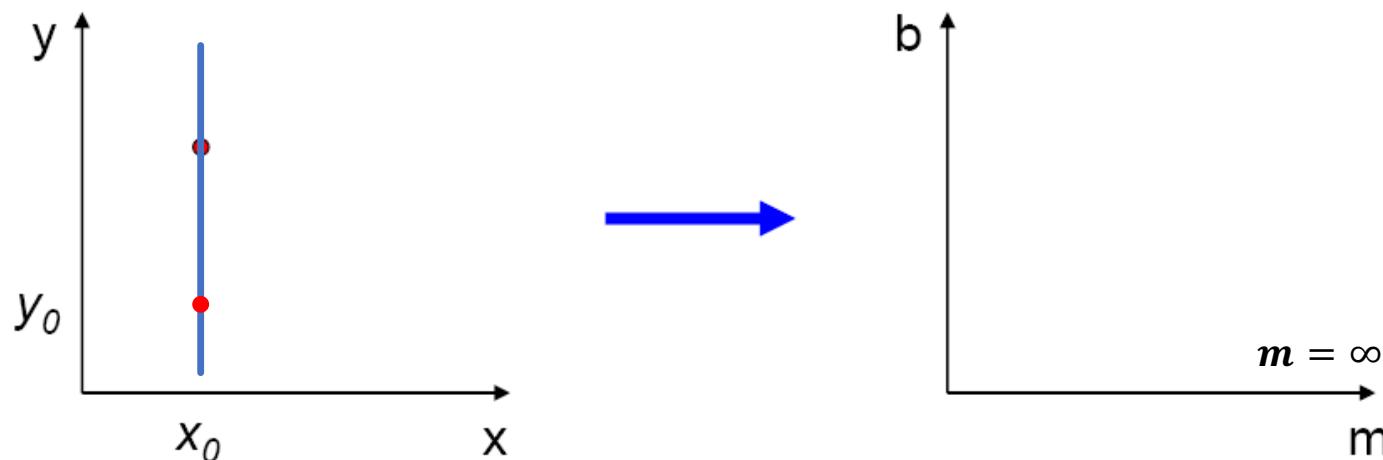
- Se si considerano **due punti** nello spazio **immagine**, la **retta** che li **contiene** entrambi è **identificata** dai parametri **(m, q)** che corrispondono al punto di intersezione delle rette corrispondenti ai due punti nello spazio dei parametri



Elim Parte II – Prof. A. Ferone

SPAZIO DEI PARAMETRI

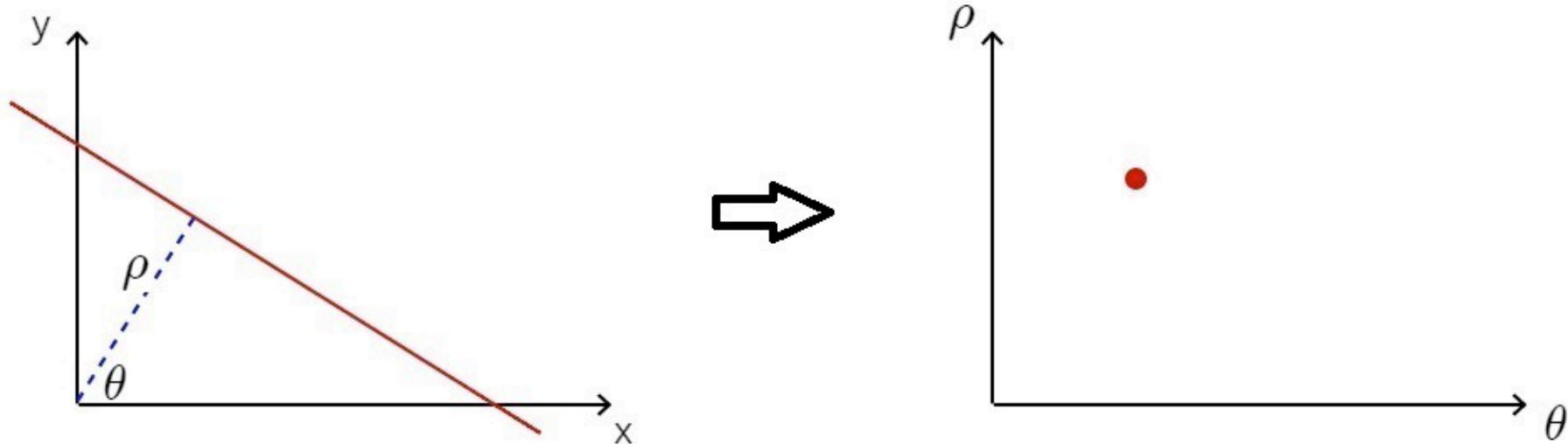
- Questa **rappresentazione** della retta presenta un **problema**
- Quando la retta è verticale $m = \infty$



Elim Parte II – Prof. A. Ferone

SPAZIO DEI PARAMETRI

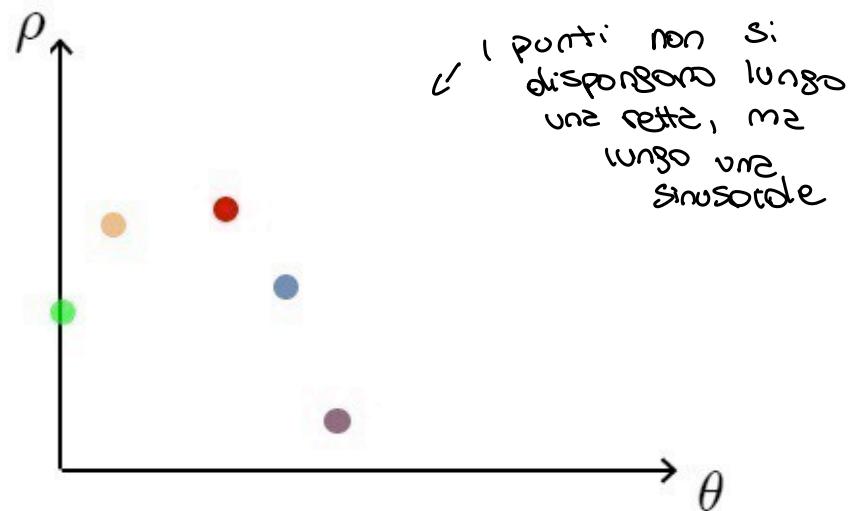
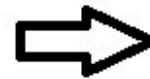
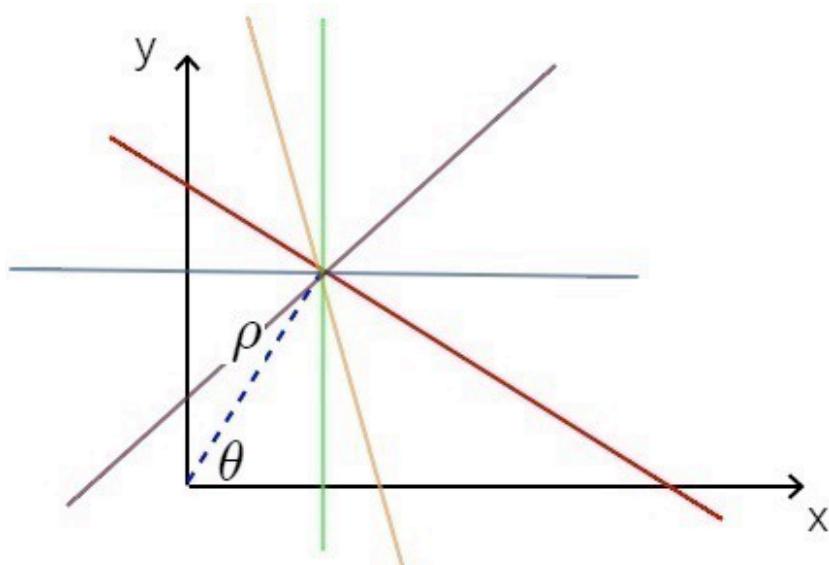
- È possibile utilizzare la **rappresentazione polare** (forma normale di Hesse)
- $\rho = x \cos \theta + y \sin \theta$
- Una **retta** nello spazio immagine viene **rappresentata** da un punto nello spazio dei **parametri** (ρ, θ)



Elim Parte II – Prof. A. Ferone

SPAZIO DEI PARAMETRI

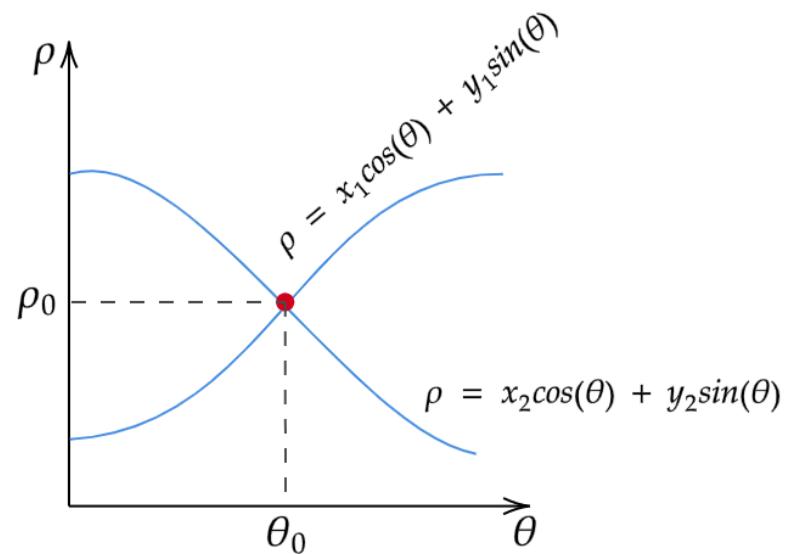
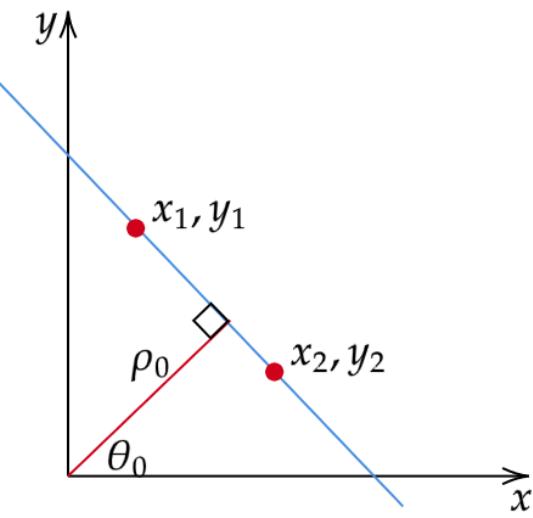
- Per un **punto** passano **infinite rette**, ognuna identificata da una **coppia di parametri**
- Un **punto** nello spazio **immagine** corrisponde ad una **sinusoide** nello spazio dei **parametri**



Elim Parte II – Prof. A. Ferone

SPAZIO DEI PARAMETRI

- Il **punto di intersezione** nello spazio dei parametri individua la **retta** su cui **giacciono i punti** nel spazio **immagine**



Elim Parte II – Prof. A. Ferone

SCHEMA DI VOTO

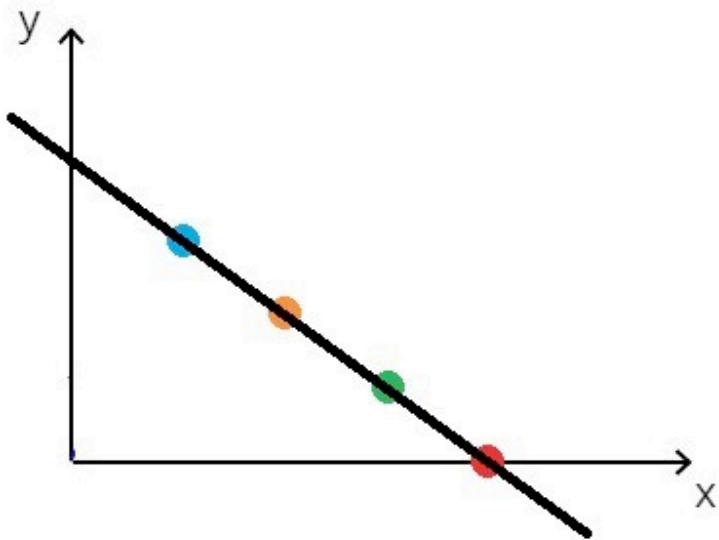
Ogni pixel di edge "vota" un certo numero di rette che lo possono attraversare

- Come possiamo **trovare i punti di intersezione** delle rette?
- Si può immaginare di **quantizzare** lo **spazio** dei **parametri**
- Lo spazio quantizzato, detto **spazio dei voti**, è caratterizzato da "celle" (**accumulatori**)
- Per ogni **punto** nello spazio **immagine** si aggiunge un "**voto**" (incrementa) nelle **celle** (accumulatore) corrispondenti ai **parametri** delle **rette** che **passano** per quel **punto**
- Quando **tutti i punti** hanno "**votato**", le celle contenenti un **numero di voti superiore** ad una certa **soglia** corrispondono alle **rette** nello **spazio immagine**

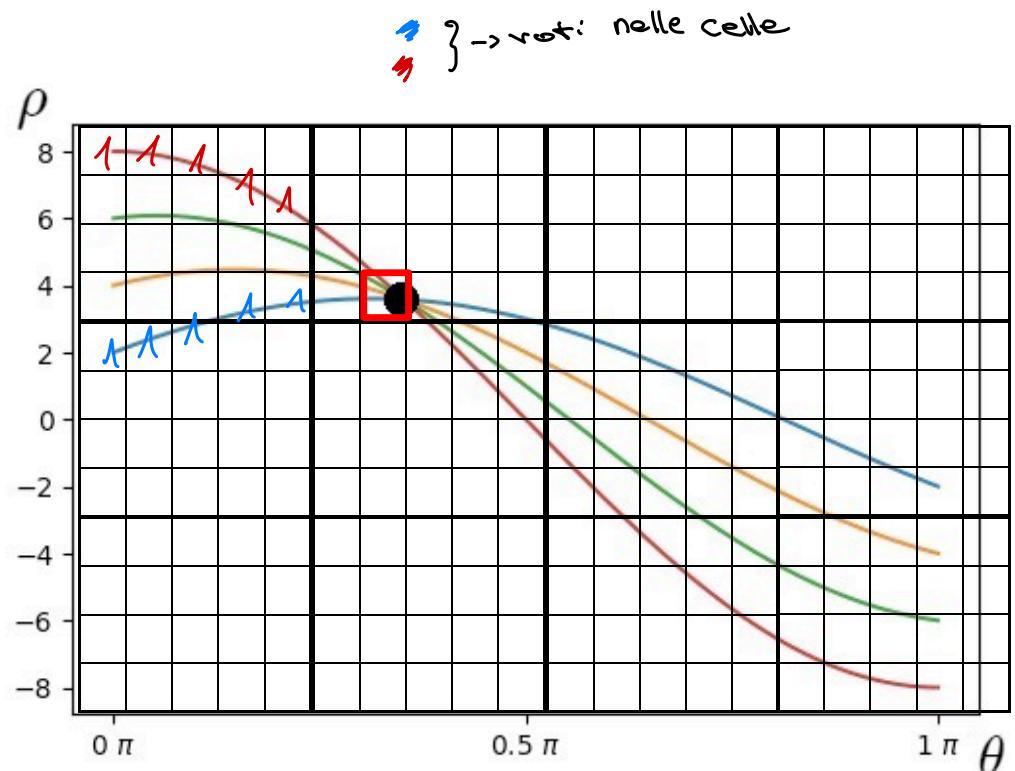
Una retta che tutti hanno rotato si vede dove sono le intersezioni, cioè se c'è una coppia di rette + punti

Elim Parte II – Prof. A. Ferone

SCHEMA DI VOTO

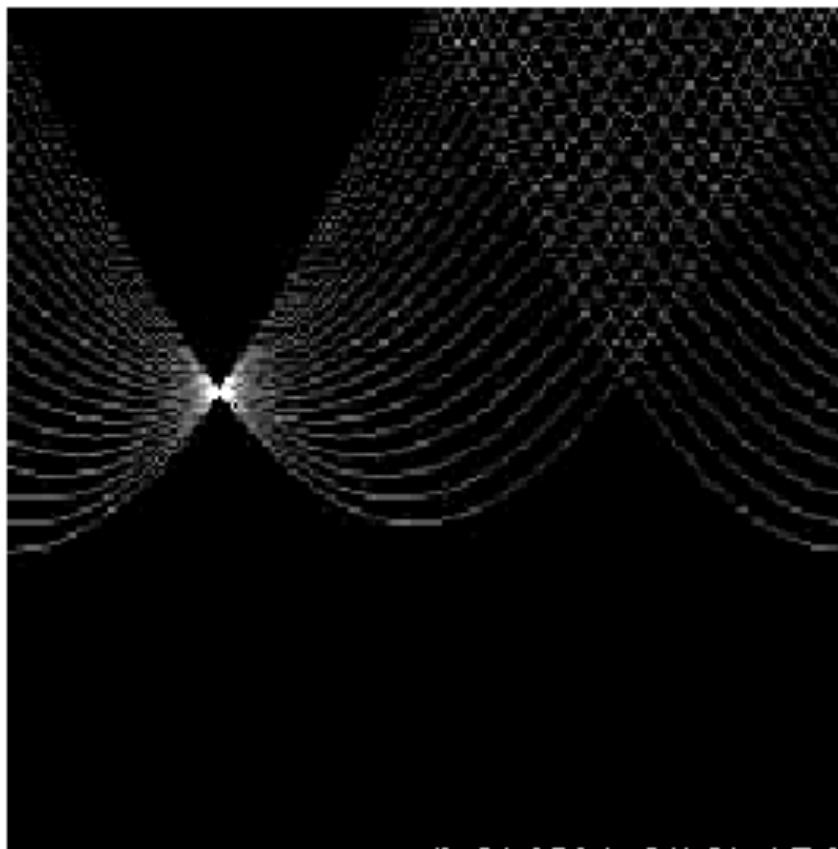
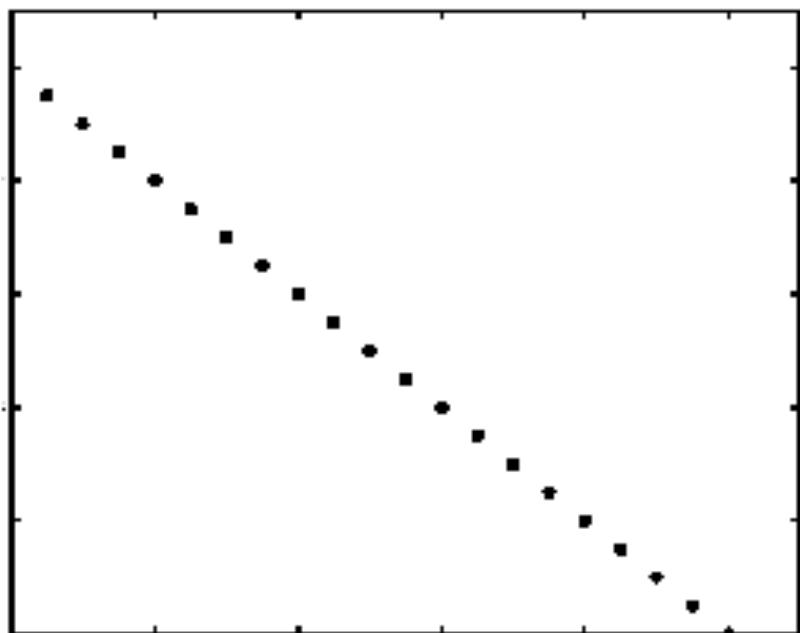


Lo schema di voto risolve il problema dei voti che non giacciono esattamente sulla retta



Elim Parte II – Prof. A. Ferone

SCHEMA DI VOTO



Elim Parte II – Prof. A. Ferone

TRASFORMATA DI HOUGH PER CERCHI

- Utilizzando lo **stesso procedimento** è possibile trovare i **cerchi** presenti all'interno dell'immagine
- Una circonferenza con centro (a,b) e raggio R è rappresentato dall'equazioni parametrica, al variare dell'angolo θ in $[0,360]$

$$x = a + R \cos \theta$$

$$y = b + R \sin \theta$$

- In un'immagine possono essere presenti cerchi con **differenti centri** e differenti **lunghezze** dei **raggi**
- In questo caso lo **spazio** dei **parametri** è **tridimensionale** (a,b,R) così come lo **spazio** dei **voti**

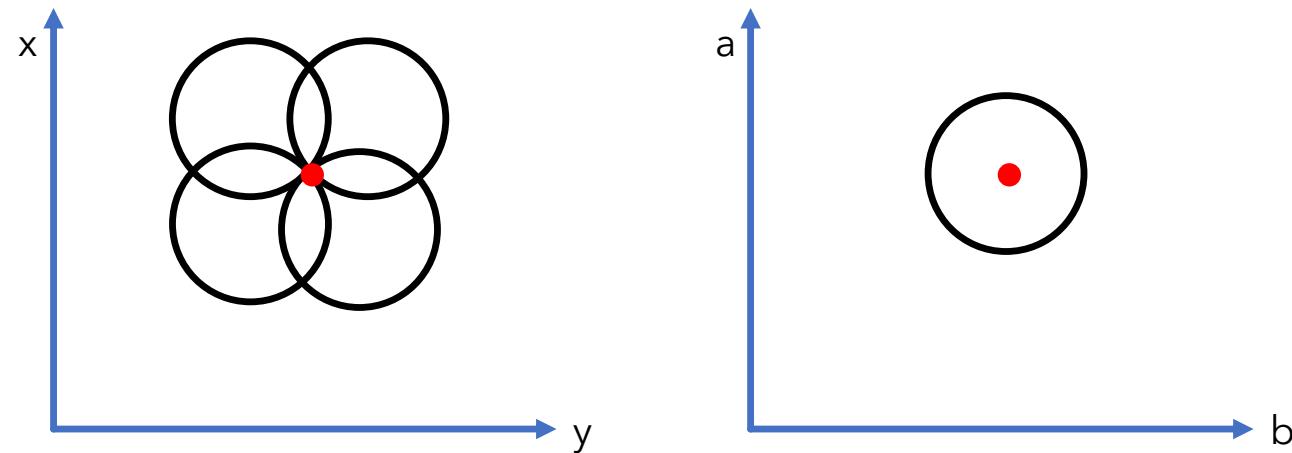
Fissato R , i parametri che variano sono solo a e b \Rightarrow ritorno allo spazio



es. Fisso $R = 5$ voglio tutte le circonferenze di raggio 5

TRASFORMATA DI HOUGH PER CERCHI

- Ipotizziamo di **conoscere** la **lunghezza** del **raggio** delle circonference che stiamo cercando
- Dei tre **parametri** delle equazioni parametriche **solo due** posso **variare** (a e b) per cui lo **spazio** dei **parametri** è un **piano**
- Per un **punto** nello **spazio immagine** passano **infinite circonference** di raggio R



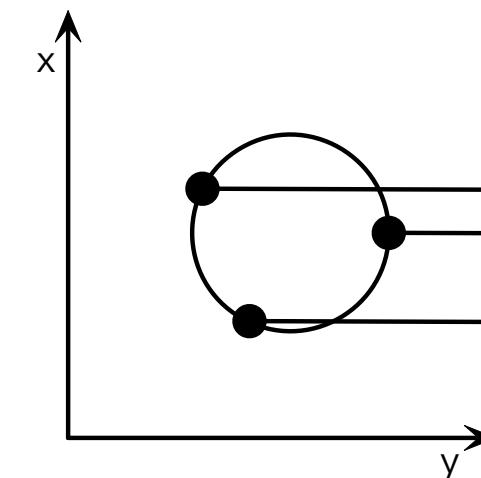
Elim Parte II – Prof. A. Ferone

TRASFORMATA DI HOUGH PER CERCHI

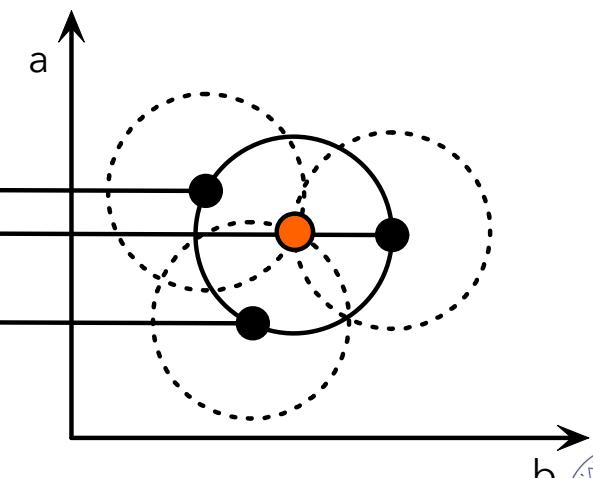
- Ogni **punto** nello **spazio immagine** corrisponde ad una **circonferenza** nello **spazio dei parametri**
- Per **tre punti non allineati** passa **una** ed una sola **circonferenza** le cui **coordinate del centro** corrispondono al **punto di intersezione** nello **spazio dei parametri**

$$a = x - r * \cos\left(\theta * \frac{\pi}{180}\right)$$

$$b = y - r * \sin\left(\theta * \frac{\pi}{180}\right)$$

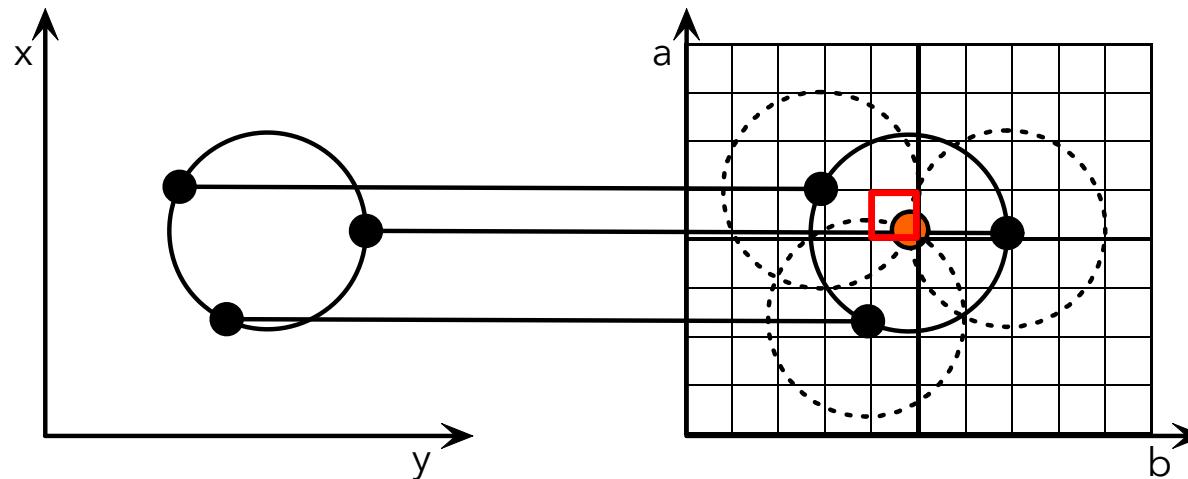


Elim Parte II – Prof. A. Ferone



TRASFORMATA DI HOUGH PER CERCHI

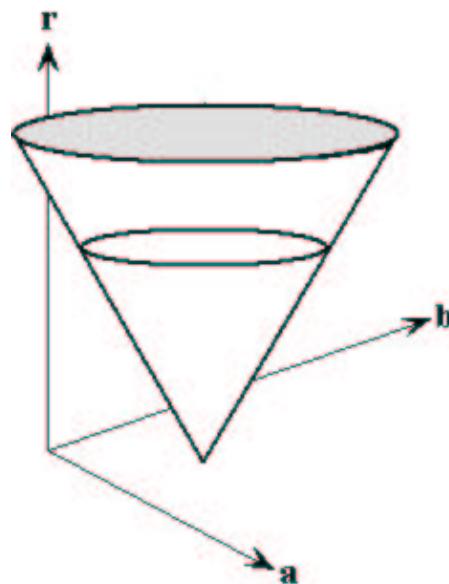
- Ogni **punto** nello **spazio immagine** "vota" per una **circonferenza** nello **spazio dei voti** (ovvero lo spazio dei parametri quantizzato)
- Le **celle** con un **numero** di **voti superiore** ad una data **soglia** corrispondono alle **circonferenze** presenti nell'**immagine**



Elim Parte II – Prof. A. Ferone

TRASFORMATA DI HOUGH PER CERCHI

- Nel caso in cui la **lunghezza** del **raggio** non sia **nota**, tutti i parametri possono variare per cui lo **spazio** dei **parametri** è **tridimensionale**
- In particolare, ogni **punto** nello **spazio immagine** corrisponde alla **superficie di un cono** nello **spazio dei parametri** (una circonferenza per ogni raggio)



Elim Parte II – Prof. A. Ferone

CARATTERISTICHE

- Per **ridurre** il **numero** dei **parametri** è possibile utilizzare l'informazione relativa alla **direzione** del **gradiente**
- La trasformata di Hough è **robusta al rumore** presente nell'immagine e ad eventuali **interruzioni** (gap) nelle **forme** ricercate
- La trasformata di Hough può **individuare** più **istanze** di una forma in una **singola esecuzione**
- La trasformata di Hough è **parallelizzabile**

PASSI DELL'ALGORITMO DI HOUGH PER RETTE

- Inizializzare l'accumulatore H (SPAZIO DEI ROTI) *-hypot* *calcola la lunghezza delle diagonale*
- Applicare l'algoritmo di Canny per individuare i punti di edge
- Per ogni punto (x, y) di edge
- Per ogni angolo $\theta = 0: 180$ calcolare $\rho = x \cos \theta + y \sin \theta$
- Incrementare $H(\rho, \theta) = H(\rho, \theta) + 1$
- Le celle $H(\rho, \theta)$ con un valore maggiore di una soglia th corrispondono alle rette nell'immagine

PASSI DELL'ALGORITMO DI HOUGH PER CERCHI

- Inizializzare l'accumulatore H
- Applicare l'algoritmo di Canny per individuare i punti di edge
- Per ogni punto (x, y) di edge
- Per ogni angolo $\theta = 0: 360$ e per ogni raggio $r=R_{\min} : R_{\max}$
- Calcolare $a = x - r * \cos\left(\theta * \frac{\pi}{180}\right)$ e $b = y - r * \sin\left(\theta * \frac{\pi}{180}\right)$
- Incrementare $H(a, b, r) = H(a, b, r) + 1$
- Le celle $H(a, b, r)$ con un valore maggiore di una soglia th corrispondono alle circonferenze nell'immagine

TRANSFORMATA DI HOUGH IN OPENCV

```
void cv::HoughLines(  
    cv::InputArray image,           // Input single channel image  
    cv::OutputArray lines,          // N-by-1 two-channel array  
    double rho,                   // rho resolution (pixels)  
    double theta,                 // theta resolution (radians)  
    int threshold,                // Unnormalized accumulator threshold  
    double srn = 0,                // rho refinement (for MHT)  
    double stn = 0                 // theta refinement (for MHT)  
);  
  
void cv::HoughCircles(  
    cv::InputArray image,           // Input single channel image  
    cv::OutputArray circles,         // N-by-1 3-channel or vector of Vec3f  
    int method,                   // Always cv::HOUGH_GRADIENT  
    double dp,                     // Accumulator resolution (ratio)  
    double minDist,                // Required separation (between lines)  
    double param1 = 100,             // Upper Canny threshold  
    double param2 = 100,             // Unnormalized accumulator threshold  
    int minRadius = 0,              // Smallest radius to consider  
    int maxRadius = 0               // Largest radius to consider  
);
```

Elim Parte II – Prof. A. Ferone

ESERCIZI

- Implementare la trasformata di Hough per rette
- Implementare la trasformata di Hough per cerchi

