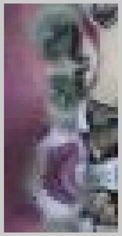
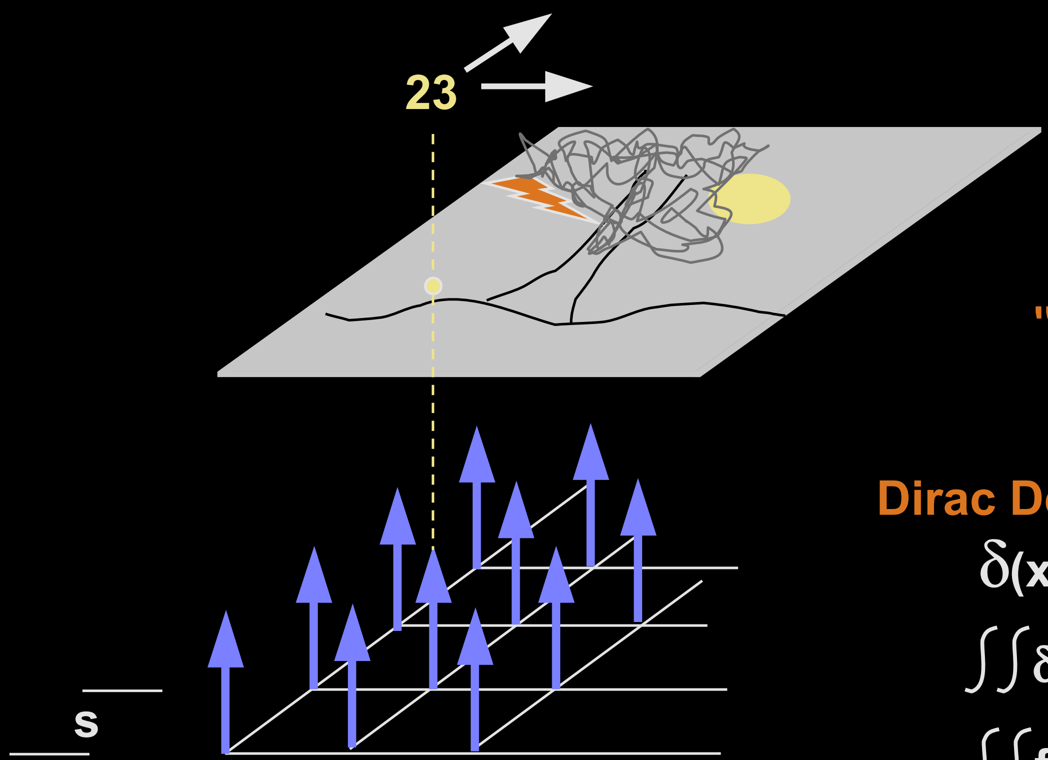
Human Eye Sampling

Cartesian image ----- Log-Polar representation ----- Retinal representation



- Rough Idea: Ideal Case



"Digitized Image"

"Continuous Image"

Dirac Delta Function 2D "Comb"

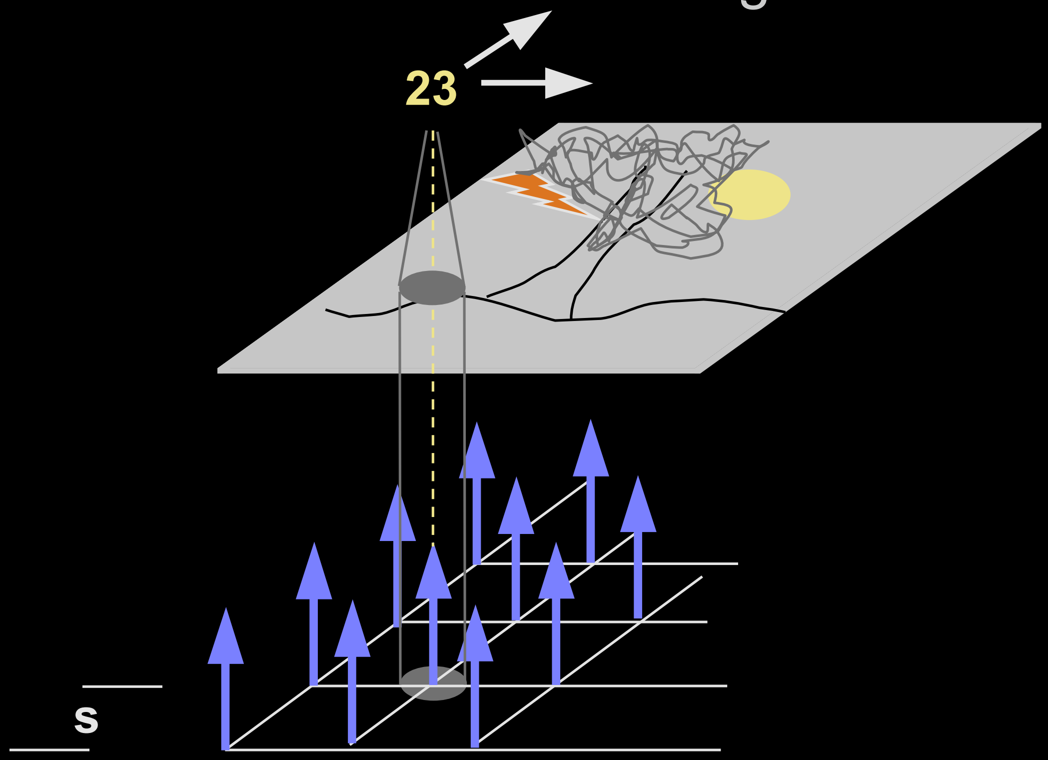
$\delta(x,y) = 0$ for $x \neq 0, y \neq 0$

$\iint \delta(x,y)\, dx\, dy = 1$

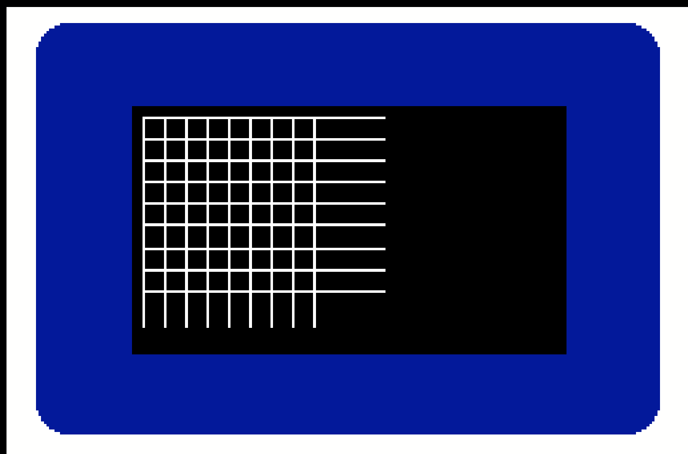
$\iint f(x,y)\delta(x-a,y-b)\, dx\, dy = f(a,b)$

$\delta(x-ns,y-ns)$ for $n = 1\ldots.32$ (e.g.)

- Rough Idea: Actual Case
  - Can't realize an ideal point function in real equipment
  - "Delta function" equivalent has an area
  - Value returned is the average over this area



Projection through a pixel



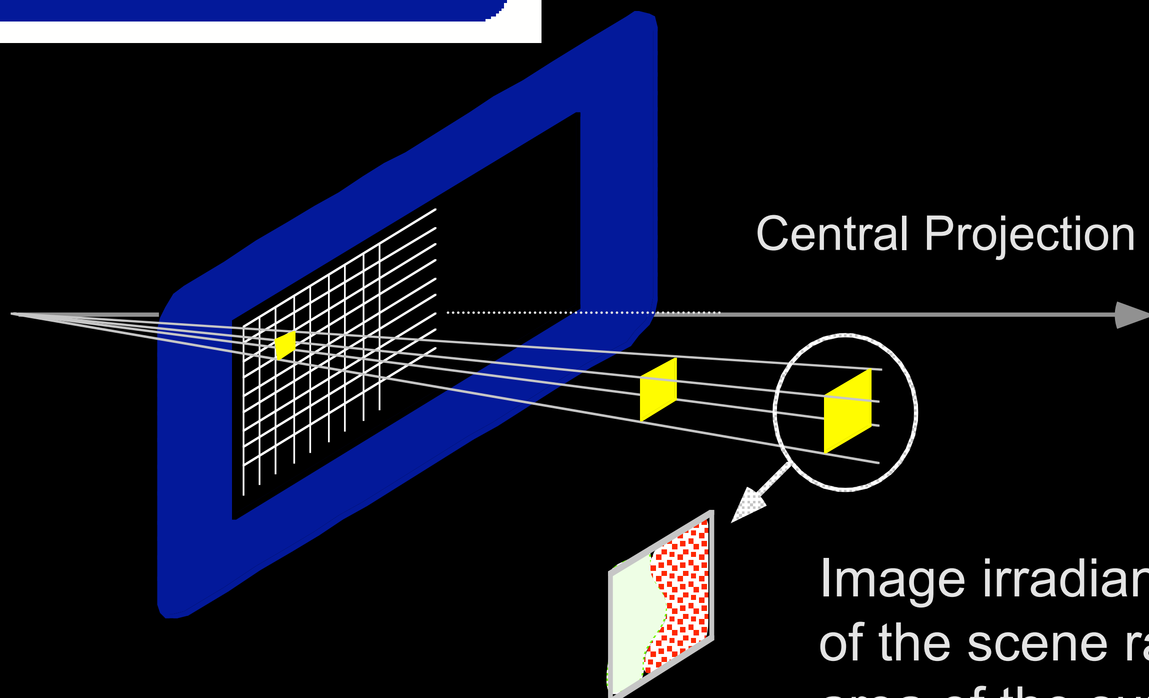
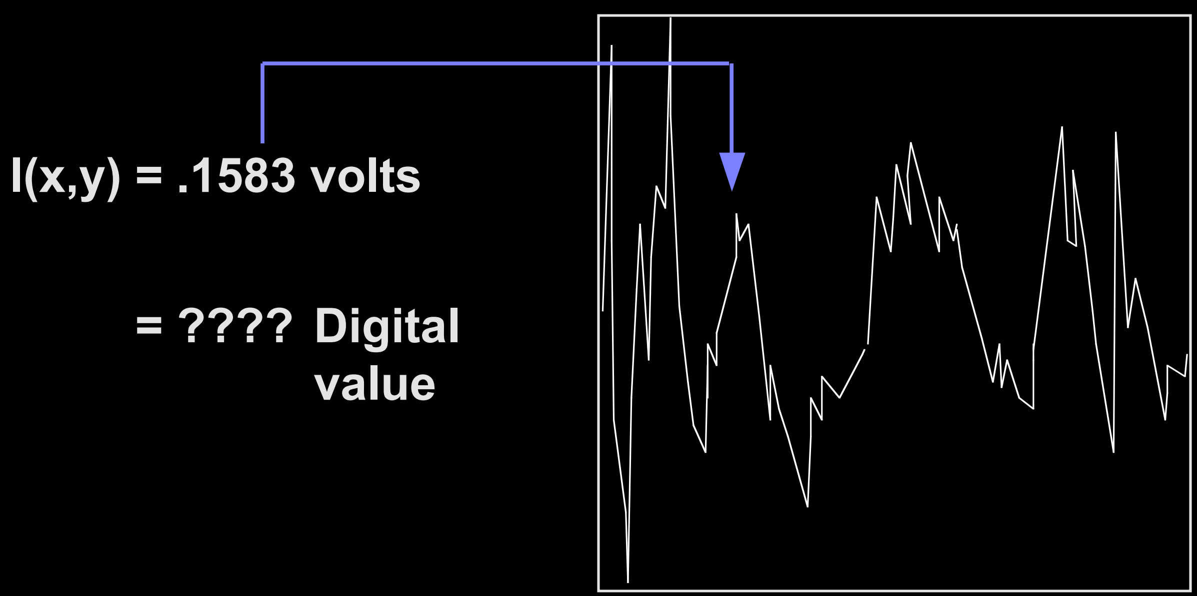


Image irradiance is the average of the scene radiance over the area of the surface intersecting the solid angle!

Mixed Pixel Problem

Signal Quantization

- Goal: determine a mapping from a continuous signal (e.g. analog video signal) to one of K discrete (digital) levels.



Quantization

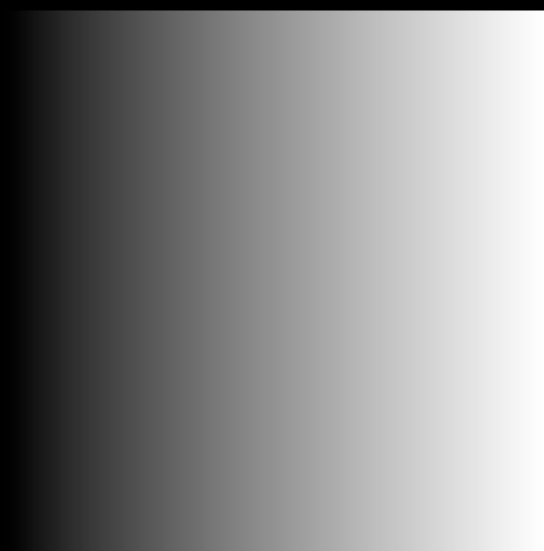
- $I(x,y)$ = continuous signal: $0 \le I \le M$
- Want to quantize to K values 0,1,....K-1
- K usually chosen to be a power of 2:

K: #Levels	#Bits
2	1
4	2
8	3
16	4
32	5
64	6
128	7
256	8

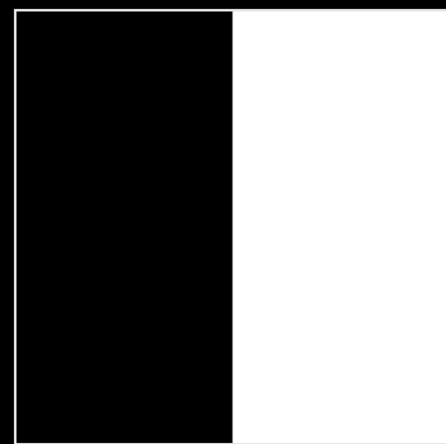
- Mapping from input signal to output signal is to be determined.
- Several types of mappings: uniform, logarithmic, etc.

Choice of K

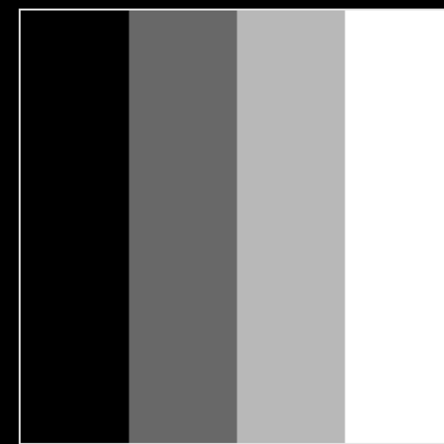
Original



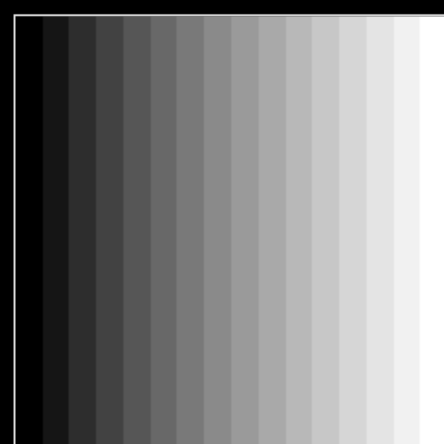
Linear Ramp



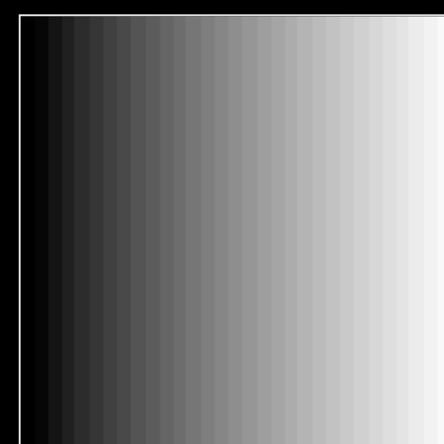
K=2



K=4

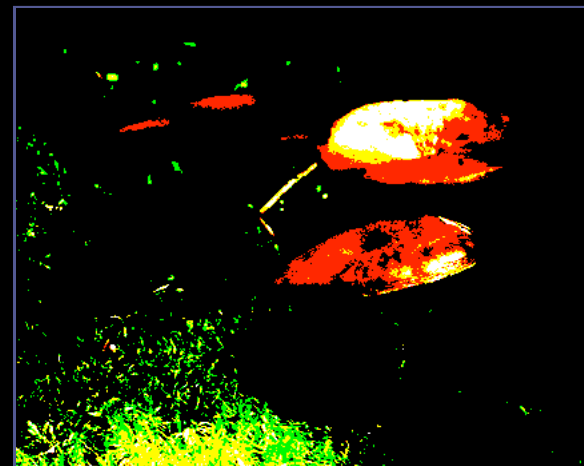


K=16



K=32

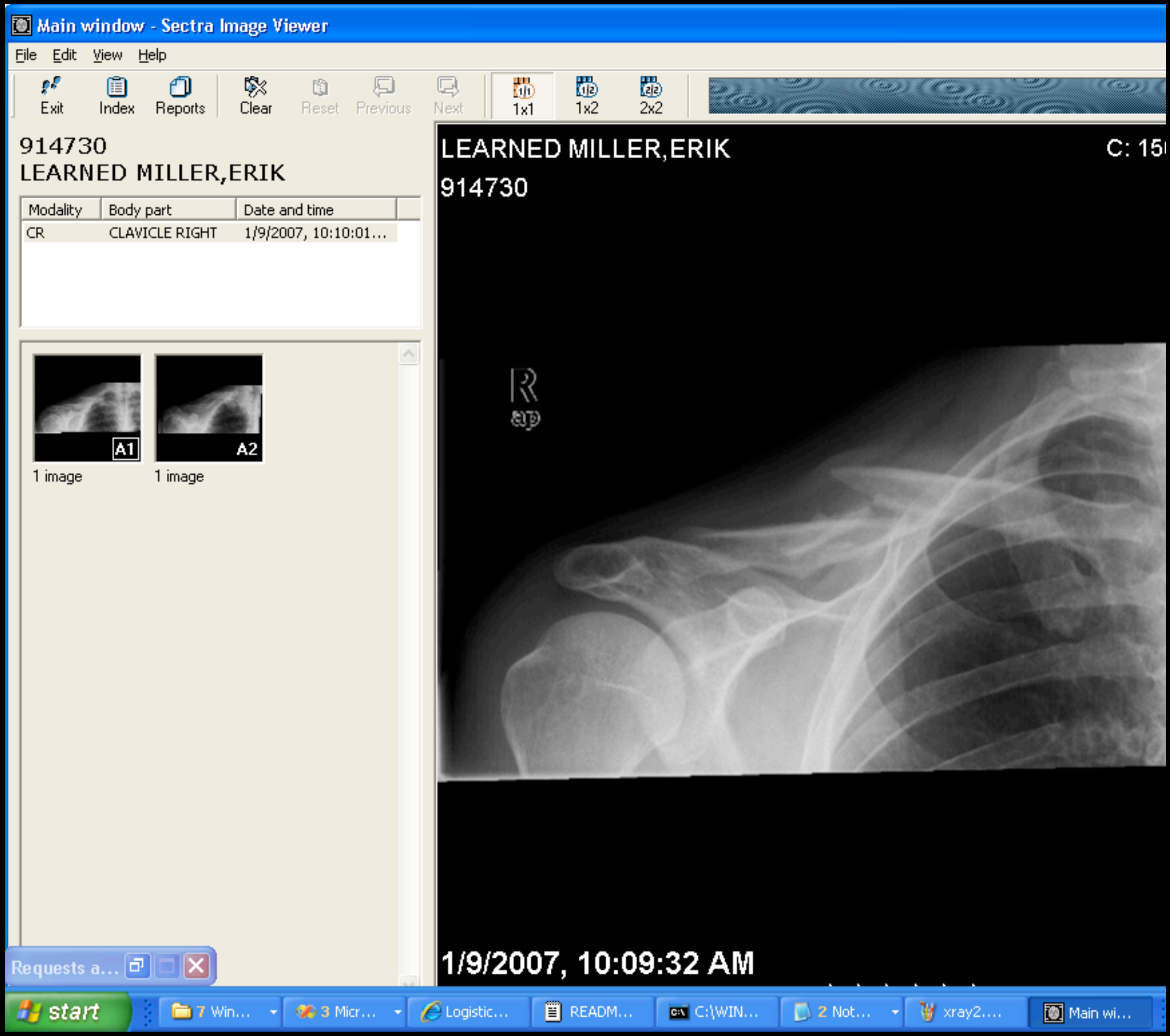
Choice of K



K=2 (each color)

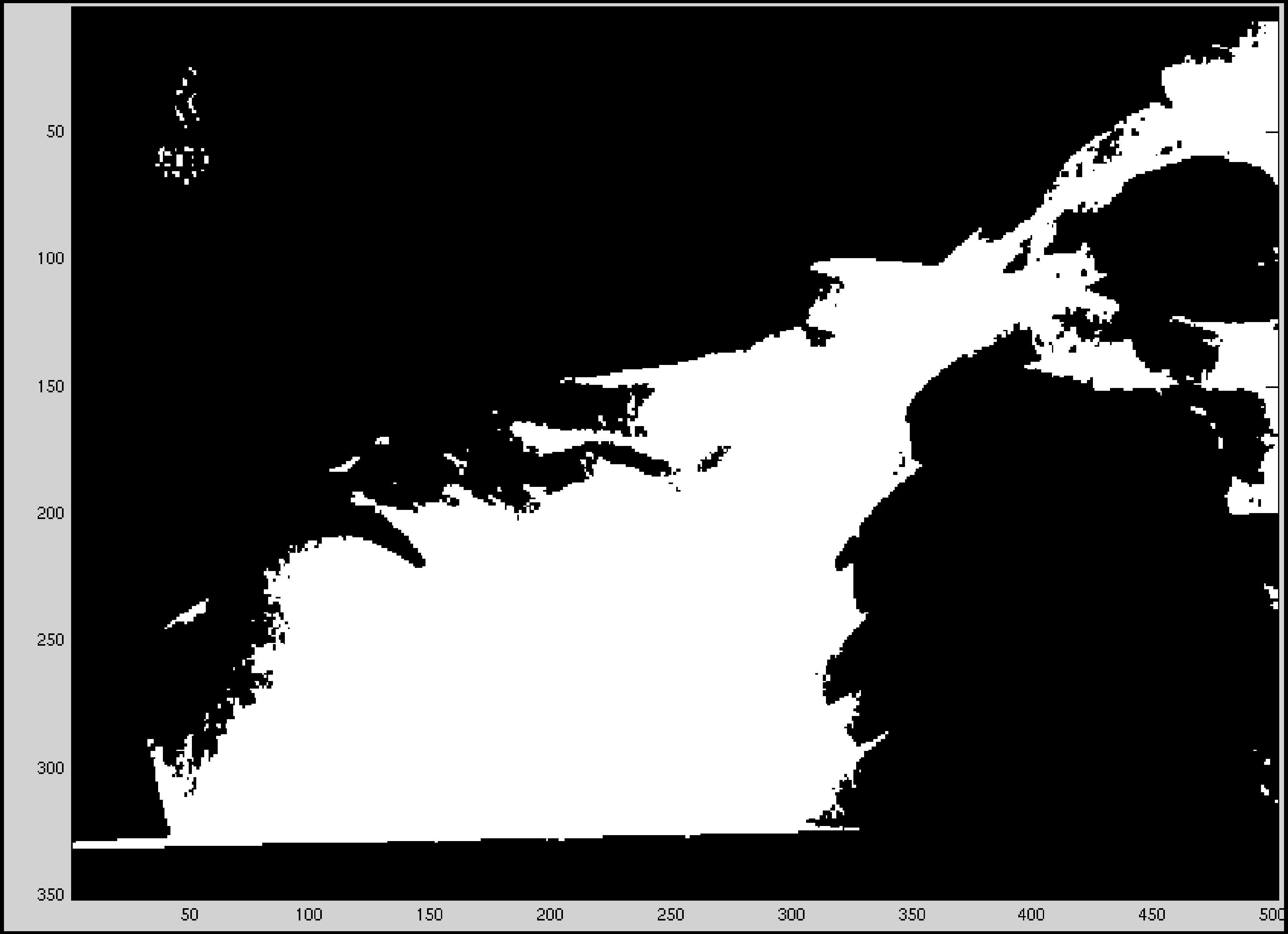
K=4 (each color)

Digital X-rays

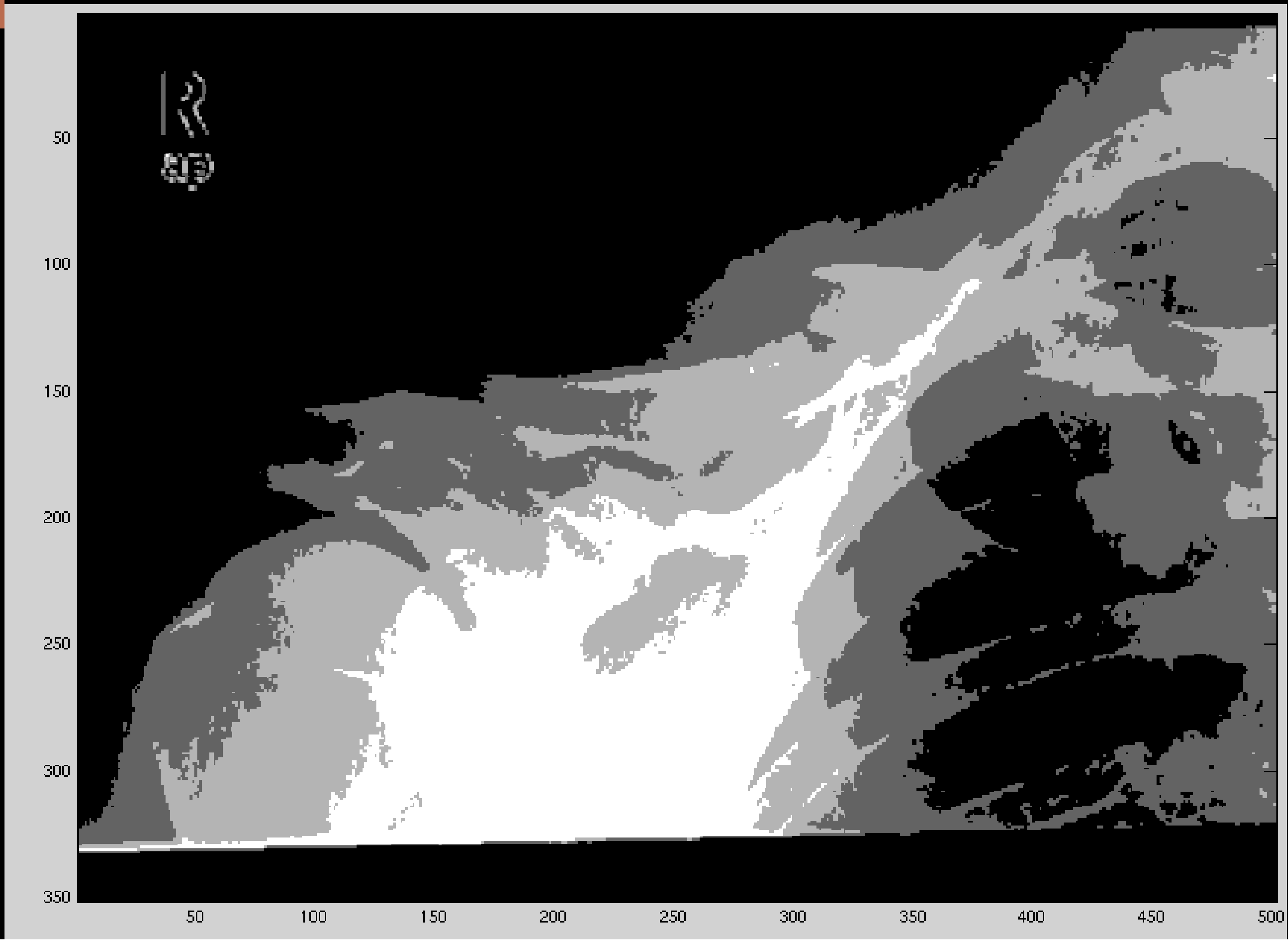


Digital X-rays: 8 is enough?

Digital X-rays: 1 bit



Digital X-rays: 2 bits

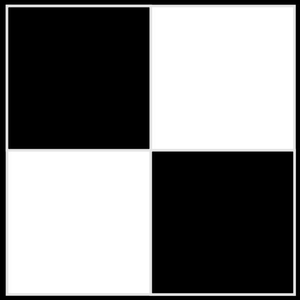


Digital X-rays: 3 bit

Digital X-rays: 8 is enough?

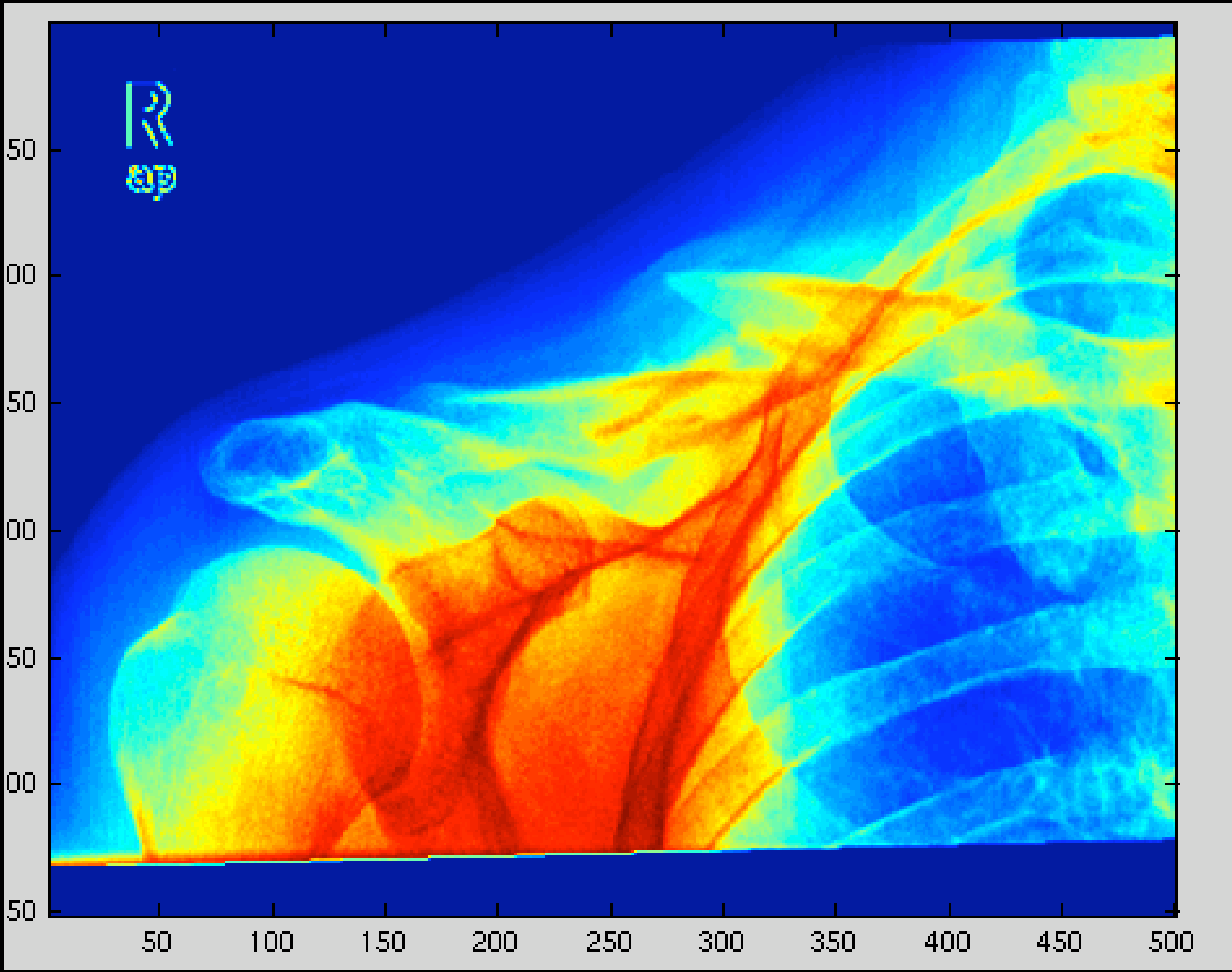
Gray Levels-Resolution Trade Off

- More gray levels can be simulated with more resolution.
- A “gray” pixel:

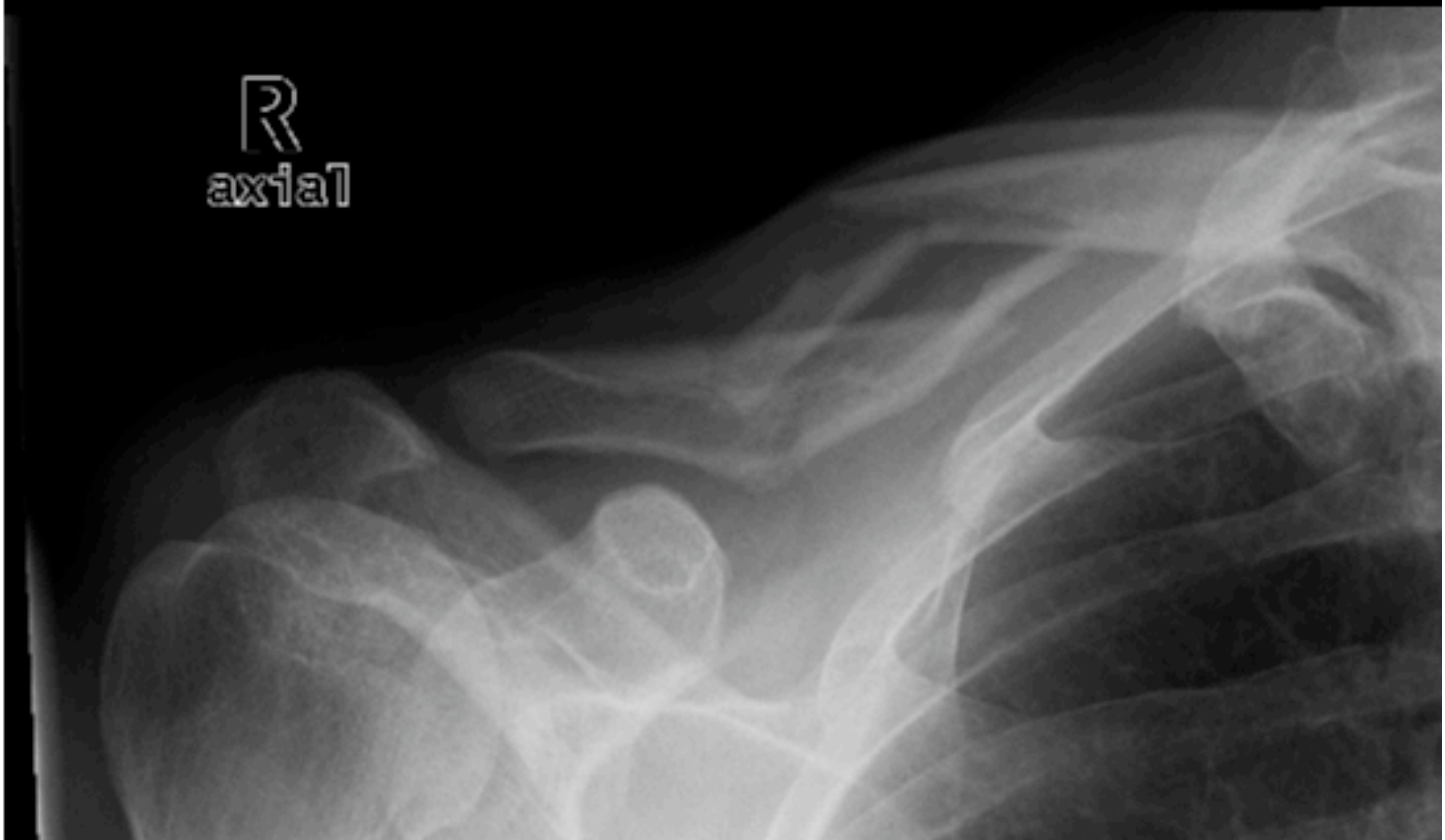


- Doubling the resolution in each direction adds at least four new gray levels. But maybe more?

Pseudocolor



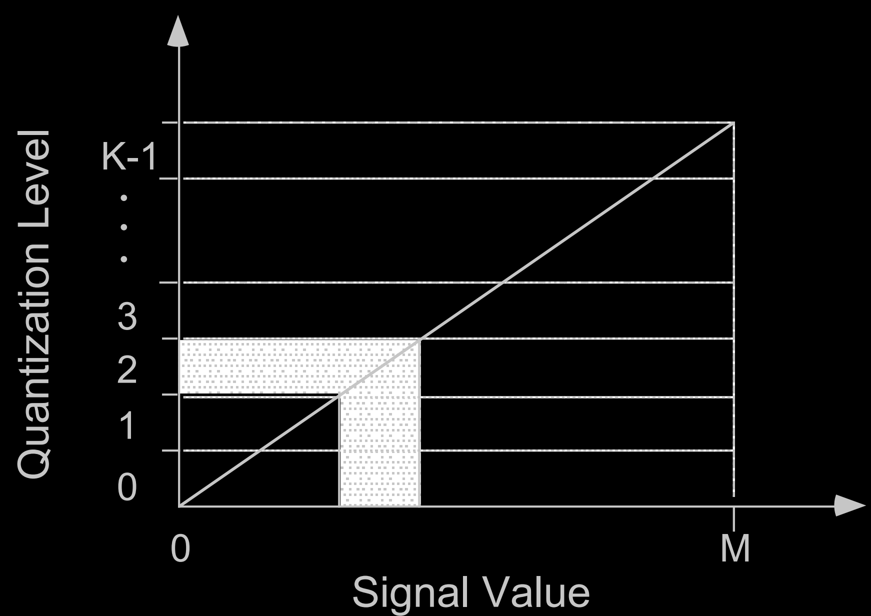
Digital X-rays: 8 is enough?



MRI

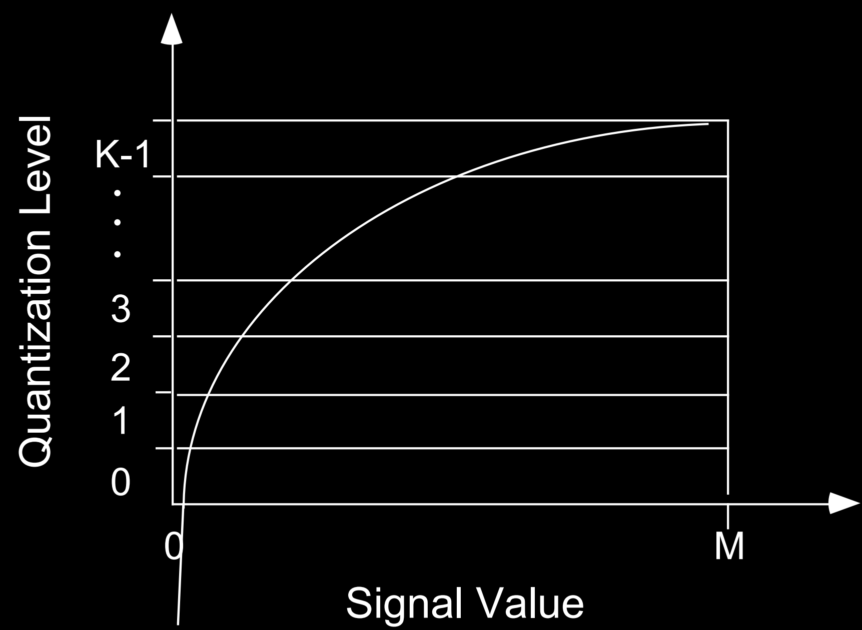
Choice of Function: Uniform

- Uniform sampling divides the signal range [0-M] into K equal-sized intervals.
- The integers 0,...K-1 are assigned to these intervals.
- All signal values within an interval are represented by the associated integer value.
- Defines a mapping:



Logarithmic Quantization

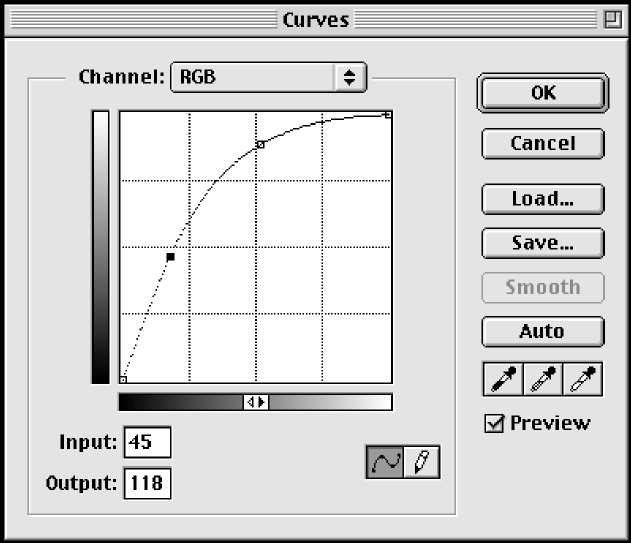
- Signal is log I(x,y).
- Effect is:



- Detail enhanced in the low signal values at expense of detail in high signal values.

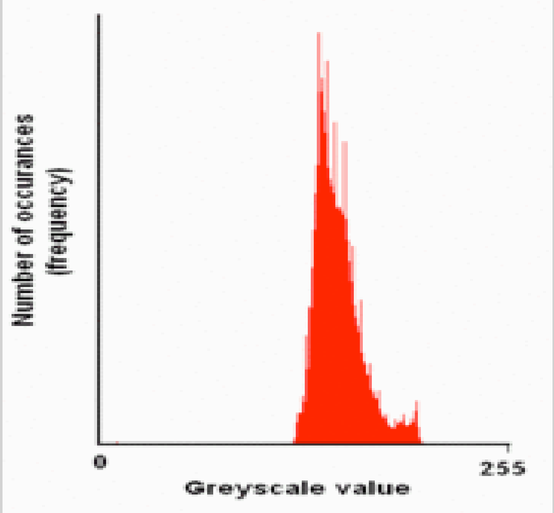
Logarithmic Quantization

Quantization Curve



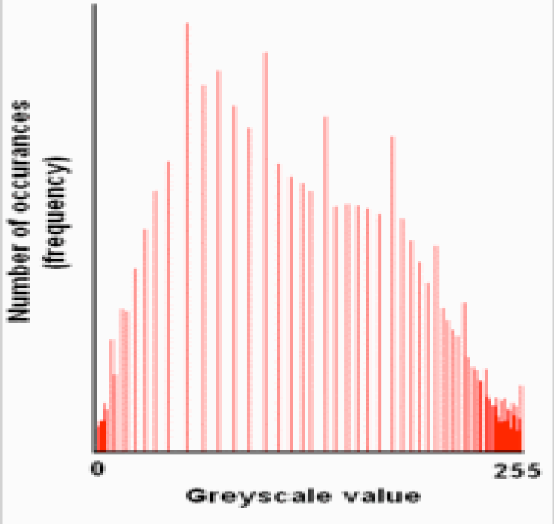
Histogram Equalization

An unequalized image



Corresponding histogram

Same image after histogram equalization



Corresponding histogram

Brightness Equalization

- Two methods:
  - Change the data (histogram equalization)
  - Use a look up table (brightness or color remapping)

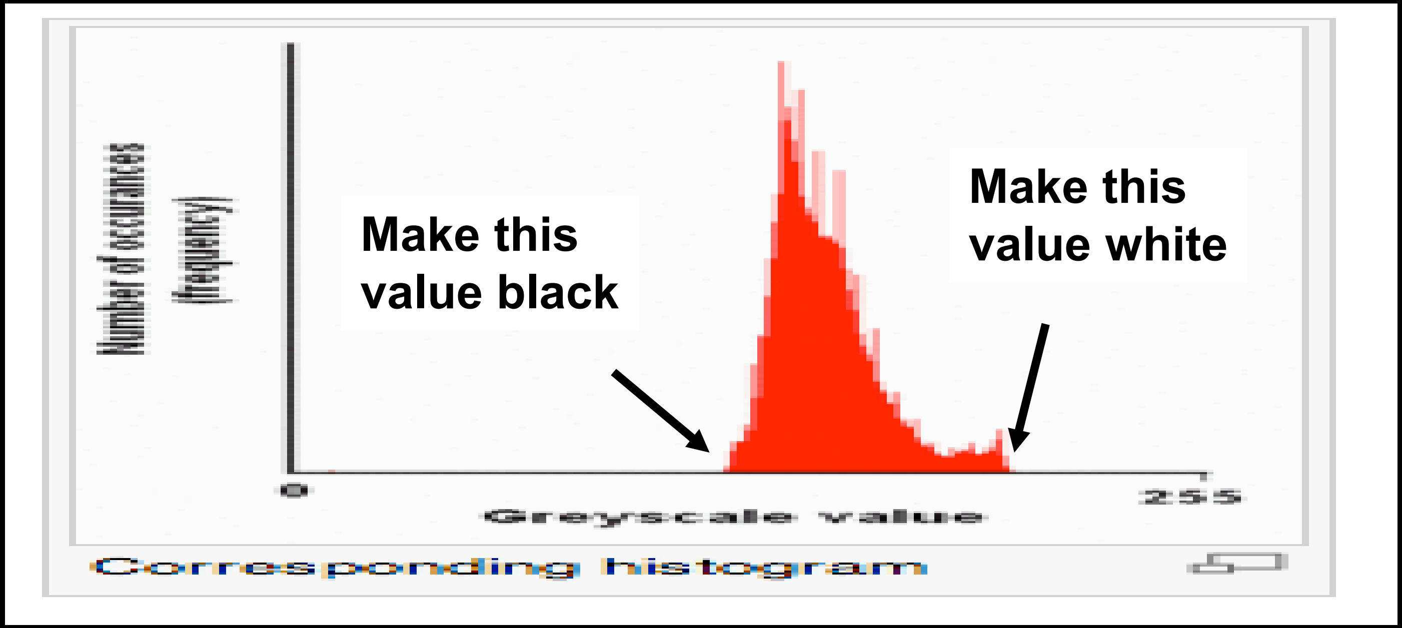
Look up tables

Maps Brightness Value -> RGB Color

- 0 -> (1, 0, 0)
- 1 -> (0, 1, 0)
- 2 -> (0, 0, 1)
- 3 -> (0 , 1, 1)
- ...
- 255 -> (1, 1, 1)

Brightness Equalization

- Two methods:
  - Change the data.
  - Use a look up table.



Look up tables

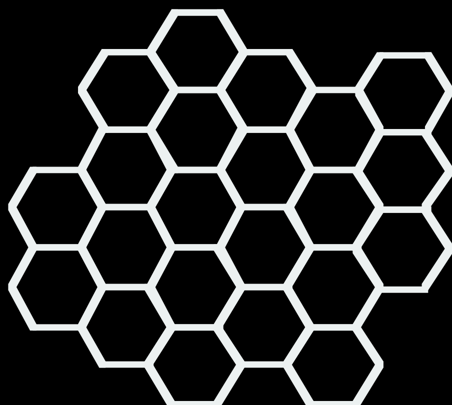
Maps Brightness Value -> RGB Color

- 0 -> (0, 0, 0)
- 1 -> (0, 0, 0)
- 2 -> (0, 0, 0)
- 3 -> (0 , 0, 0)
- ...
- 130-> (0,0,0)
- 131-> (.01, .01, .01)
- 132-> (.02,.02,.02)
- ...
- 200->(1,1,1)
- 201->(1,1,1)
- ...
- 255 -> (1, 1, 1)

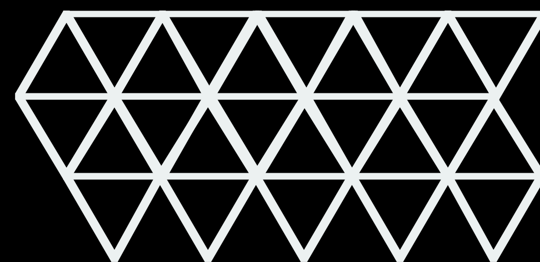
Brightness Equalization

An unequalized image

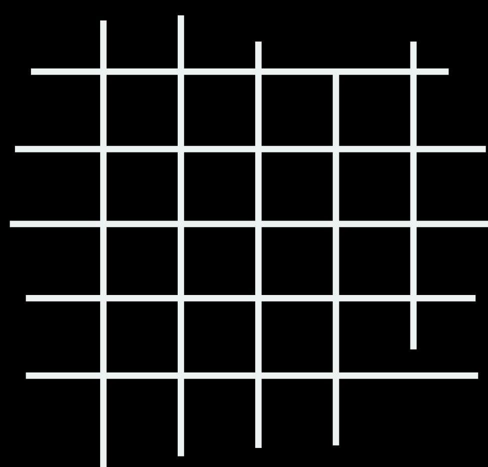
Tesselation Patterns



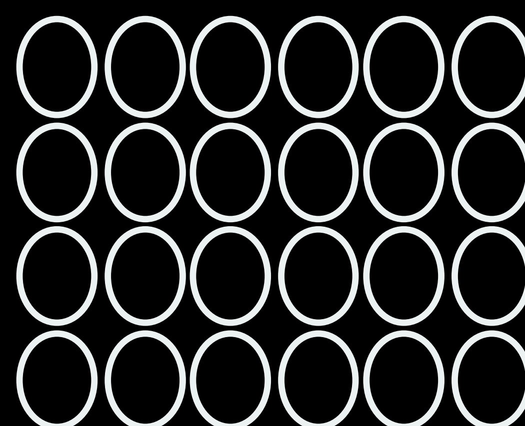
Hexagonal



Triangular



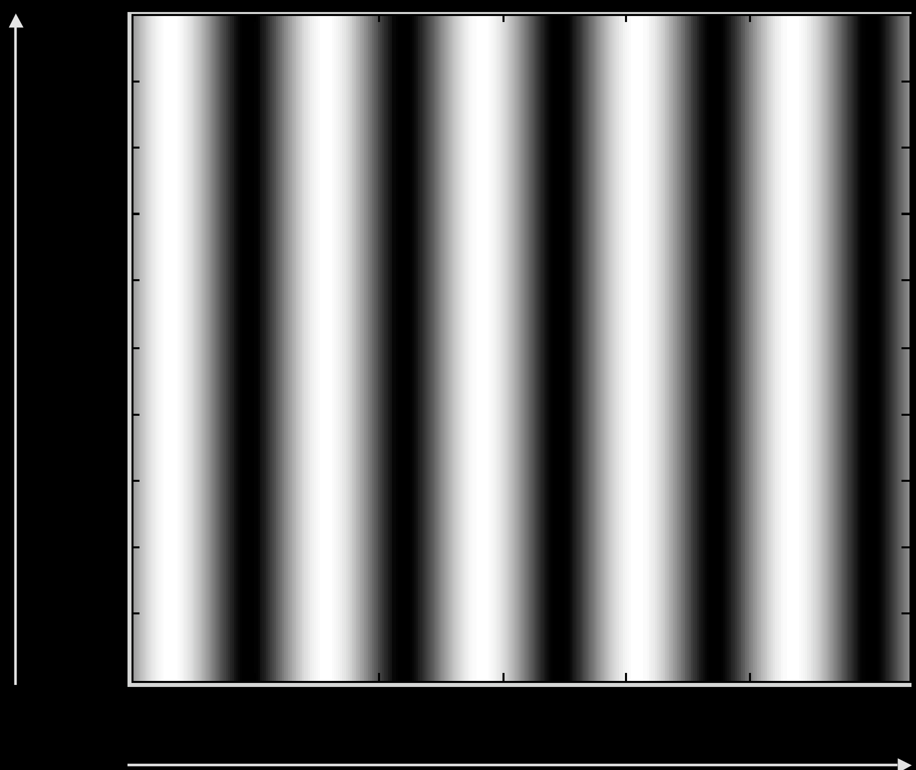
Rectangular



Typical

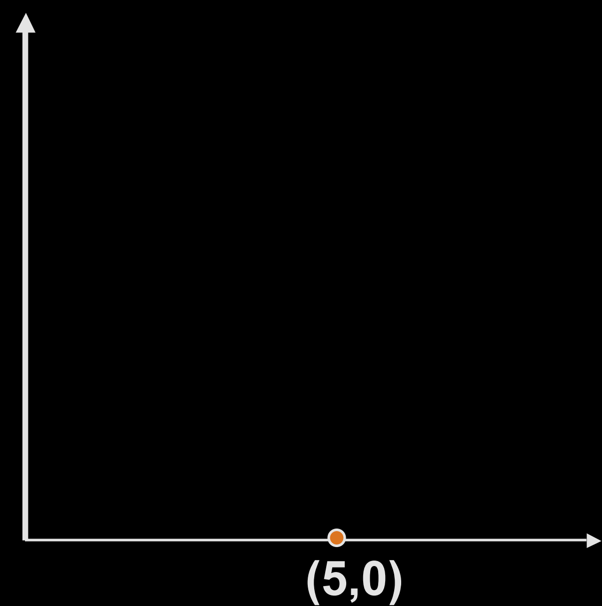
Spatial Frequencies

Image

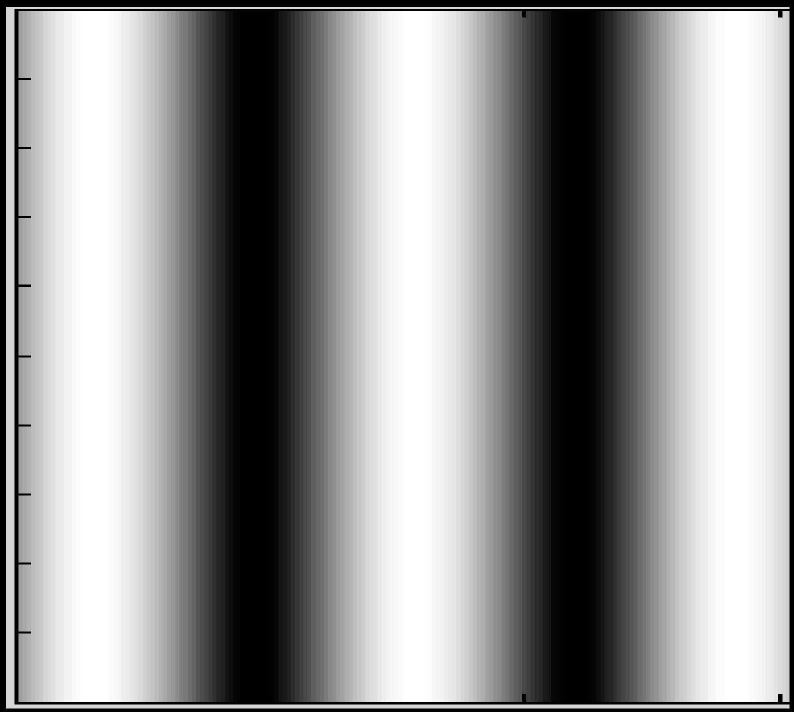


one “unit” of distance

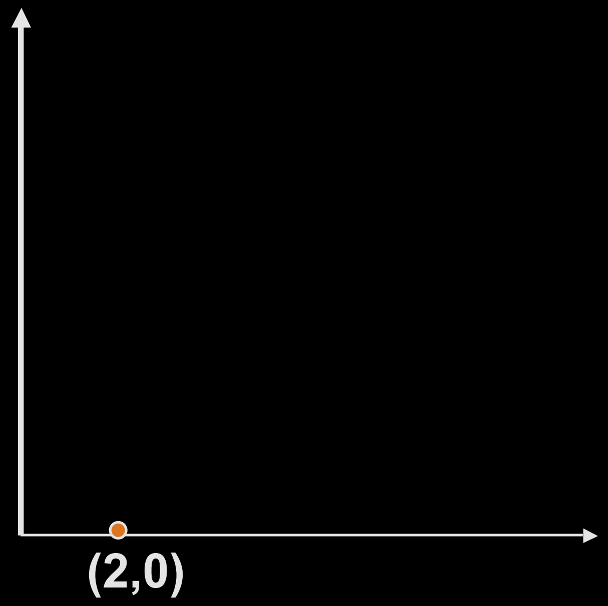
Fourier Power Spectrum



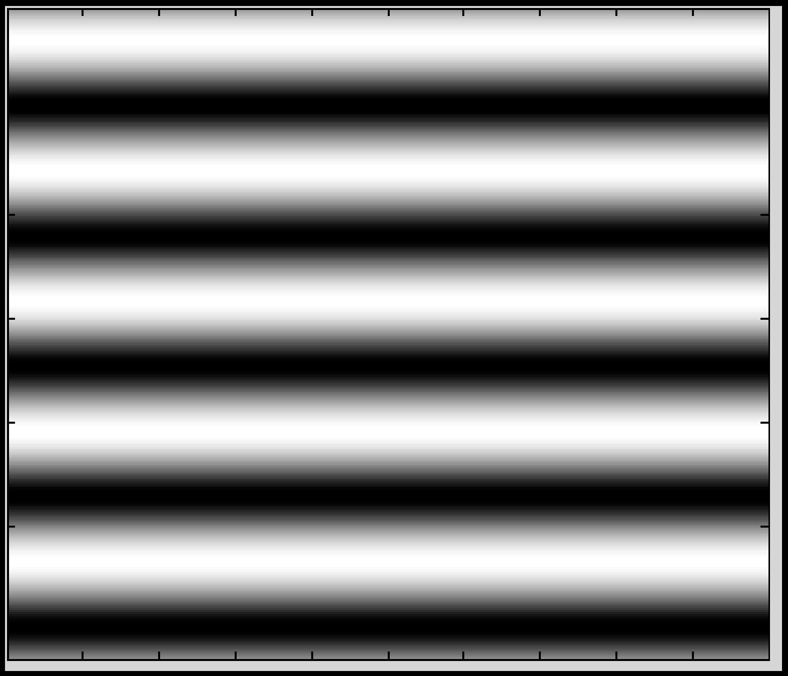
Spatial Frequencies



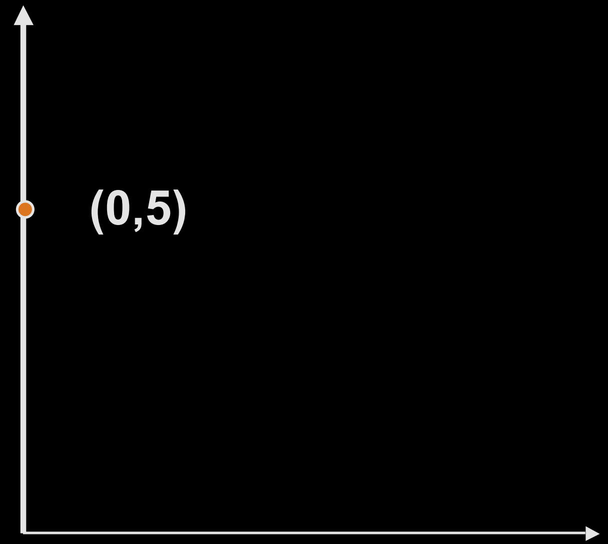
Fourier Power Spectrum



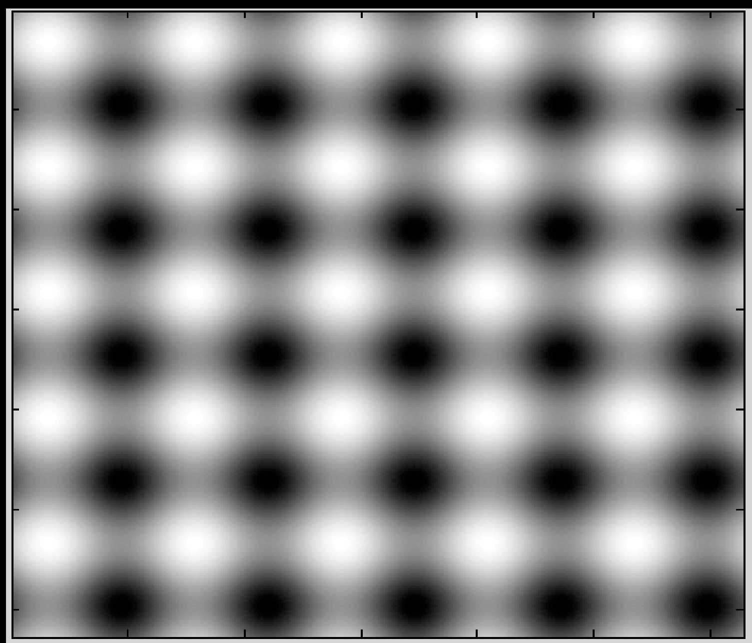
Spatial Frequencies



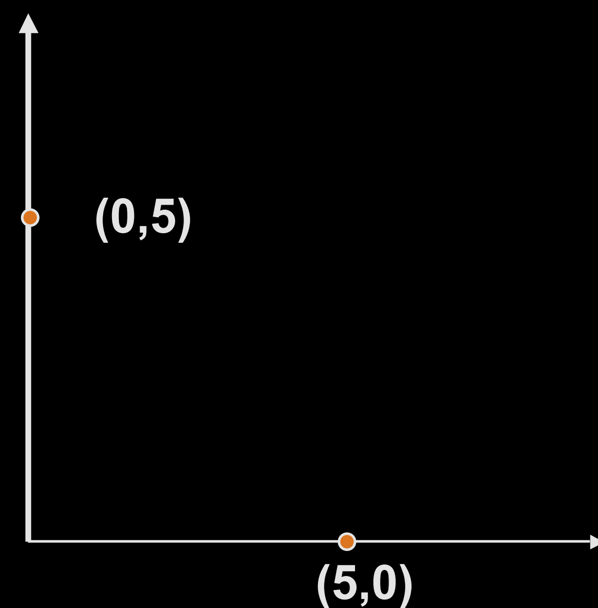
Fourier Power Spectrum



Spatial Frequencies

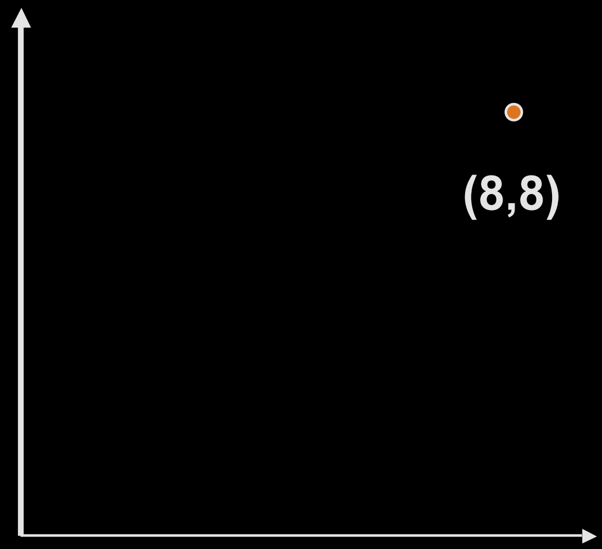


Fourier Power Spectrum



Spatial Frequencies

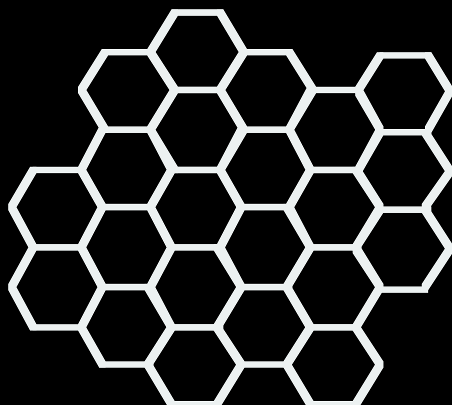
Fourier Power Spectrum



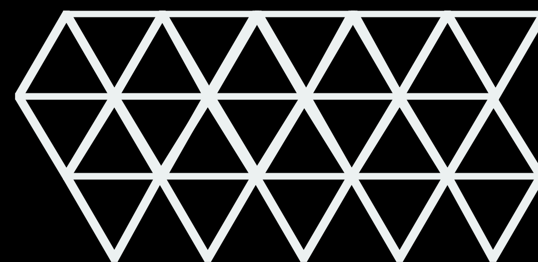
Sampling efficiency

- Every sampling scheme captures some spatial frequencies but not others:
  - Low frequency sampling doesn't capture the picket fence
  - High frequency does.
- Which two-dimensional sampling scheme is most "efficient"?

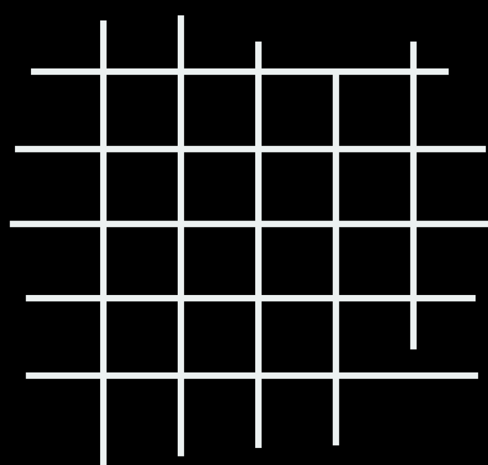
Tesselation Patterns



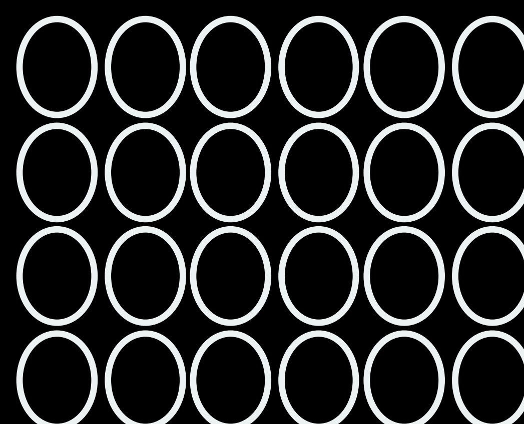
Hexagonal



Triangular

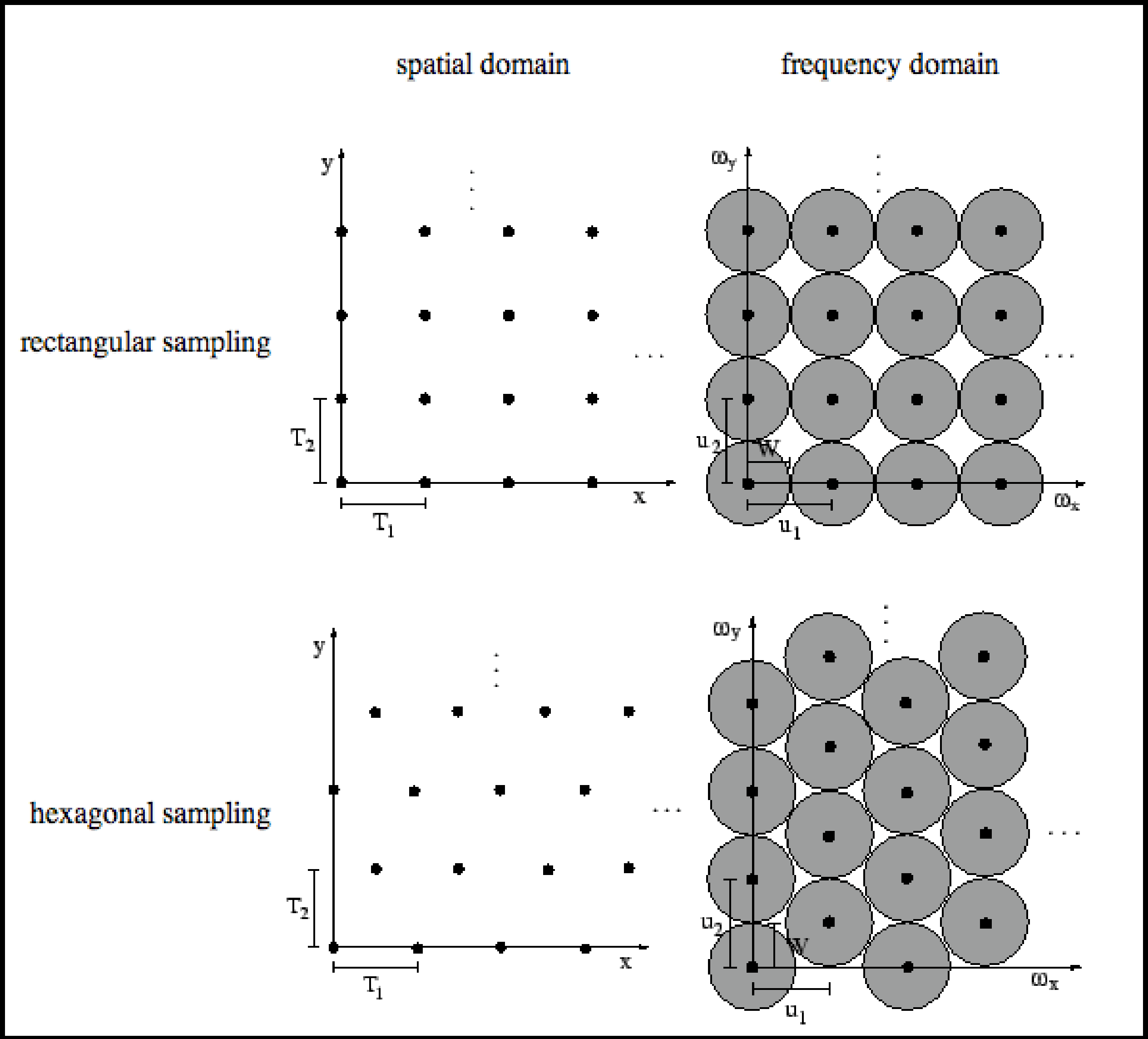


Rectangular



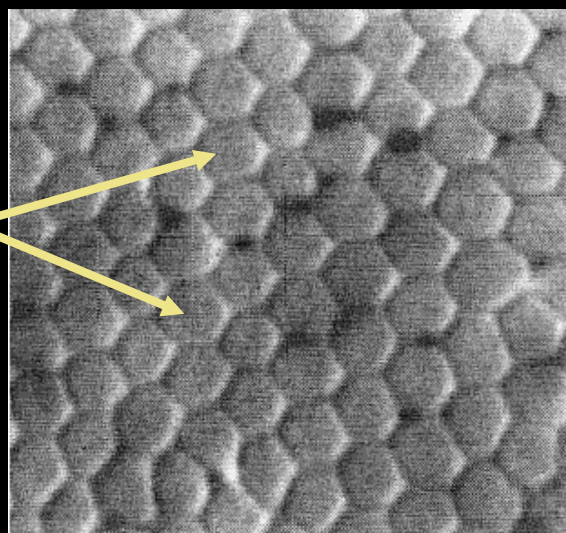
Typical

Sampling Grids



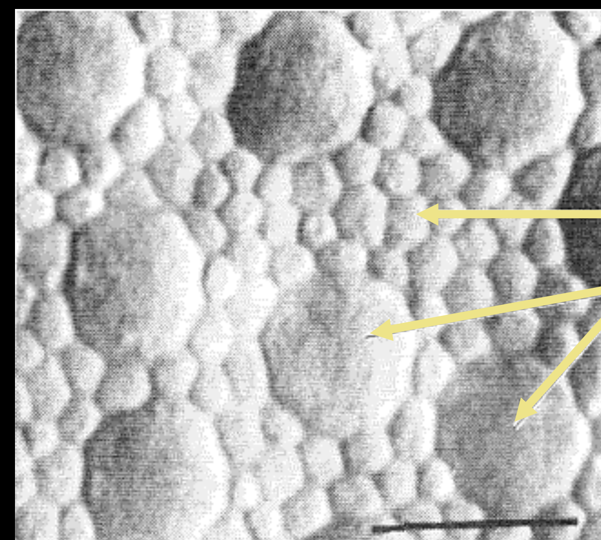
Retina

Cones in the fovea

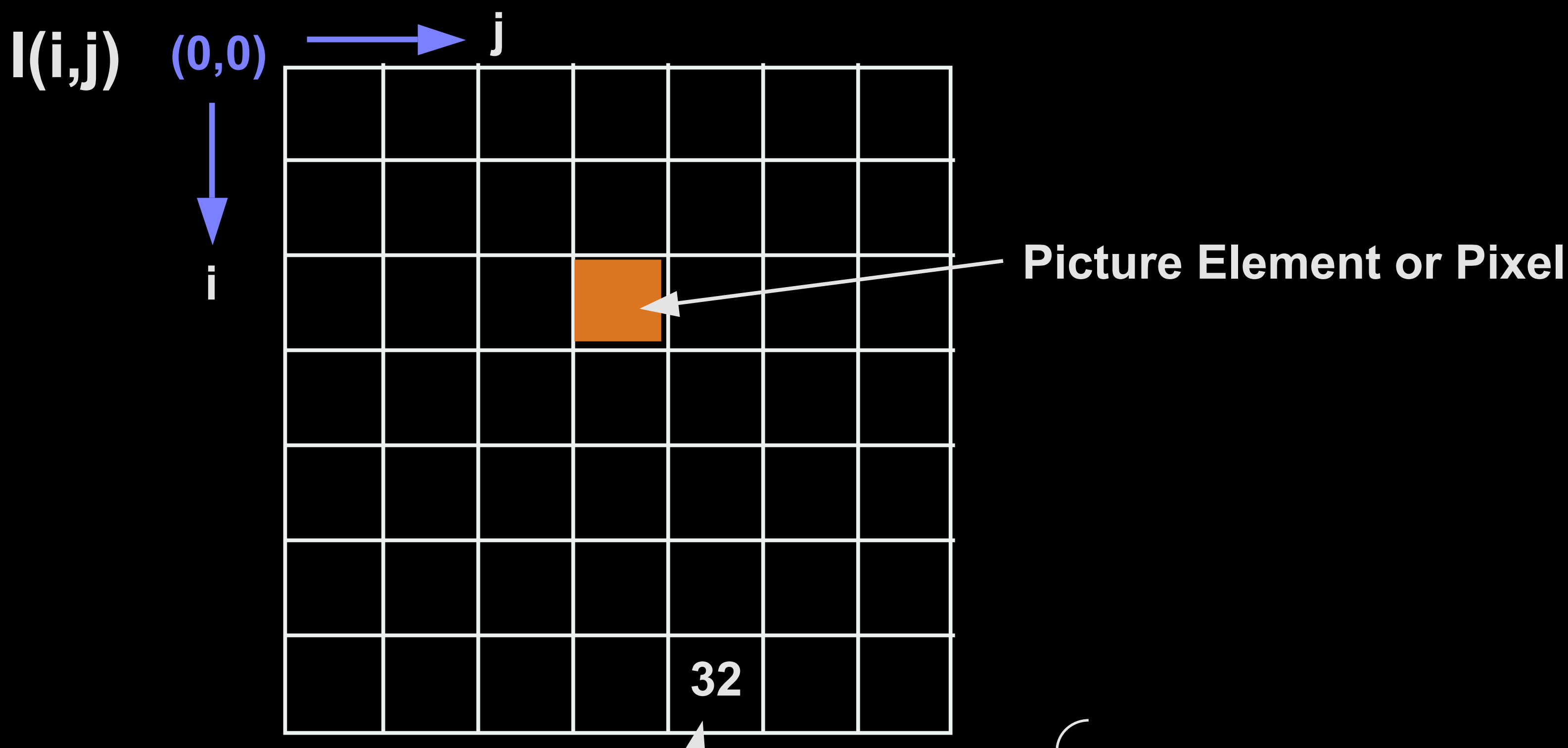


All of them are cones!

Moving outward from fovea



Digital Geometry

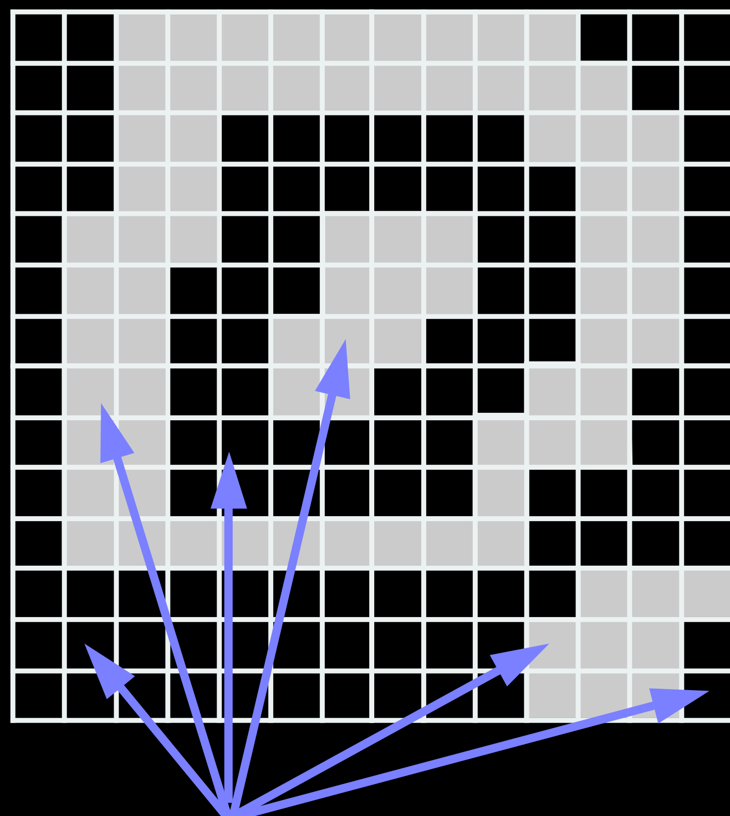


Pixel value I(I,j) =
- 0,1 Binary Image
- 0 - K-1 Gray Scale Image
- Vector: Multispectral Image

- Neighborhood
- Connectedness
- Distance Metrics

Connected Components

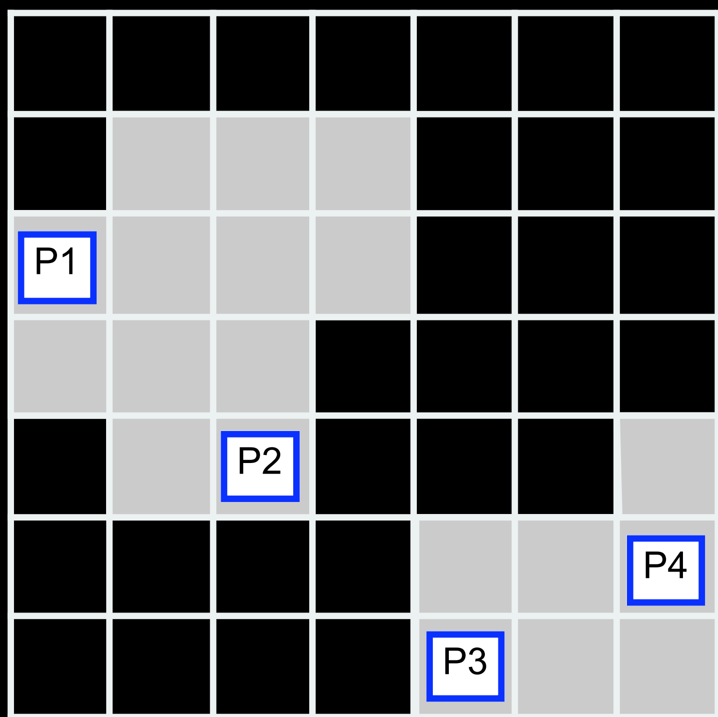
- Binary image with multiple 'objects'
- Separate 'objects' must be labeled individually



6 Connected Components

Finding Connected Components

- Two points in an image are 'connected' if a path can be found for which the value of the image function is the same all along the path.



P_1 connected to P_2

P_3 connected to P_4

P_1 not connected to P_3 or P_4

P_2 not connected to P_3 or P_4

P_3 not connected to P_1 or P_2

P_4 not connected to P_1 or P_2

Algorithm

- Pick any pixel in the image and assign it a label
- Assign same label to any neighbor pixel with the same value of the image function
- Continue labeling neighbors until no neighbors can be assigned this label
- Choose another label and another pixel not already labeled and continue
- If no more unlabeled image points, stop.

Who's my neighbor?

Example

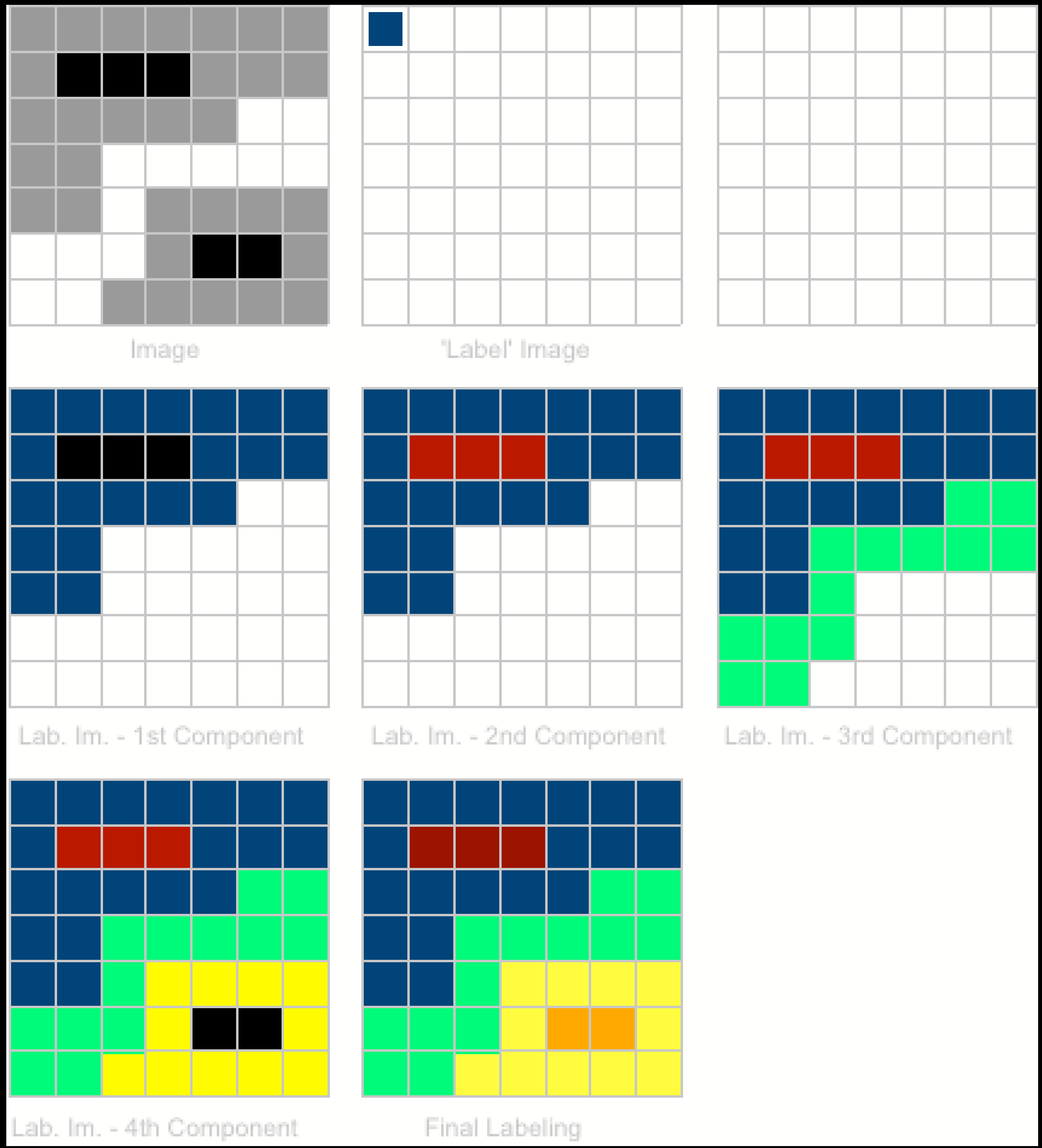
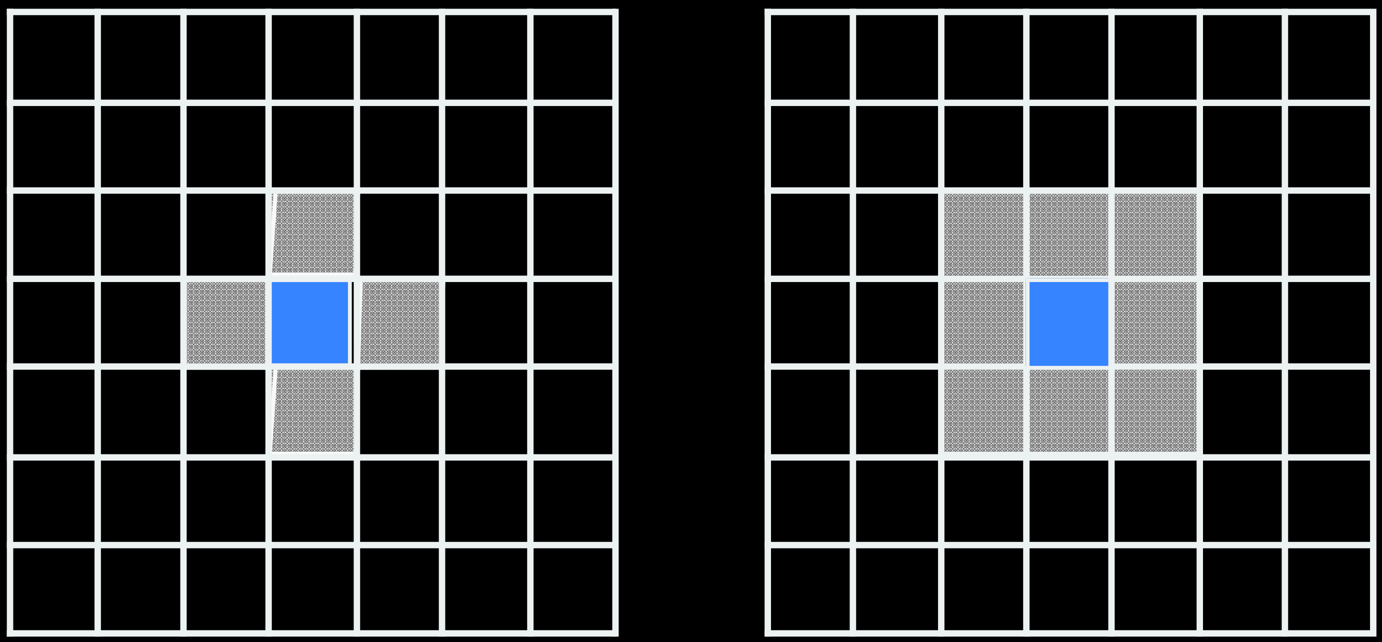


Image
'Label' Image
Lab. Im. - 1st Component
Lab. Im. - 2nd Component
Lab. Im. - 3rd Component
Lab. Im. - 4th Component
Final Labeling

Neighbor

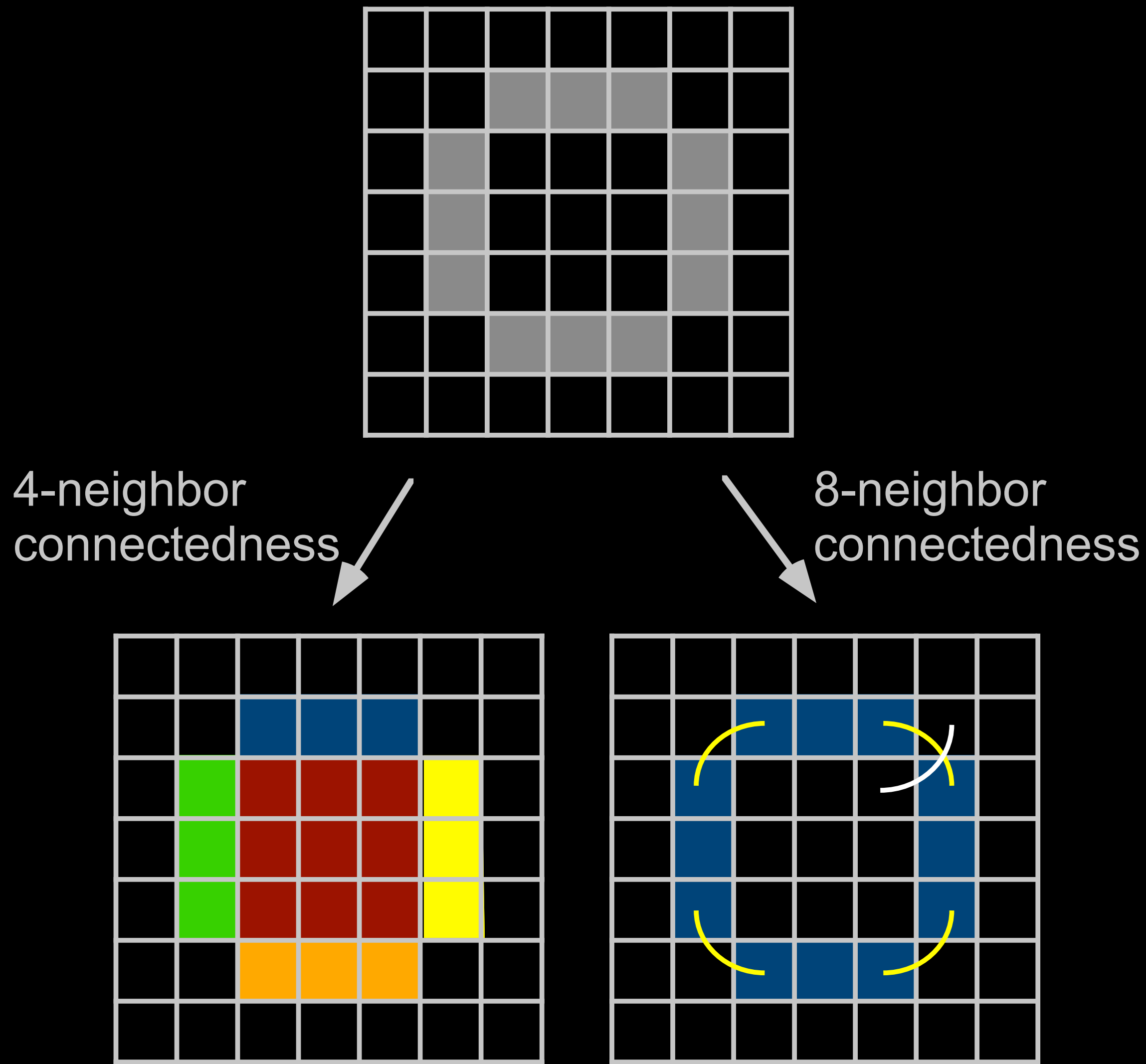
- Consider the definition of the term 'neighbor'
- Two common definitions:



Four Neighbor Eight Neighbor

- Consider what happens with a closed curve.
- One would expect a closed curve to partition the plane into two connected regions.

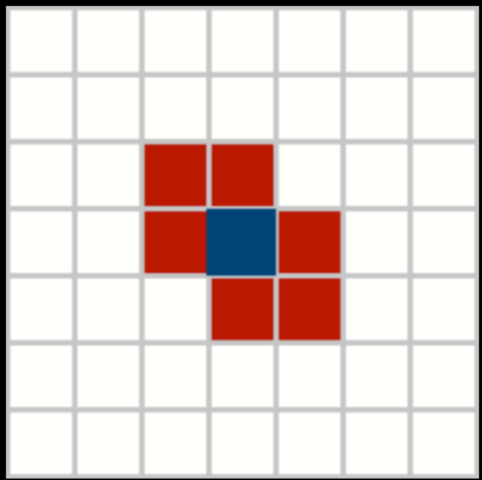
Alternate Neighborhood Definitions



Neither neighborhood definition satisfactory!

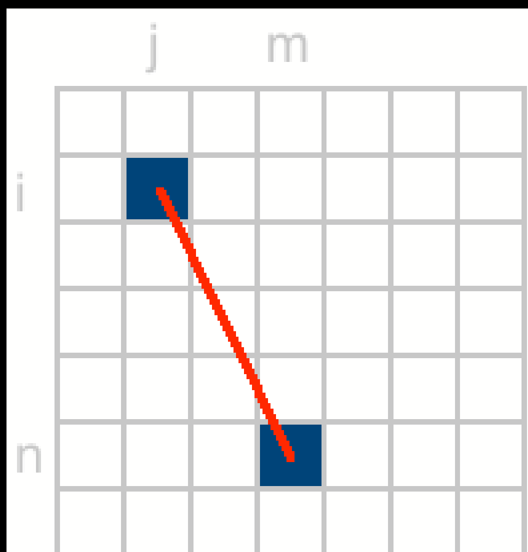
Possible Solutions

- Use 4-neighborhood for object and 8-neighborhood for background
  - requires a-priori knowledge about which pixels are object and which are background
- Use a six-connected neighborhood:



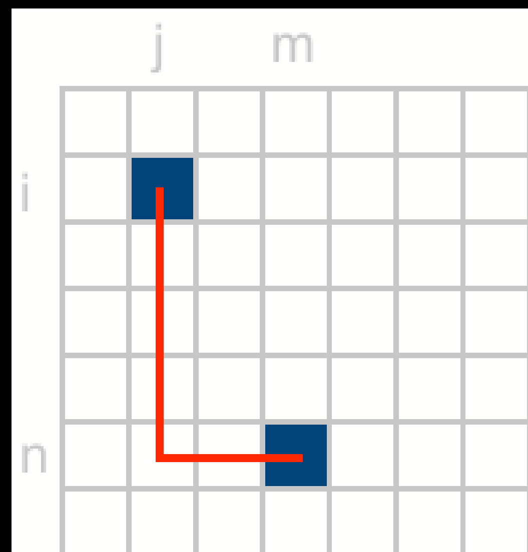
Digital Distances

- Alternate distance metrics for digital images



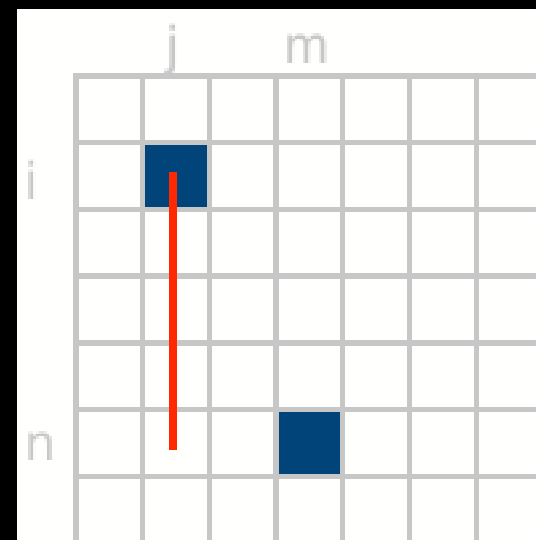
Euclidean Distance

$$= \sqrt{(i-n)^2 + (j-m)^2}$$



City Block Distance

$$= |i-n| + |j-m|$$



Chessboard Distance

$$= \max[\ |i-n|, |j-m|\ ]$$