



# Processamento Digital de Imagens

## Conceitos Básicos

E Carto/UFPR

Prof. Dr. Jorge Centeno



# Realidade e imagem

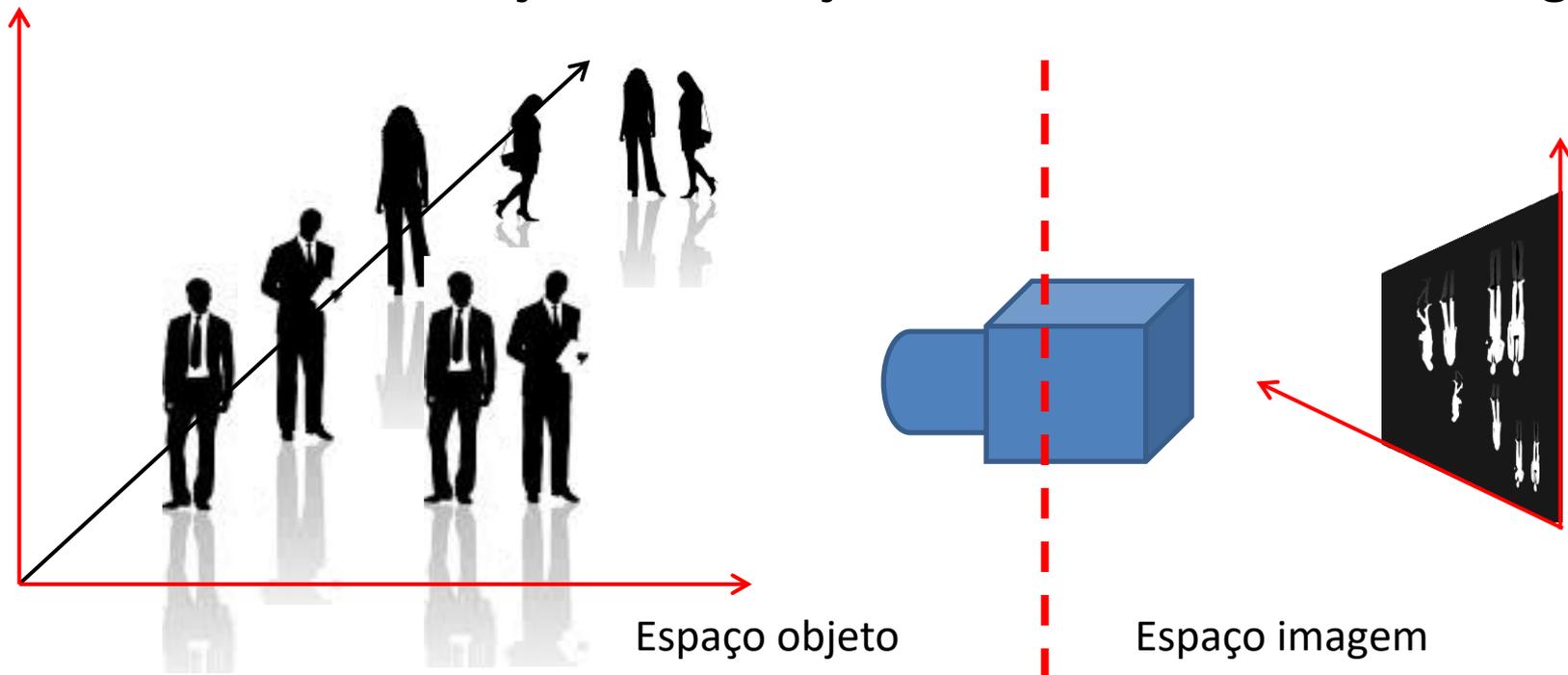
- Uma imagem é a representação pictórica de um aspecto da realidade.
- Uma imagem não é idêntica à cena real, pois ela retrata apenas uma propriedade particular:
  - Radiação refletida
  - Radiação transmitida
  - Etc
- deixando de registrar algumas propriedades e relações que podem ser importantes para a adequada compreensão da cena.



# A considerar

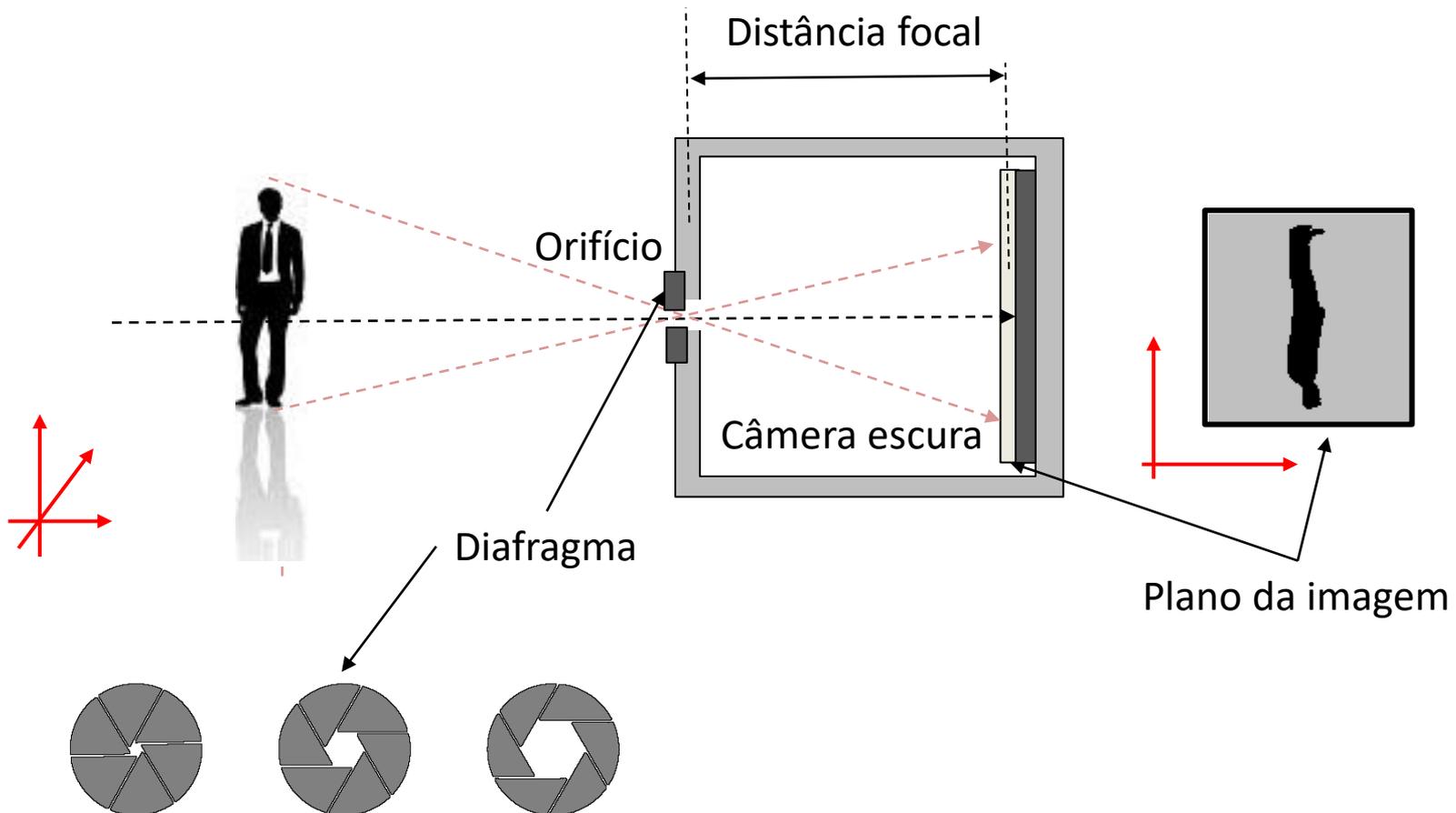
Transformação espacial: projeção do espaço objeto ao espaço da imagem.

- O registro da radiação incidente no sensor
- A transformação da radiação incidente em um valor digital



# Modelo pin-hole

- Modelo pin-hole



# *A formação da imagem*

No processo de registro da radiação incidente no sensor, a cena é simplificada de várias maneiras. Os principais aspectos são

- Aspecto Geométrico
  - discretização.
- Aspecto Radiométrico
  - quantização.

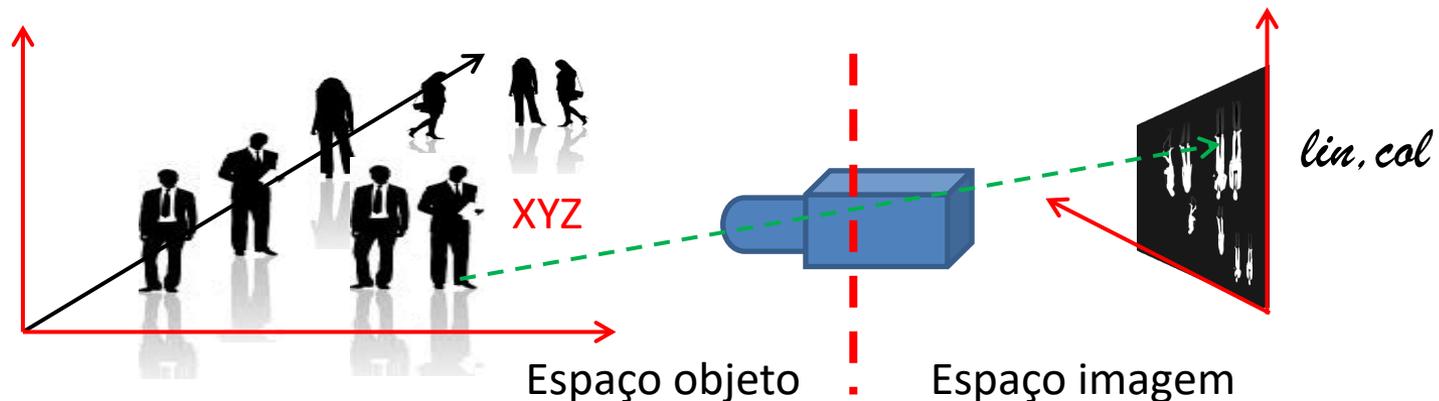
- 
- Aspecto Geométrico
    - Discretização
- 

# A transformação GEOMÉTRICA

determina a posição dos objetos (pontos) existentes na cena (mundo real) no espaço da imagem. Ele responde à pergunta:

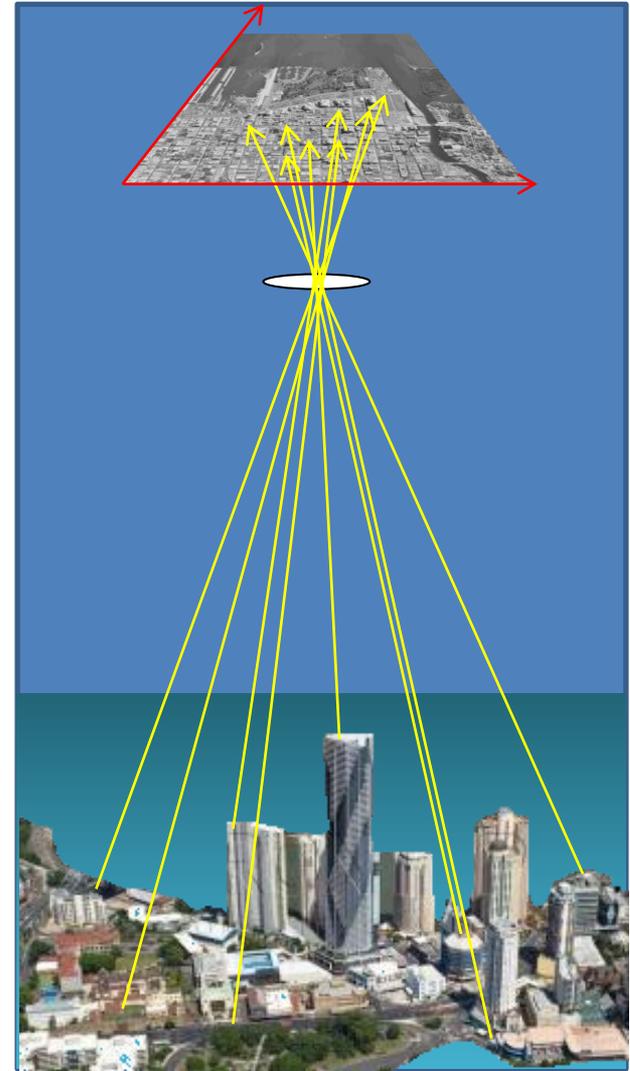
*“Onde se encontra o ponto, com coordenadas conhecidas no mundo real ( $X; Y, Z$ ) dentro do plano da imagem?”*

- Ou...
- *“A que ponto no espaço objeto corresponde o ponto com coordenadas (linha, coluna) na imagem ?”*

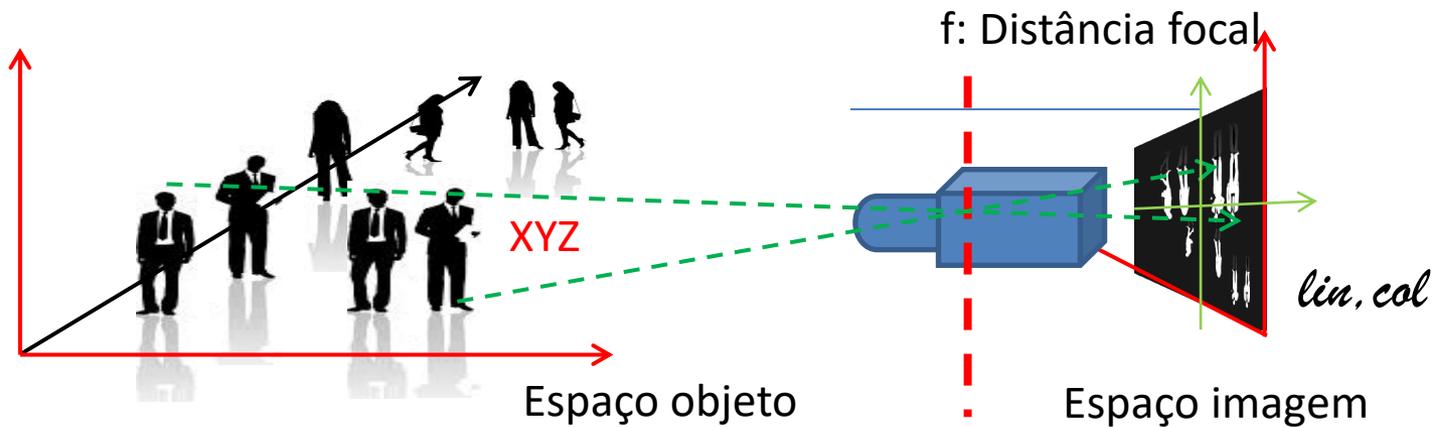


# Projeção

- Basicamente, a coleta de imagens é o resultado da projeção de um espaço tridimensional (espaço objeto) num espaço bidimensional (espaço imagem).
- Esta transformação envolve perda de informação espacial, pois uma coordenada é perdida.
- A formação da imagem podem ser modelada usando a transformação da projeção central. É assumido que todos os feixes de luz passam pelo mesmo ponto no plano focal e incidem no plano da imagem localizado a uma distância  $d'$  do plano focal, com as coordenadas:
  - $x = (x_1, x_2, d')$



# Consequências

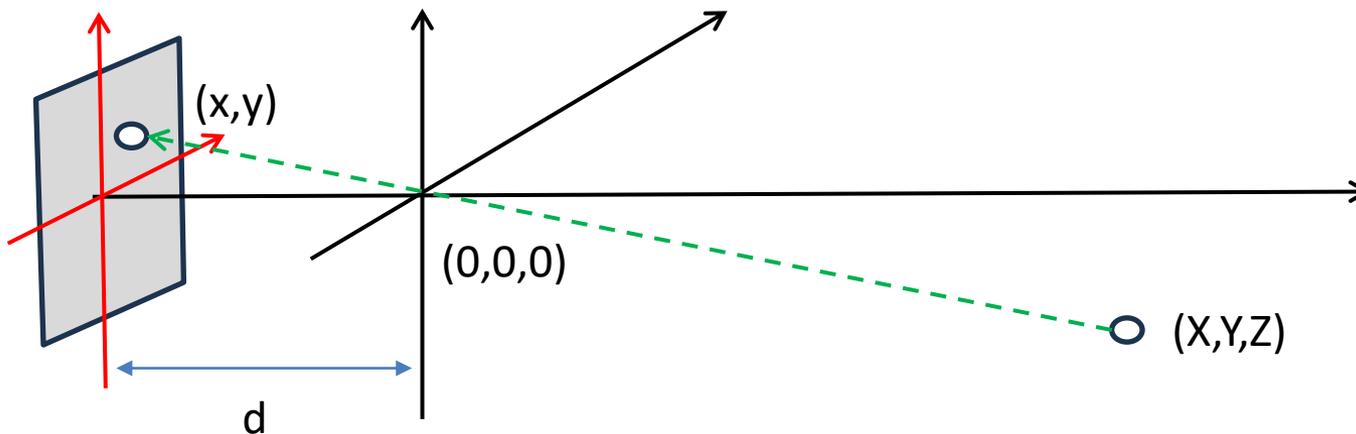


$$(lin, col) = F(X, Y, C, f)$$

- Consequências:
- Vários objetos do espaço tridimensional podem ser projetados no mesmo ponto da imagem: Oclusão.
- O modelo de câmara “pinhole” é uma simplificação. Para ser mais fiel à realidade e modelar adequadamente esta relação espacial, é necessário levar em consideração outros fatores como as deformações introduzidas pelo sistema óptico ou os efeitos da atmosfera na propagação do feixe de luz.

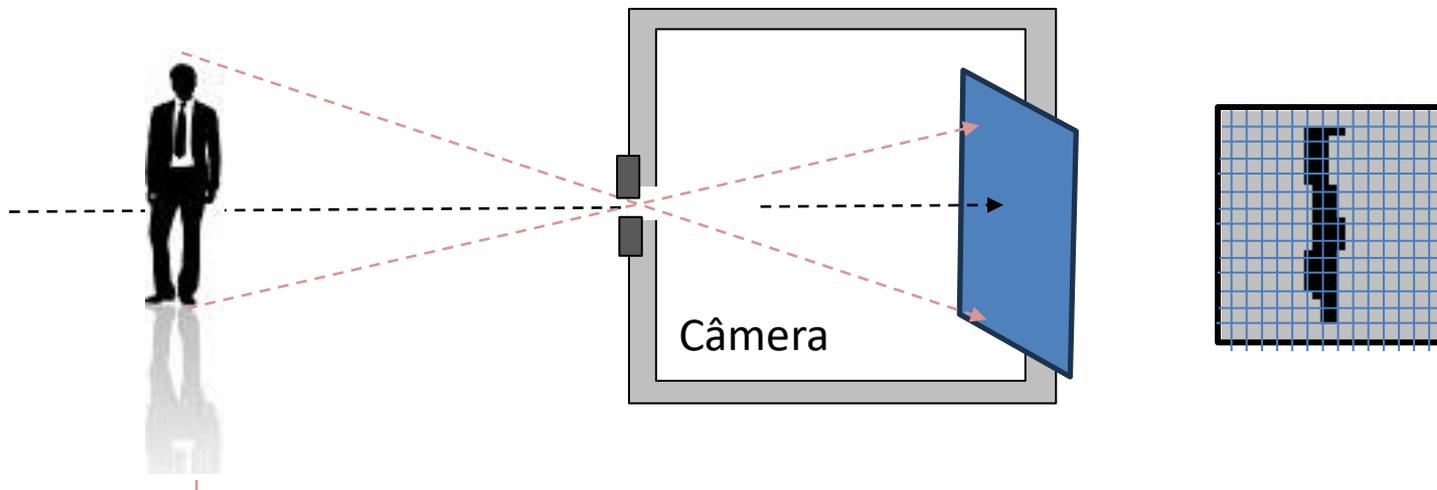
# problema

- Dadas as coordenadas de pontos no espaço objeto, e a posição do ponto principal (do modelo pin-hole), determine as coordenadas no espaço imagem de cada ponto, considerando que o plano de imagem se localiza a uma distância “d” do ponto principal...



# Discretização

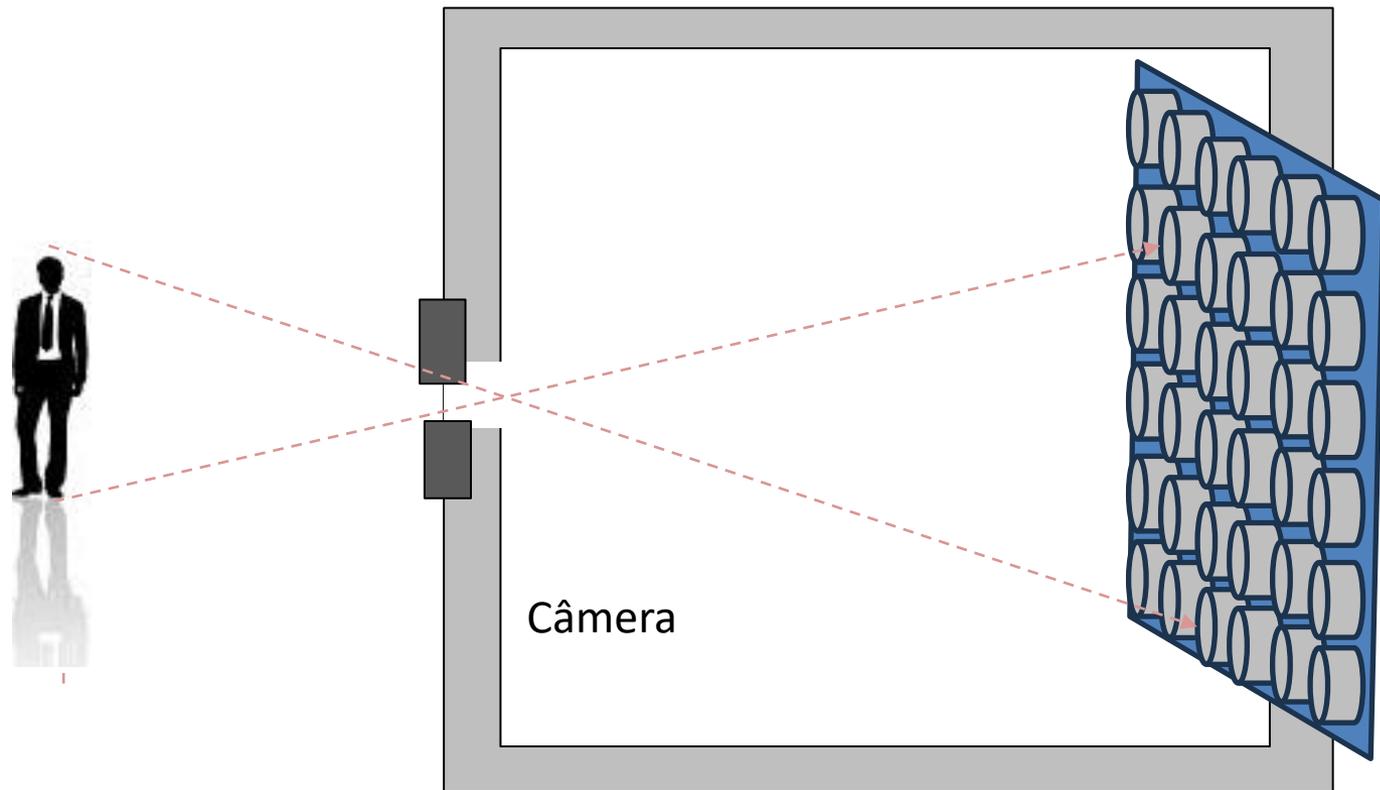
- Transformar o plano contínuo em um espaço discreto 2D
- Grade/raster



Discretização

# Discretização

O elemento sensor (CCD, CMOS) é composto por um arranjo de elementos fotossensíveis.



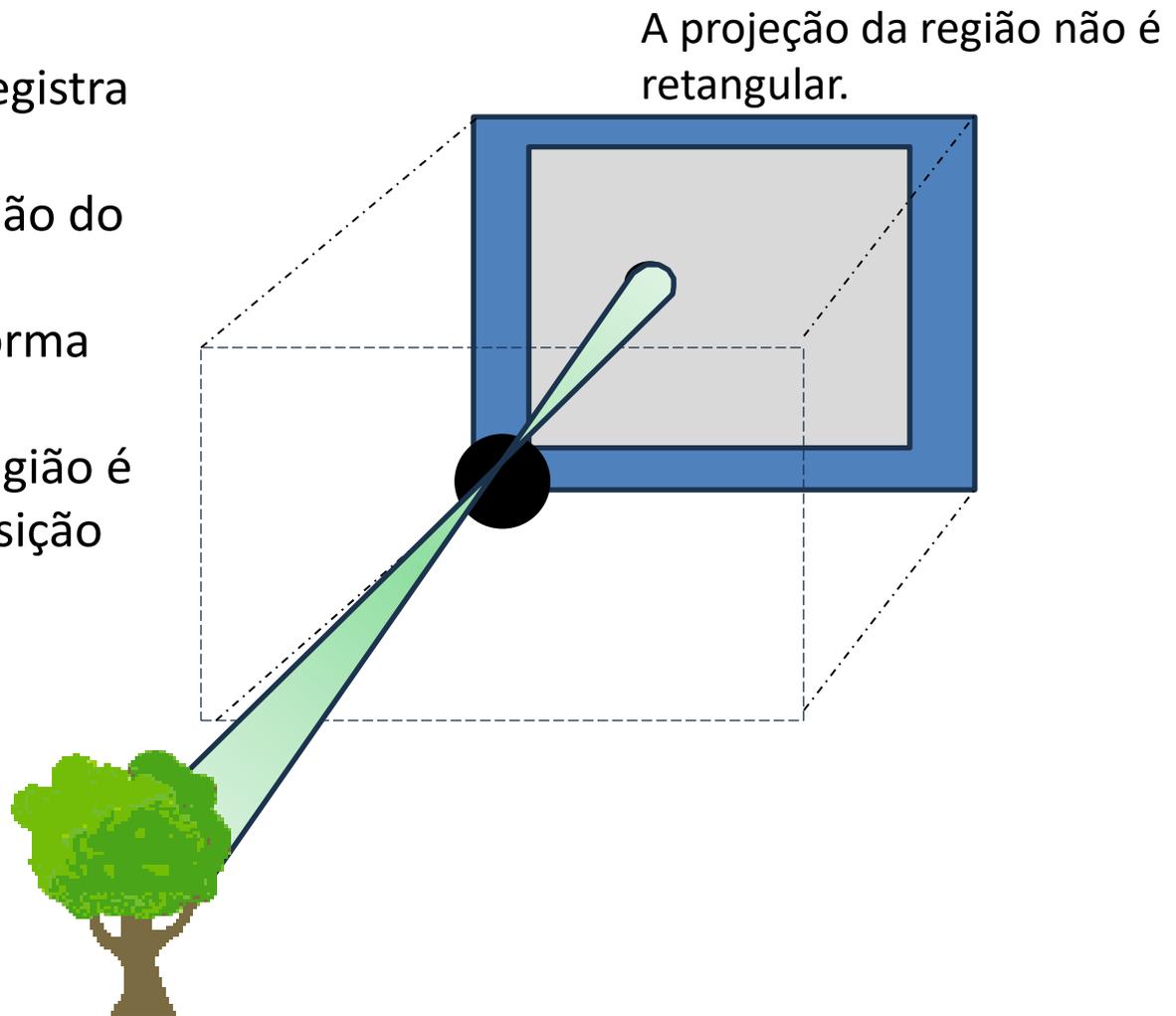
Logo, as medidas são discretas

# Discretização

Cada elemento sensor registra a radiação incidente proveniente de uma região do espaço objeto.

Cada região projetada forma um pixel.

A intensidade total da região é armazenada em essa posição discreta (valor do pixel).



# Discretização

As posições dos pontos no plano da imagem são restritas a valores inteiros dentro da faixa de variação definida pelo tamanho da grade (NlinxNcol)

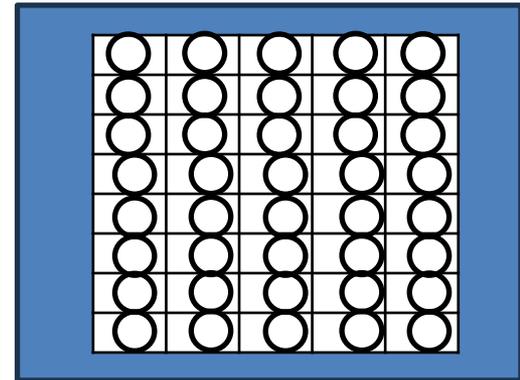
$$F(x,y)$$

Com

$$x=1,2,3,\dots,Ncol$$

$$y=1,2,3,\dots,Nlin$$

Sendo assim, cada elemento detector registra a energia incidente em uma unidade de área do plano da imagem, do que se deriva um valor único a ser armazenado na matriz digital.

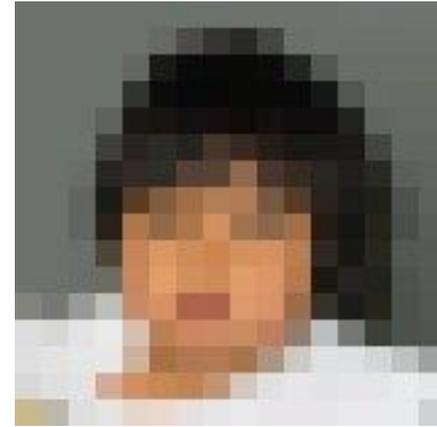
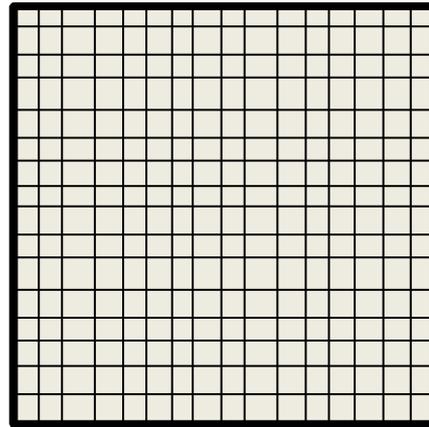


20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	27	22	25	13
32	13	34	20	20
12	38	30	68	84
11	78	164	61	57

# Discretização

O processo de simplificação do plano da imagem sob forma de uma malha regular finita, cujos elementos têm um tamanho fixo e constante (pixels), resulta em uma representação mais fácil de armazenar.

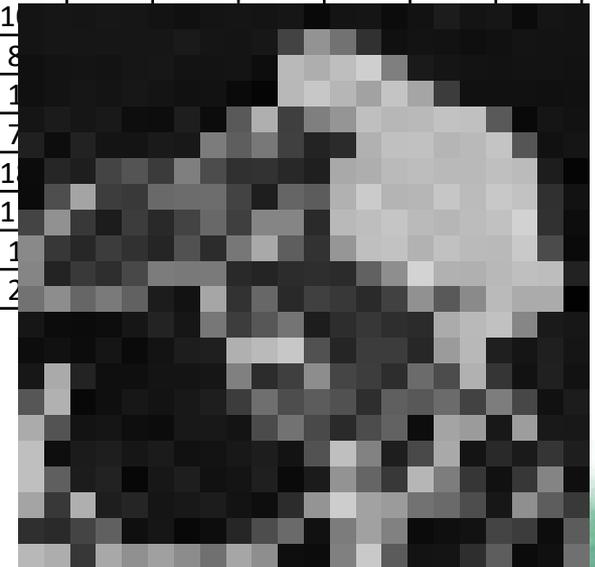
O plano da imagem é representado sob forma de uma matriz.



A radiação incidente em cada célula é medida por um elemento detector.

# Uma imagem, Uma matriz

20	21	21	22	22	19	16	20	21	20	22	9	20	21	12	18	28	19	22	11	21	19
20	22	21	23	21	23	26	21	20	23	67	147	114	49	16	19	18	15	17	20	19	19
16	20	20	19	22	23	19	19	20	12	185	174	191	207	127	26	21	19	18	19	19	17
16	19	22	20	23	20	17	19	10	5	184	199	181	163	196	165	60	17	17	17	16	17
22	27	22	25	13	13	30	11	89	176	63	126	149	190	183	186	193	192	89	9	21	18
32	13	34	20	20	25	23	125	95	119	64	35	43	177	192	193	181	185	197	85	17	21
12	38	30	68	84	60	127	76	47	50	41	31	169	174	185	190	184	186	191	186	29	5
11	78	164	61	57	105	108	109	67	30	101	92	177	202	180	182	198	187	200	195	48	18
67	145	56	28	61	41	67	104	63	134	133	42	185	190	197	186	182	188	193	210	50	13
137	55	39	60	49	37	82	44	119	169	94	50	150	191	194	178	193	186	184	201	74	9
133	35	57	46	73	125	120	122	41	35	44	46	43	97	143	211	178	176	185	190	189	34
114	141	102	122	97	27	18	167	50	103	41	63	56	43	65	145	89	138	186	171	171	2
22	12	12	13	22	34	22	118	61	87	116	30	45	55	46	44	171	185	193	134	27	23
13	17	12	22	10	19	29	31	176	187	199	80	37	60	60	38	156	185	30	21	31	21
23	170	35	15	15	19	20	22	127	44	63	140	70	61	45	1						
86	178	8	14	23	19	24	30	60	111	77	92	81	46	95	8						
172	83	18	21	14	12	23	24	20	74	114	72	42	76	98	1						
191	14	27	30	21	27	17	19	24	29	19	82	191	130	30	7						
191	95	28	34	8	23	30	17	20	30	10	27	147	101	56	1						
164	56	175	30	37	20	23	28	19	14	44	149	205	161	155	1						
47	42	67	96	15	22	9	13	37	77	106	17	124	162	129	1						
184	172	55	168	143	160	140	112	167	142	14	12	128	200	91	2						



A imagem pode ser entendida como uma matriz  
 $F(x,y)$

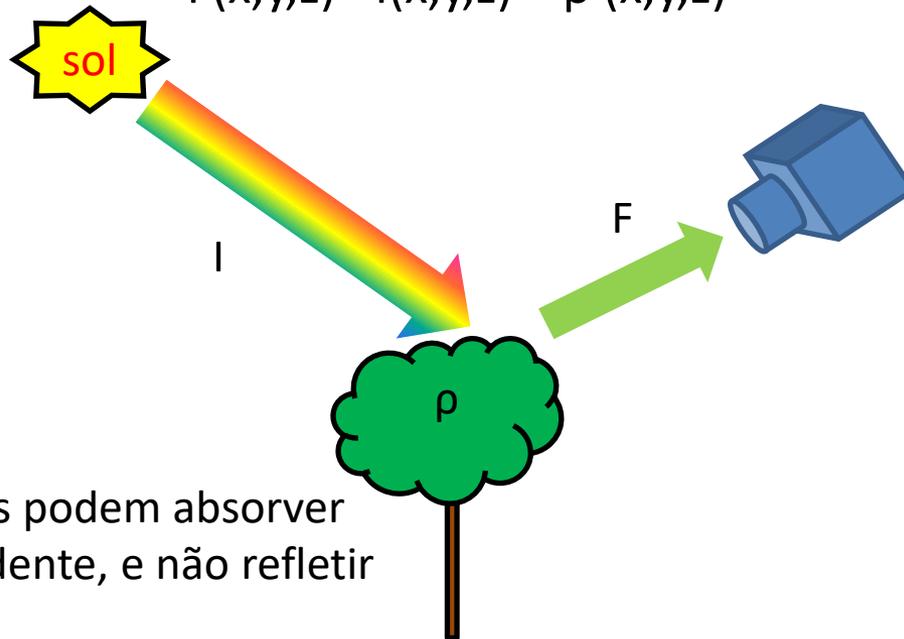
- 
- Aspecto Radiométrico
    - Quantização
- 

# A luz incidente

Em uma fotografia,  $F(x,y)$  registra a luz refletida pelos objetos. A intensidade da radiação incidente no sensor está associada a duas grandezas físicas:

- intensidade de energia incidente no **objeto** ( $I$ ) e
- **reflectância** do objeto ( $\rho$ ), ou sua capacidade de refletir a radiação incidente.

- $$F(x,y,z) = I(x,y,z) * \rho(x,y,z)$$



Algumas superfícies podem absorver toda a energia incidente, e não refletir nada.

Como as imagens são obtidas em determinadas faixas espectrais, esta relação deveria ser melhor explicitada em função do comprimento de onda da energia utilizada para obter a imagem, ou seja, a faixa espectral à qual o sensor ou filme é sensível:

$$F_{\lambda}(x,y,z) = I_{\lambda}(x,y,z) * \rho_{\lambda}(x,y,z)$$

Com

$$0 \leq I_{\lambda} < \infty$$

$$0 \leq \rho_{\lambda}(x,y) \leq 1$$

A reflectância ( $\rho$ ) é a grandeza física que mede a proporção da energia incidente que o objeto é capaz de refletir num comprimento de onda determinado. Ela é uma propriedade intrínseca da superfície e depende da natureza e estado do objeto.



Red

Green

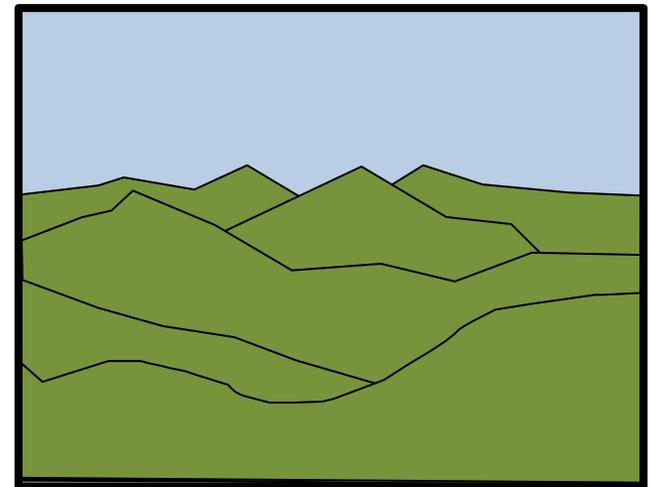
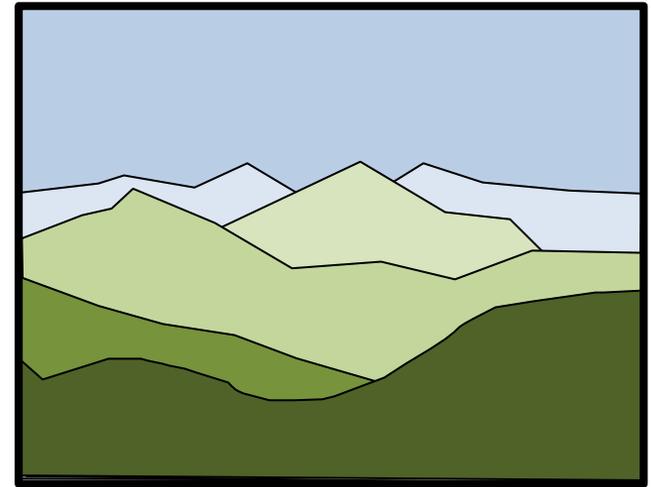
Blue



Outros fatores são responsáveis pela atenuação da energia, como por exemplo o efeito da atmosfera ou a opacidade do sistema de lentes utilizado para captar a imagem, uma função de atenuação (A).

- $F = A * I * \rho$

É comum, encontrar os efeitos da atmosfera em fotografias aéreas tiradas de grande altitude, ou em imagens de satélite. Mas você também pode notar isto ao ver uma cadeia montanhosa à distância.



# Aspecto radiométrico

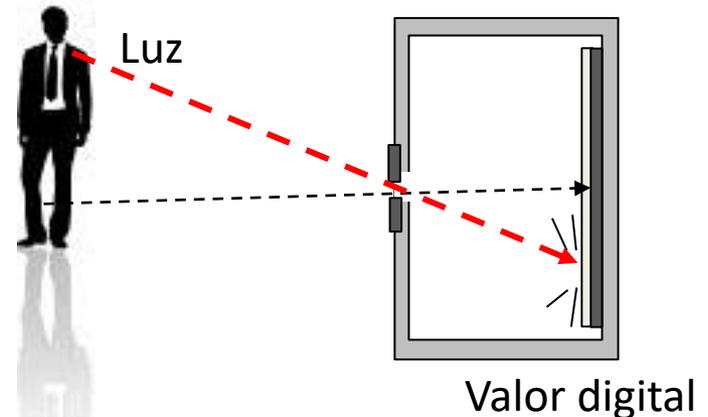
## Luz -> valor digital

Está associado à maneira como o brilho ou a energia proveniente de um objeto ou superfície é registrado e a fidelidade com a que ele é representado na imagem.

A radiação incidente em cada sensor é medida e deve ser armazenada. Como armazenar esta grandeza?

Esta transformação é chamada de **quantização**

Distância focal



## Aspecto radiométrico

# Quantização

- A energia incidente no sensor é uma grandeza contínua. Sua completa representação (de forma digital) demandaria o uso de muita memória.
- Na prática costuma-se armazenar apenas uma quantidade finita, mas suficientemente grande, de níveis de energia em lugar de todos os valores possíveis. É feita então uma segunda discretização, desta vez nos valores armazenados nos pixels.

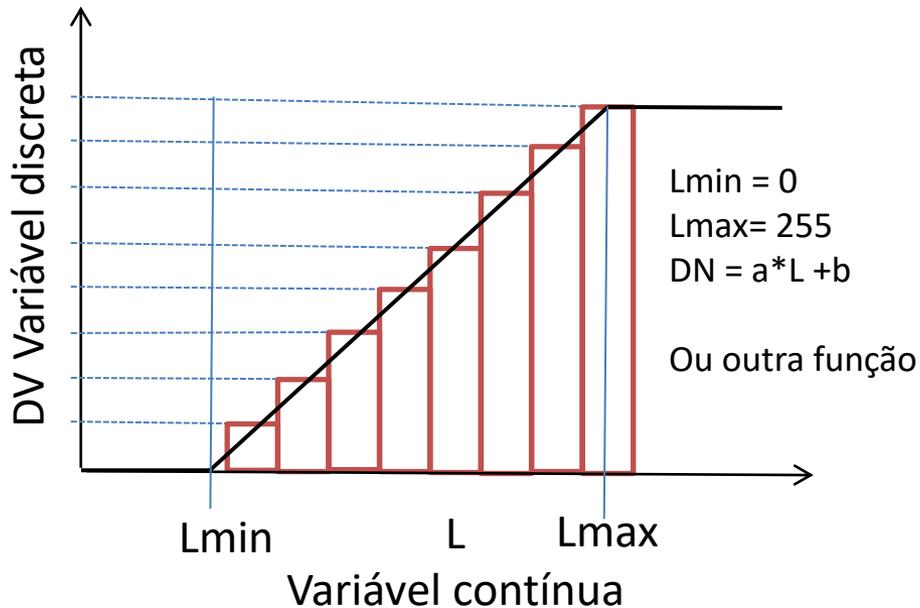
120.023	121.081	121.301	121.999	121.901
120.123	122.234	121.435	123.491	120.917
116.010	120.001	120.011	119.034	122.222
116.112	119.238	122.431	120.318	123.198
122.187	127.236	122.345	125.458	112.998
132.000	113.100	134.201	120.371	119.890
112.333	138.222	130.234	168.012	184.021
111.111	177.999	164.000	161.442	157.020

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	27	22	25	13
32	13	34	20	20
12	38	30	68	84
11	78	64	61	57

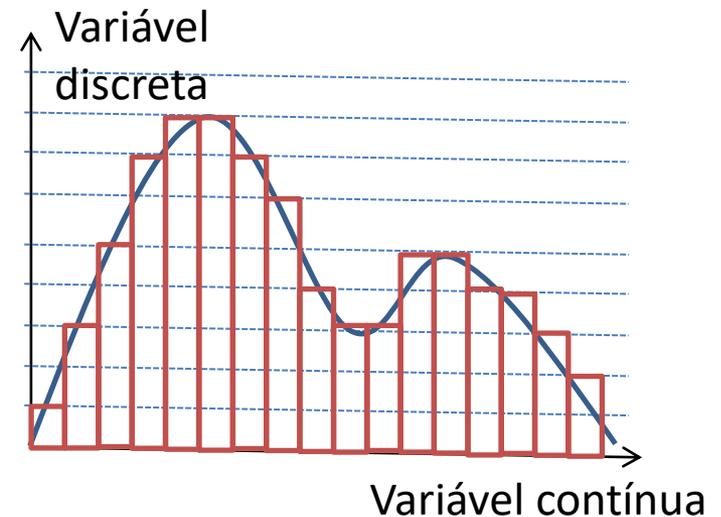
# Quantização

É definida uma escala discreta de valores, com valores  $L_{min}$  e  $L_{max}$  fixos e uma quantidade de valores intermediários possíveis.

Transformação



exemplo



A geometria não é alterada, mas a maneira como a energia incidente sim.

A quantidade de valores possíveis é chamada de resolução radiométrica e, na prática, é definida pelo número de bits usados para armazenar a informação. Geralmente, são usados 8 bits para cada pixel, o que corresponde a um total de 256 diferentes níveis possíveis entre zero (ausência de energia incidente) e 255 (saturação do sensor).



8 bits (256 DN)



3 bits (8 DN)



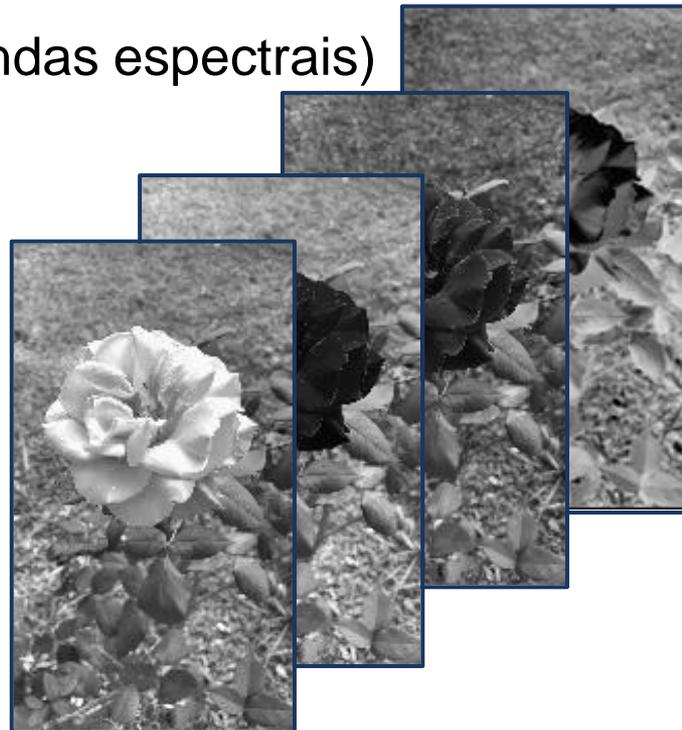
1 bit (2 DN)

Quanto maior o número de níveis (número de bits), mais tonalidades podem ser armazenados, e a informação é aumentada.

Por outro lado, aumentando o número de bits por pixel cresce o tamanho da memória necessário para armazenar e processar a imagem.

# Resolução

- A resolução de uma imagem é medida em termos de sua capacidade de representar a informação da cena.
- Resolução espacial (tamanho do pixel na superfície do objeto)
- Resolução radiométrica (número de bits usados para representar a informação)
- Resolução espectral (bandas espectrais)



# Uma imagem simples

- Imagem de uma única banda.
- Os valores dos pixels são armazenados de maneira seqüencial. Começando no pixel superior esquerdo e terminando no pixel inferior direito, efetuando a varredura da imagem de maneira seqüencial. estrutura espacial implícita, pois, conhecendo-se o tamanho da imagem formada pelos dados, é facilmente reconstruída a imagem.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Imagem tamanho  $N \times M$  é armazenada como uma série de  $N * M$  valores

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

# Processamento de imagens

Entrada: Uma imagem, parâmetros

Saída, pode ser

- Uma imagem
- Um valor
- Um vetor

$A = \text{função}(I, \text{param})$

# Níveis de processamento

Global

Pixel

Local

Objeto

# Global

Consiste derivar um valor, ou uma série de valores, de toda a imagem.

Por exemplo,

- A média dos valores digitais de uma imagem.
- O histograma de uma imagem.

Ex:

$H = \text{histograma}(I)$

# Global

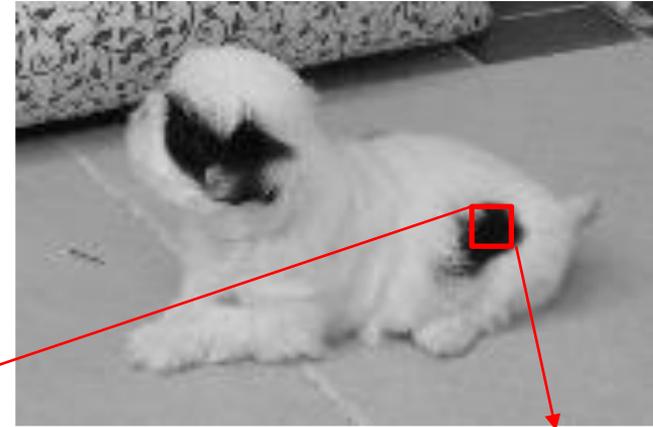
Cada pixel tem um valor digital

Do conjunto de pixels da imagem, podemos calcular estatísticas.

Por exemplo, para uma imagem em nível de cinza (1 banda) Podemos calcular:

- Média (mean);
- Valor Mínimo, máximo (min/max)
- Mediana (median)
- Variância (variance)

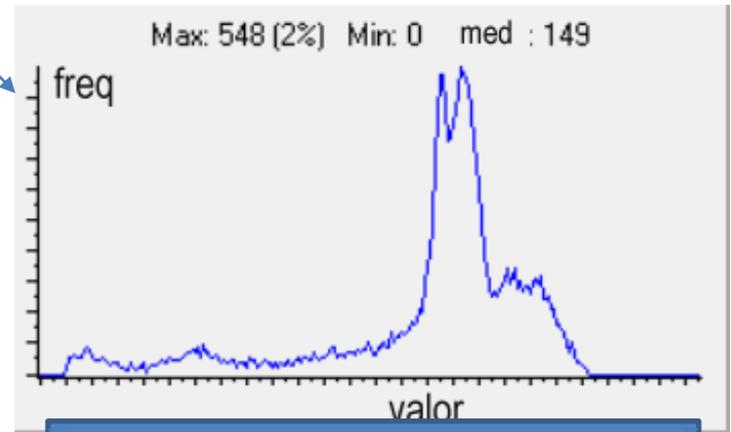
Ou Podemos calcular e representar seu histograma... o que é um histograma?



10	10	11	11	11	10	11	10	10	11	10	11
11	10	9	9	9	8	8	7	8	10	10	10
10	10	6	2	2	1	1	1	1	9	10	10
11	10	5	2	2	3	3	3	1	8	10	10
10	10	6	3	7	10	10	6	3	7	10	10
10	10	7	3	6	10	9	8	3	6	10	10
10	10	9	1	5	9	1	4	1	6	9	10
9	10	9	1	4	9	1	4	1	6	9	10
10	10	9	1	3	8	7	5	1	6	9	10
10	10	10	1	1	2	2	1	0	6	8	10
10	10	10	6	6	6	6	6	6	6	8	10
10	10	10	10	8	9	9	9	9	10	10	9

# O Histograma

Representação da frequência relativa dos valores digitais na imagem.



A faixa do histograma depende da resolução radiométrica.

# Contar

Contar a ocorrência de cada valor na imagem

```
H=np.zeros( (256))
for i in range(nlinhas):
    for j in range(ncolunas):
        v=int( I[i,j] )
        H[v]=H[v]+1
plt.plot(H)
plt.title("Histograma da imagem")
plt.legend()
plt.show()
```

10	10	11	11	11	10	11	10	10	11	10	11
11	10	9	9	9	8	8	7	8	10	10	10
10	10	6	2	2	1	1	1	1	9	10	10
11	10	5	2	2	3	3	3	1	8	10	10
10	10	6	3	7	10	10	6	3	7	10	10
10	10	7	3	6	10	9	8	3	6	10	10
10	10	9	1	5	9	1	4	1	6	9	10
9	10	9	1	4	9	1	4	1	6	9	10
10	10	9	1	3	8	7	5	1	6	9	10
10	10	10	1	1	2	2	1	0	6	8	10
10	10	10	6	6	6	6	6	6	6	8	10
10	10	10	10	8	9	9	9	9	10	10	9

# Nível de Pixel

- Consiste em alterar o valor de um pixel baseado apenas no seu valor original.
- Um exemplo é a limiarização.

IF  $F(x,y) > 100$ :

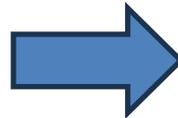
$S(x,y) = 1$

Else:

$S(x,y) = 0$

F

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	127	22	25	13
32	13	134	120	120
12	38	130	168	184
11	78	164	161	157



S

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	1	0	0	0
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1

## Nível local

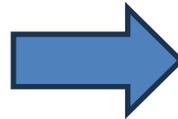
- Consiste em alterar o valor de um pixel baseado no seu valor e os valores de seus vizinhos.
- Um exemplo é o cálculo da média “local”, considerando uma vizinhança 3x3.

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	127	22	25	13
32	13	134	120	120
12	38	130	168	184
11	78	164	161	157

# Nível de Pixel

```
for i in range(1:nlinhas-1):  
    for j in range(1:ncolunas-1):  
        s=0  
        for ii in range(i-1:i+2):  
            for jj in range(j-1:j+2):  
                v=int( I[ii,jj] )  
                s=s+v  
  
        m=int(s/9)  
        J(i,j)=m
```

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	127	22	25	13
32	13	134	120	120
12	38	130	168	184
11	78	164	161	157



		33		

$$(20+20+19+19+22+20+127+22+25)/9 = 32.6 = 33$$

# Nível de região

Consiste em alterar o valor de um pixel baseado em todos os valores de uma determinada região. Para isto, a região deve estar definida, digamos, em uma segunda imagem.

Por exemplo, na imagem B, calcular a média da região identificada por pixels com valor 1 na imagem A

1	1			
	1	1	1	
	1	1	1	
	1	1	1	
		1	1	1
	1	1	1	

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	127	22	25	13
32	13	134	120	120
12	38	130	168	184
11	78	164	161	157

# região

```
S=0
N=0
for i in range(nlinhas):
    for j in range(ncolunas):
        if A(i,j)==1:
            v=int( B[i,j] )
            S=S+v
            N=N+1
m=int(S/N)
```

1	1			
	1	1	1	
	1	1	1	
	1	1	1	
		1	1	1
	1	1	1	

A

20	21	21	22	22
20	22	21	23	21
16	20	20	19	22
16	19	22	20	23
22	127	22	25	13
32	13	134	120	120
12	38	130	168	184
11	78	164	161	157

B

- Você pode fazer pequenos programas para criar e manipular imagens em Python...



pyzo

+ Código + Texto

programa para ler imagem e calcular histograma

Comentários (Texto)

BASICOS

```

#basics
import matplotlib.pyplot as plt
import numpy as np
from skimage import io

from google.colab import drive
drive.mount('/content/drive')

```

AQUI VOCE ESCREVE SEU CODIGO

Mounted at /content/drive

mostra os resultados de seu código e os erros

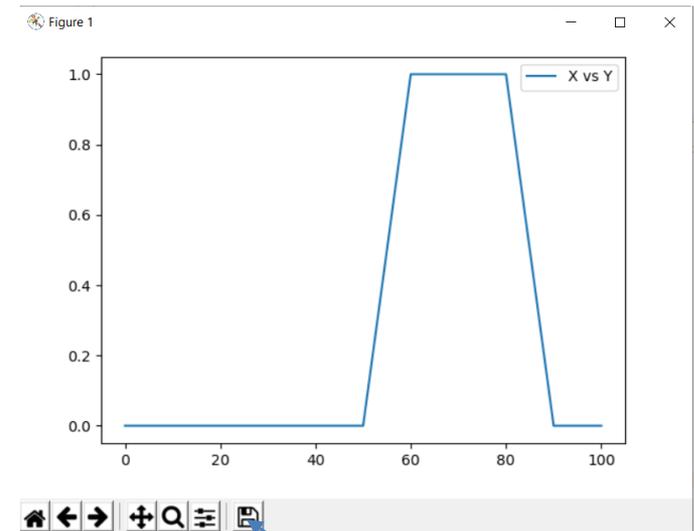
Listar imagens para ver se está com a imagem em seu drive e depois ler

# Remarks

- Em Python, comentário começa com "cerquinha" #
- Ex:
- # inicio de programa
- Geralmente você inicia o Python básico, sem muitos recursos, mas como ele é modular, você pode e deve carregar as bibliotecas que for usar. Existem muitas. Isto é feito usando o comando **import**.
- Duas bibliotecas básicas são:
- **NumPy**: pacote que suporta arrays e matrizes multidimensionais, possui uma série de funções matemáticas para trabalhar com estas estruturas.
- **Matplotlib** pacote para apresentar gráficos e figuras na tela, junto com sua extensão **pyplot**.

# Um exemplo simples grafico x vs y

- `import numpy as np`
- `import matplotlib.pyplot as plt`
- `# definir dois vetores x e y`
- `x=[0, 50, 60, 80, 90, 100]`
- `y=[0, 0, 1, 1, 0, 0]`
- `# plotar o gráfico x vs y`
- `plt.plot(x,y)`
- `plt.show()`
- `# ou com legenda...`
- `plt.plot(x,y, label='X vs Y')`
- `plt.legend()`
- `plt.show()`

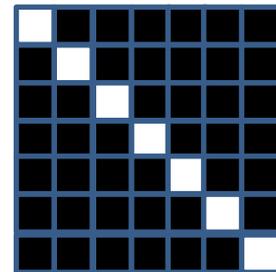
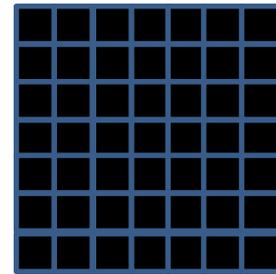


Aqui tem opções para “salvar” a imagem, dar zoom, etc

# Manipular imagens com PY

Vamos escrever um pequeno programa em PY para entender como se mudam os valores de uma imagem (uma matriz)

1. Criamos uma matriz vazia (ex. 7x7) com valores todos zero (preto)
2. Mudaremos os valores da diagonal principal para 255



Escreva o código com a ajuda a seguir e execute seu programa

# PROGRAMA básico - criar uma imagem

```
# # primeiro: incluir algumas bibliotecas básicas
import numpy as np
import matplotlib.pyplot as plt

print('#####')
print('# Programa inicial : criando uma imagem #')
print('#####')
```

**Import:** carrega bibliotecas úteis, neste caso:

Numpy ... Para processar números

Matplotlib ... Para processar e visualizar gráficos

PYthon permite imprimir textos com "print"

## Continuando...

```
# criamos matriz vazia (com valores reais
FLOAT, tamanho 17 x 17 pixels)
I= np.zeros( (17,17) )
# ler as dimensões da matriz, caso não
conheça
n1, nc = I.shape
# imprimimos os valores número de linhas e
colunas
print("Linhas=", n1, " Colunas=", nc)
```

# continuando...

```
# Mostrar a matriz na tela
```

```
plt.imshow(I, cmap='gray', vmin=0, vmax=255)  
plt.show()
```

```
# imshow()= cria uma imagem a partir de uma matriz.
```

```
Argumentos:
```

- I: matriz,
- Cmap: mapa de cores, neste caso em nível de cinza 'gray',
- Vmin,vmax: faixa de variação dos valores em I a serem vistos, neste caso entre 0 a 255 [8 bits]

```
# show() mostra o resultado na tela
```

# Agora temos como criar uma imagem

```
# incluir algumas bibliotecas básicas
import numpy as np
import matplotlib.pyplot as plt
# criamos matriz vazia
I= np.zeros( (17,17) )
# ler as dimensões da matriz, caso não conheça
nl, nc = I.shape
# imprimimos os valores número de linhas e colunas
print("Linhas=", nl, " Colunas=", nc)
```

# Continue

```
# Toda a imagem está cheia de zeros, vamos mudar os
valores da diagonal para 255
for L in range(n1):
    I[L,L]= 255
# arredondamos e transformamos a matriz a uint8 para
armazenar como imagem
P= np.uint8( np.round( I ) )

# salvar imagem como"saida.pgn"
plt.imsave('saida.png',P,cmap='gray')

# Mostrar o resultado
imgplot = plt.imshow(P, cmap='gray', vmin=0,
vmax=255)
```