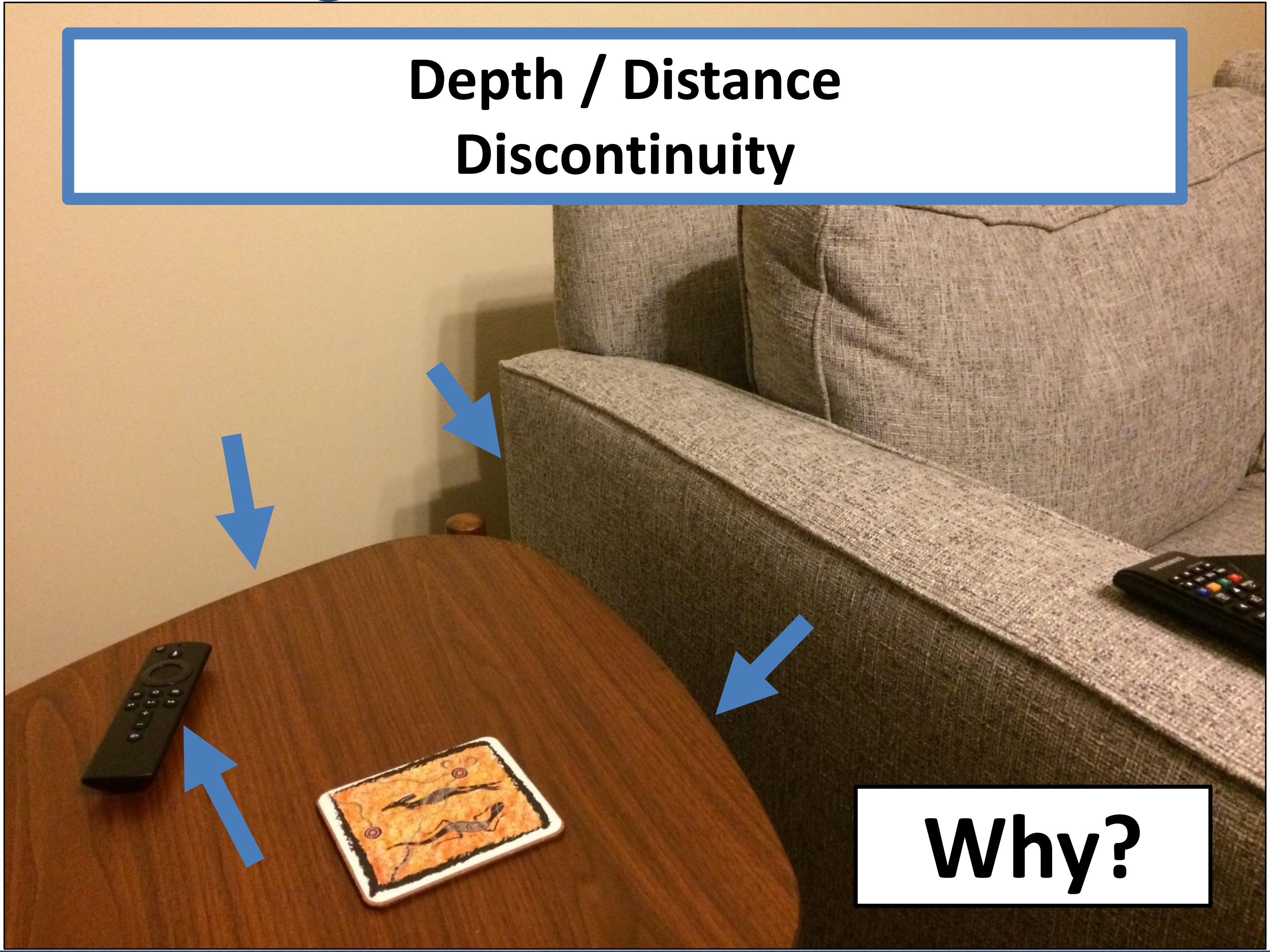
計算認知神經科學
計算視覺

Edge

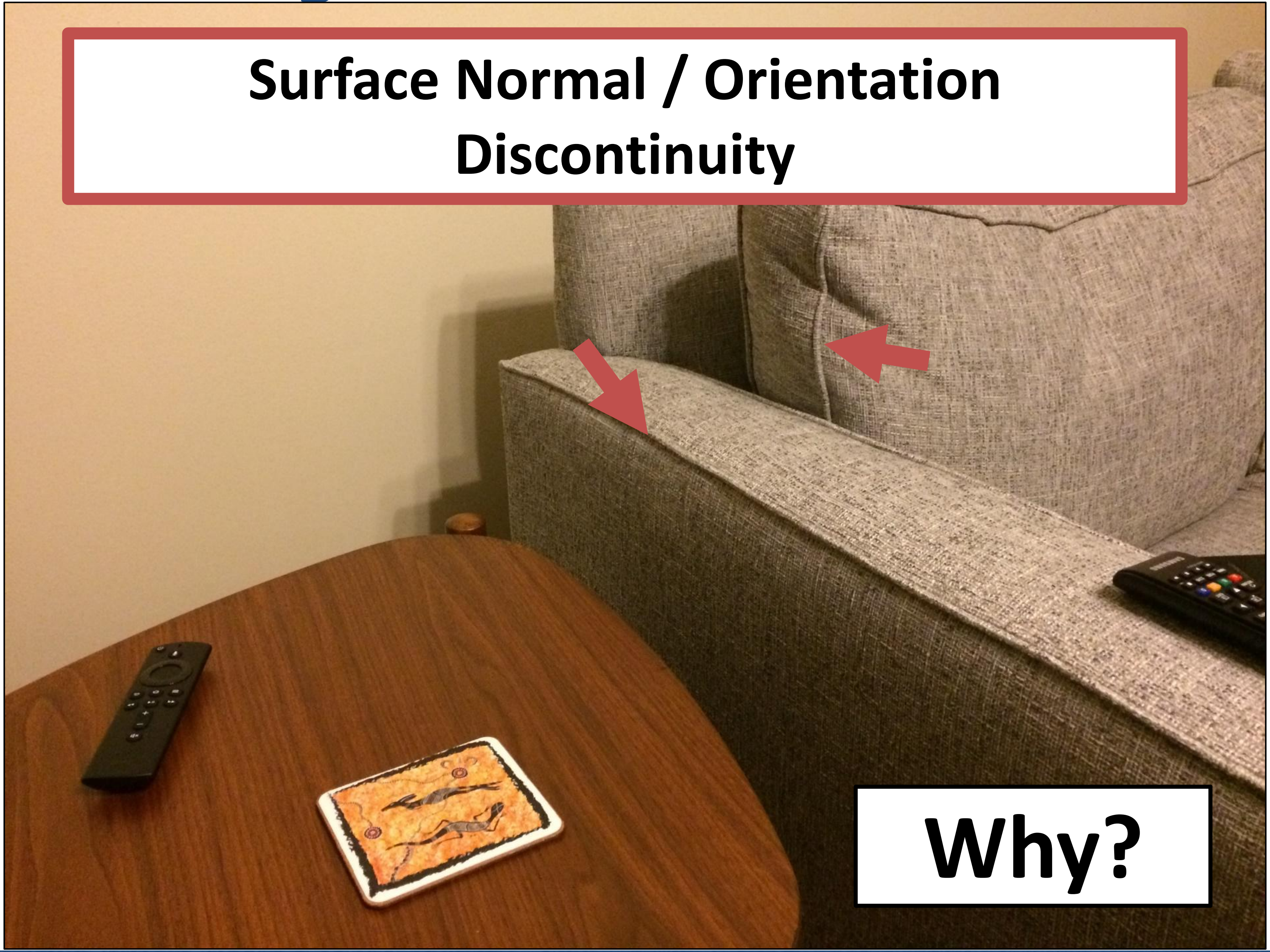
Color often not a powerful feature

However, these are all images of people but the colors in each image are very different.

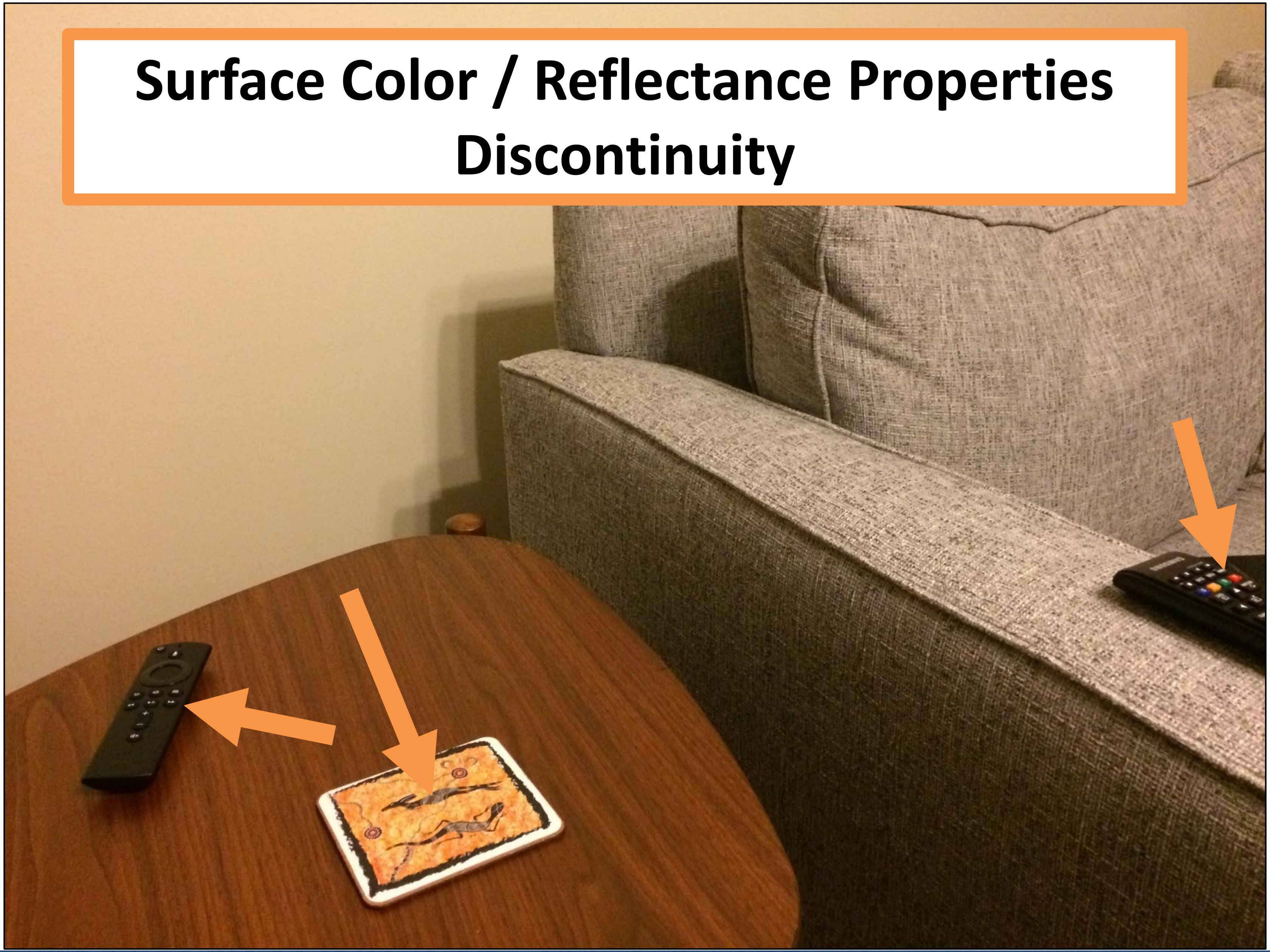
Where do Edges Come From?



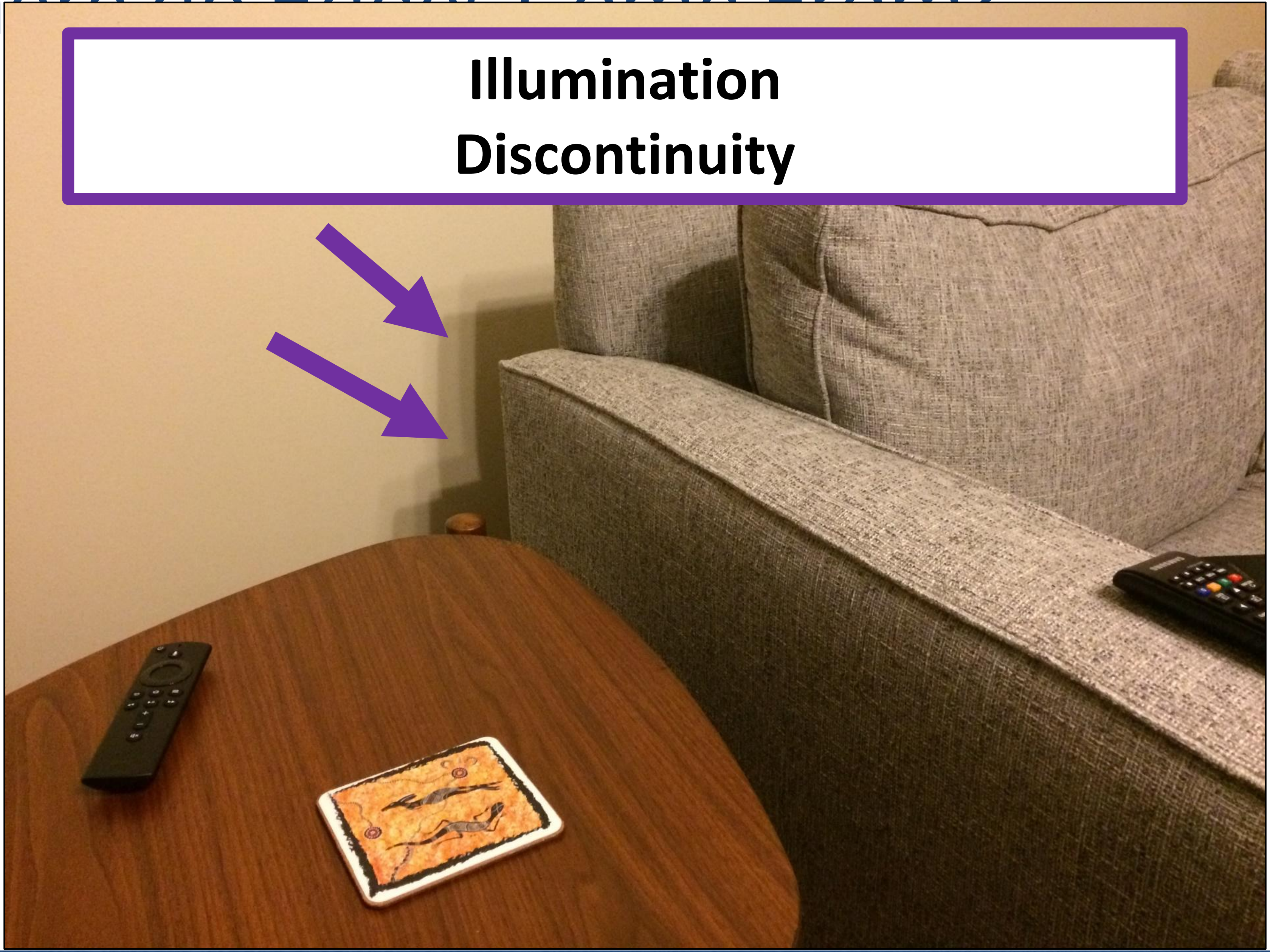
Where do Edges Come From?



Where do Edges Come From?

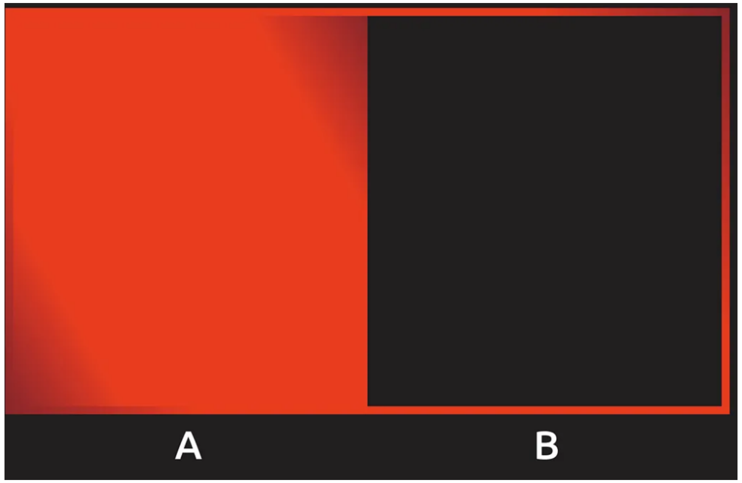


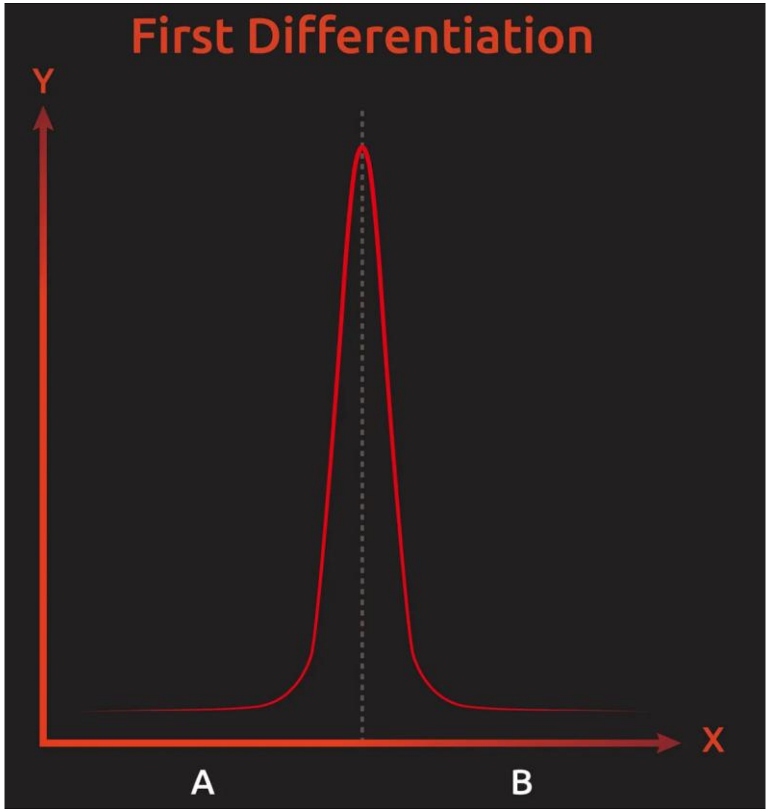
Where do Edges Come From?

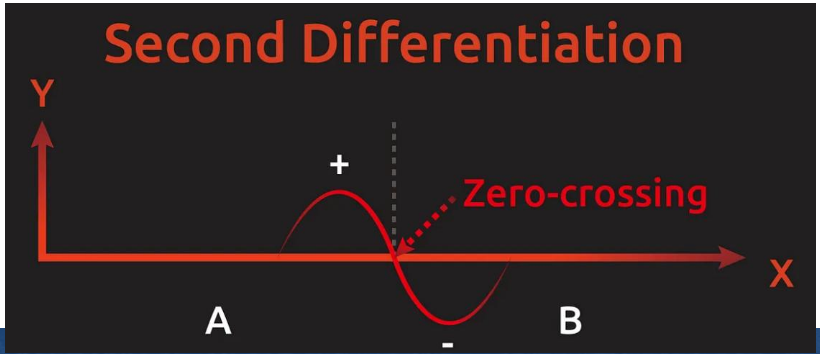


邊緣(edge)

在 A 、B 兩個區域內的像素灰階值是非常相近的，而兩個區域之間的灰階值會出現「 **急劇變化 (Abrupt change)** 」，可以把該區域之灰階值做一次微分，並繪製成二維示意圖來觀察

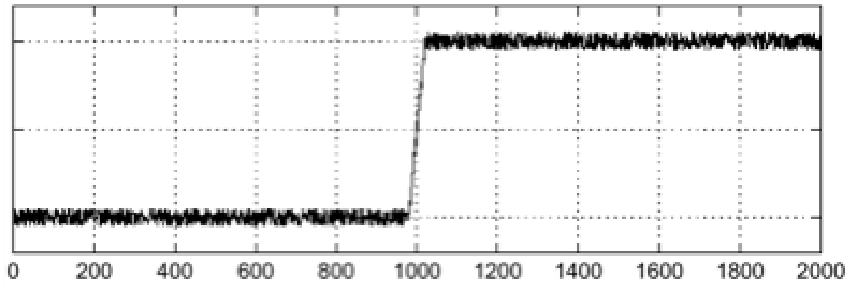






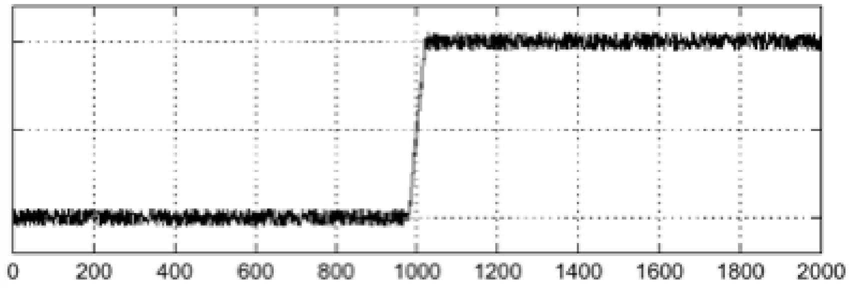
How do you find the edge of this signal?

intensity plot



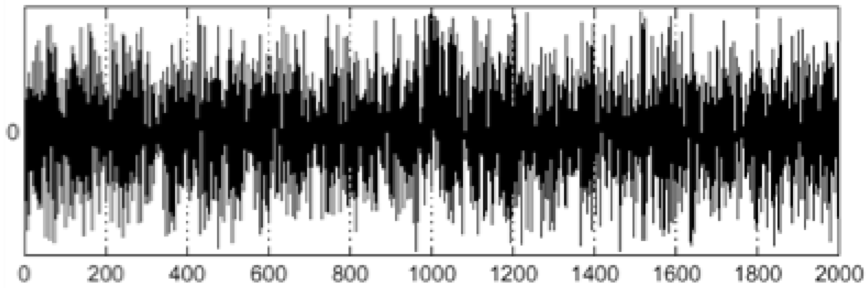
How do you find the edge of this signal?

intensity plot



Using a derivative filter:

derivative plot



What's the problem here?

Differentiation is very sensitive to noise

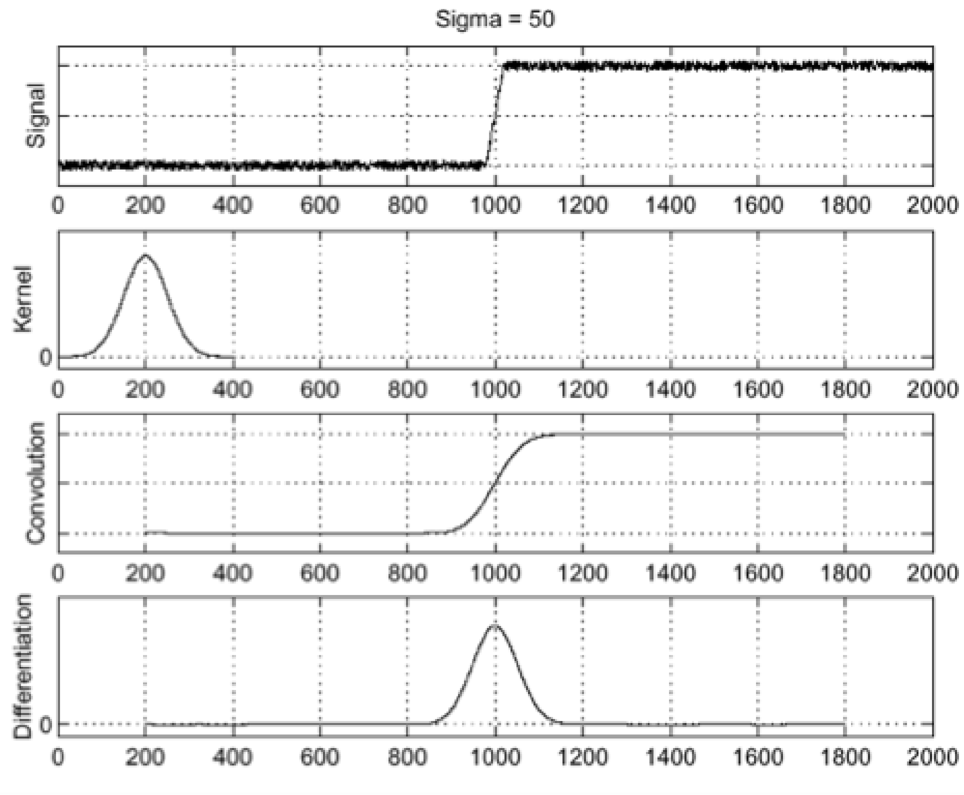
When using derivative filters, it is critical to blur first!

input

Gaussian

blurred

derivative of blurred



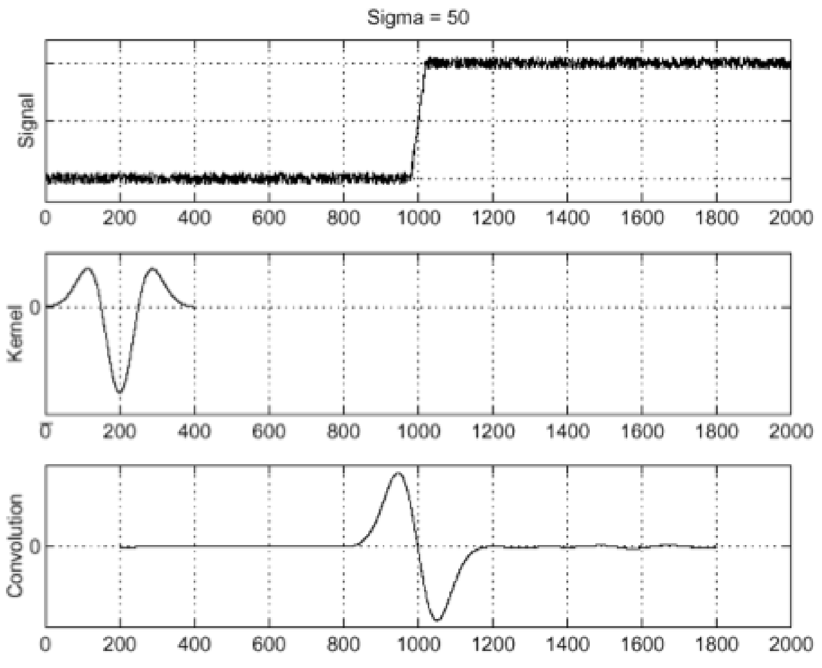
Laplacian of Gaussian (LoG) filter

As with derivative, we can combine Laplace filtering with Gaussian filtering

input

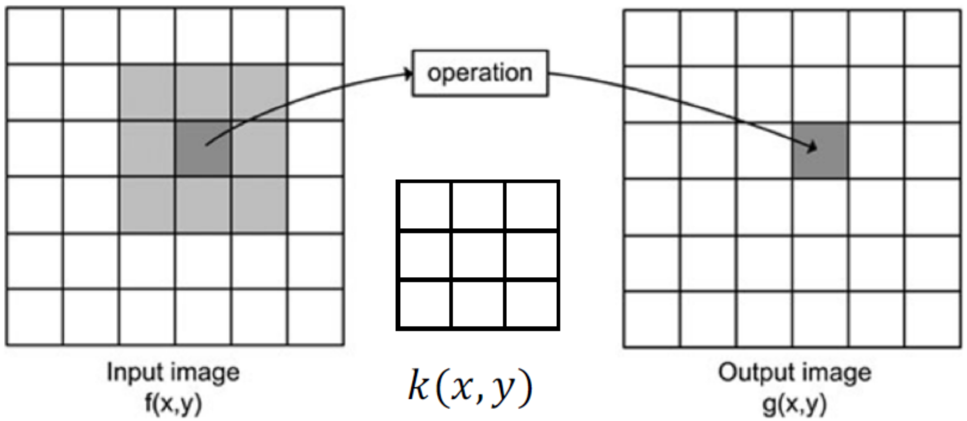
Laplacian of Gaussian

output



“zero crossings” at edges

Image filtering: Convolution operator
Important Filter: Sobel operator



$k(x,y)$ =

1	0	-1
2	0	-2
1	0	-1

Image Credit: http://what-when-how.com/introduction-to-video-and-image-processing/neighborhood-processing-introduction-to-video-and-image-processing-part-1/

The Sobel filter

$$\begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline 2 & 0 & -2 \\ \hline 1 & 0 & -1 \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 1 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline \end{array}$$

Sobel filter

What filter is this?

1D derivative filter

The Sobel filter

$$
\begin{array}{|c|c|c|}
\hline
1 & 0 & -1 \\
\hline
2 & 0 & -2 \\
\hline
1 & 0 & -1 \\
\hline
\end{array}
\quad = \quad
\begin{array}{|c|}
\hline
1 \\
\hline
2 \\
\hline
1 \\
\hline
\end{array}
\quad * \quad
\begin{array}{|c|c|c|}
\hline
1 & 0 & -1 \\
\hline
\end{array}
$$

Sobel filter

Blurring

1D derivative filter

The Sobel filter

Horizontal Sober filter:

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$$

What does the vertical Sobel filter look like?

The Sobel filter

Horizontal Sober filter:

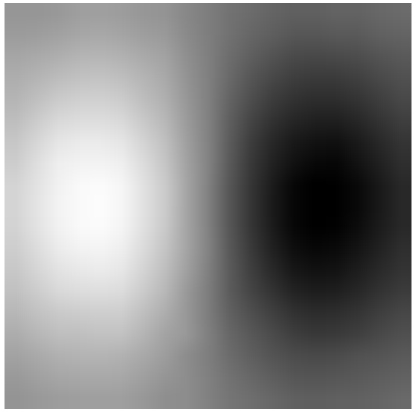
$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$$

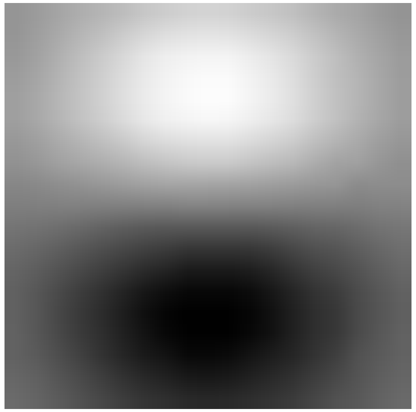
Vertical Sobel filter:

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Filters We've Seen

Gaussian Derivative



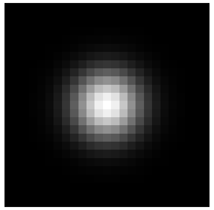
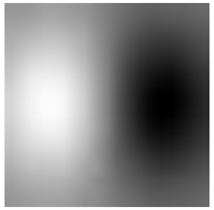
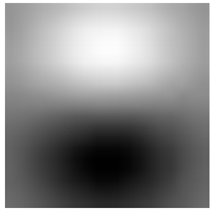


Sobel Filter

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Why would anybody use the bottom filter?

Filters We've Seen

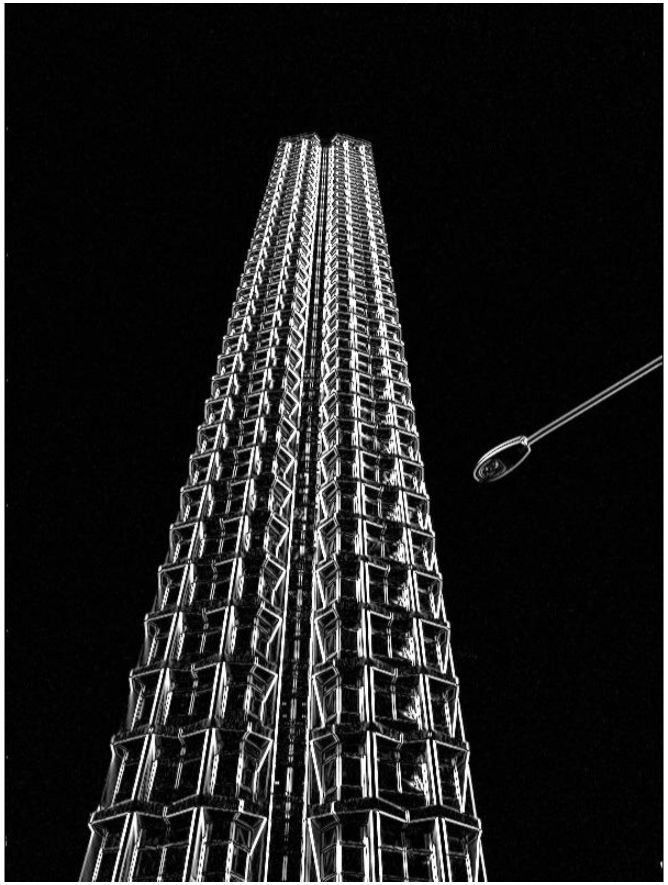
	Smoothing	Derivative
		 
Example	Gaussian	Deriv. of gauss
Goal	Remove noise	Find edges
Only +?	Yes	No
Sums to	1	0

Why sum to 1 or 0, intuitively?

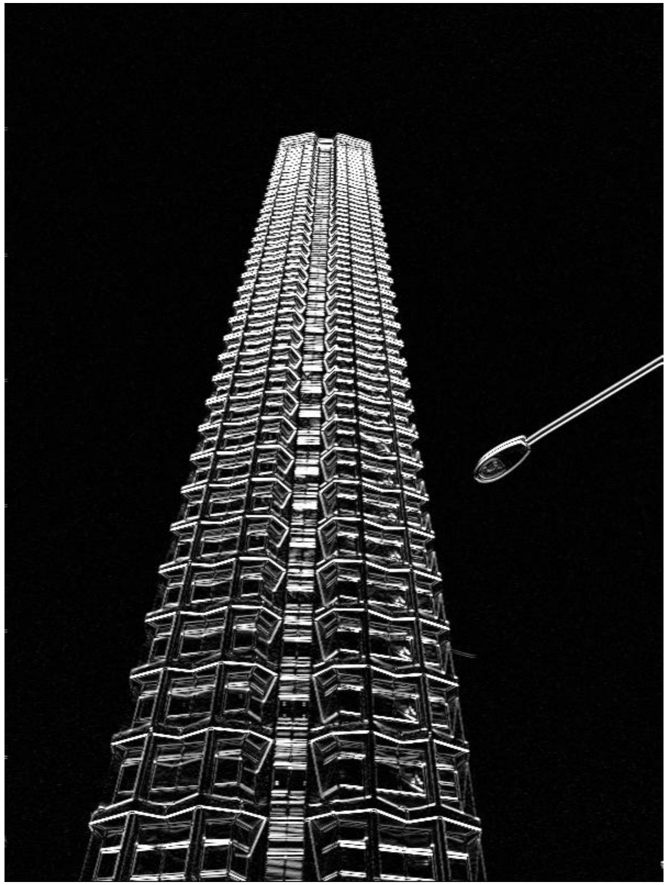
Slide Credit: J. Deng

Sobel filter example

original



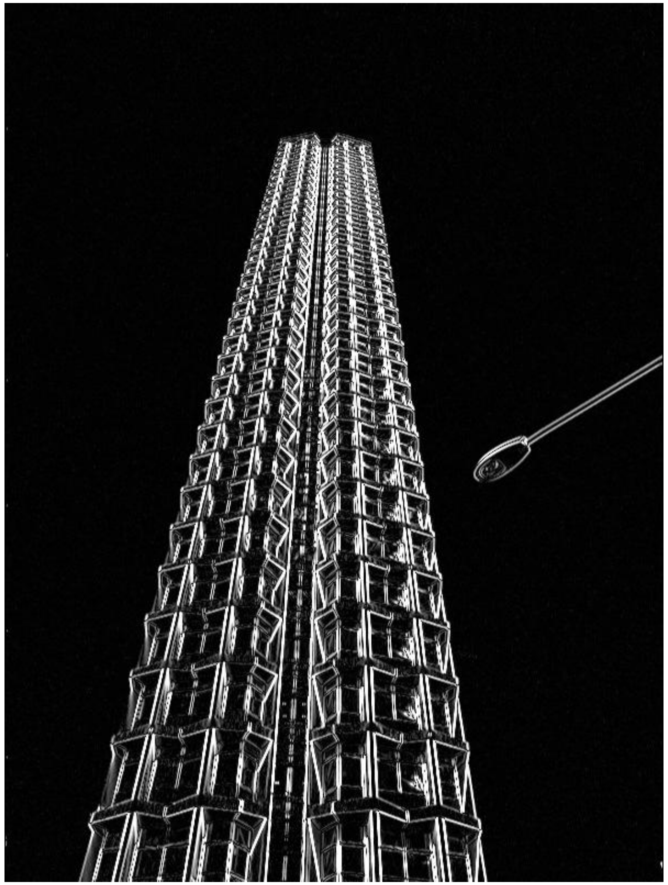
which Sobel filter?



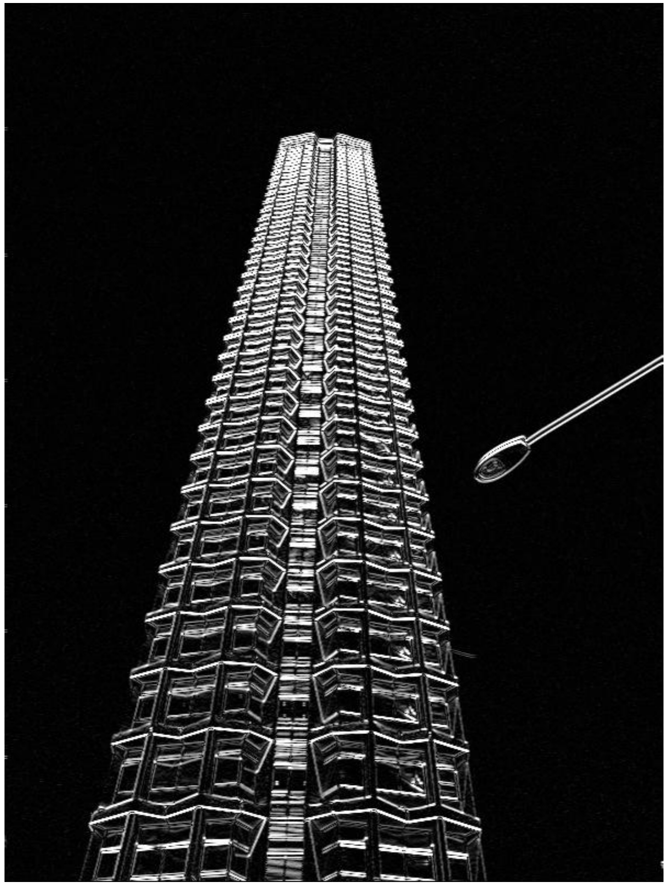
which Sobel filter?

Sobel filter example

original

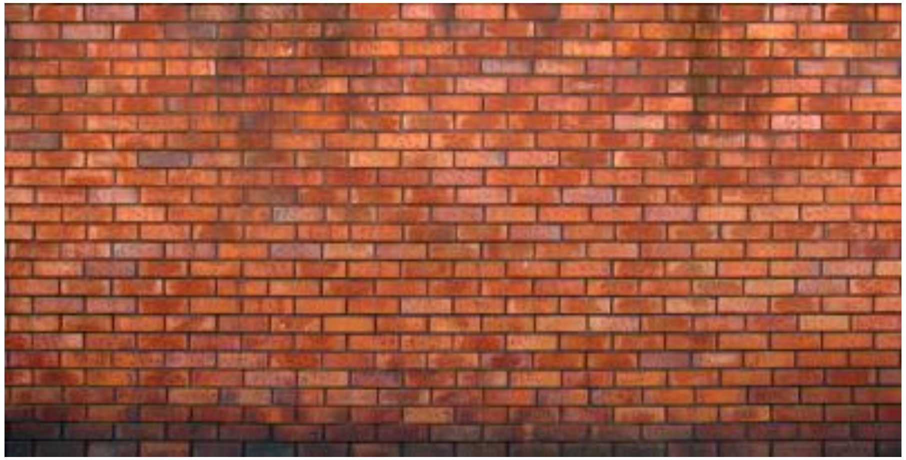


horizontal Sobel filter



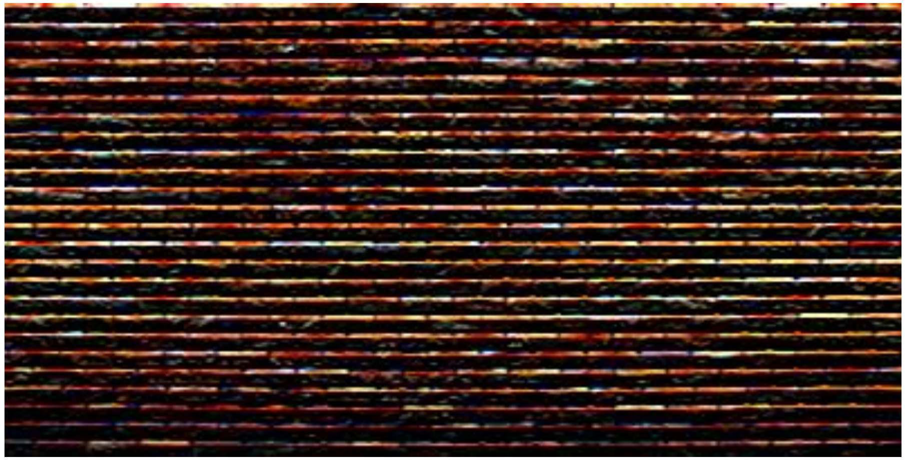
vertical Sobel filter

Sobel filter example



original

horizontal Sobel filter



vertical Sobel filter

Several derivative filters

Sobel

1	0	-1
2	0	-2
1	0	-1

1	2	1
0	0	0
-1	-2	-1

Scharr

3	0	-3
10	0	-10
3	0	-3

3	10	3
0	0	0
-3	-10	-3

Prewitt

1	0	-1
1	0	-1
1	0	-1

1	1	1
0	0	0
-1	-1	-1

Roberts

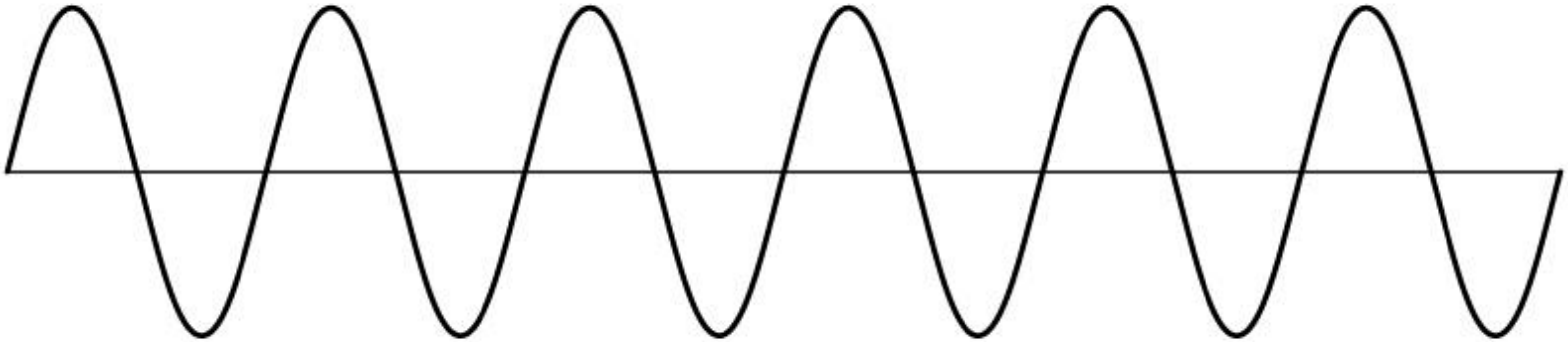
0	1
-1	0

1	0
0	-1

- How are the other filters derived and how do they relate to the Sobel filter?
- How would you derive a derivative filter that is larger than 3x3?

Sampling

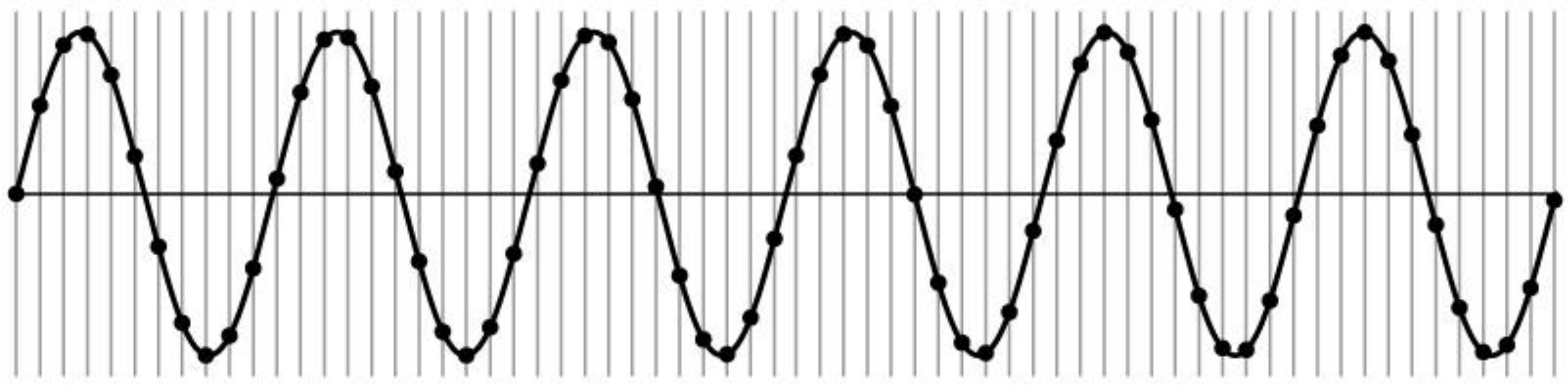
Very simple example: a sine wave



How would you discretize this signal?

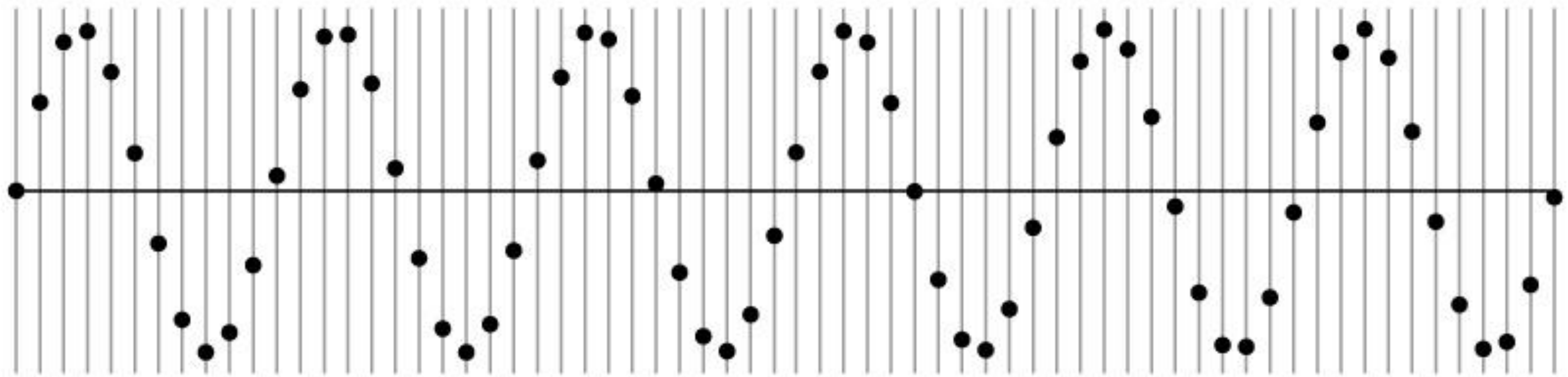
Sampling

Very simple example: a sine wave



Sampling

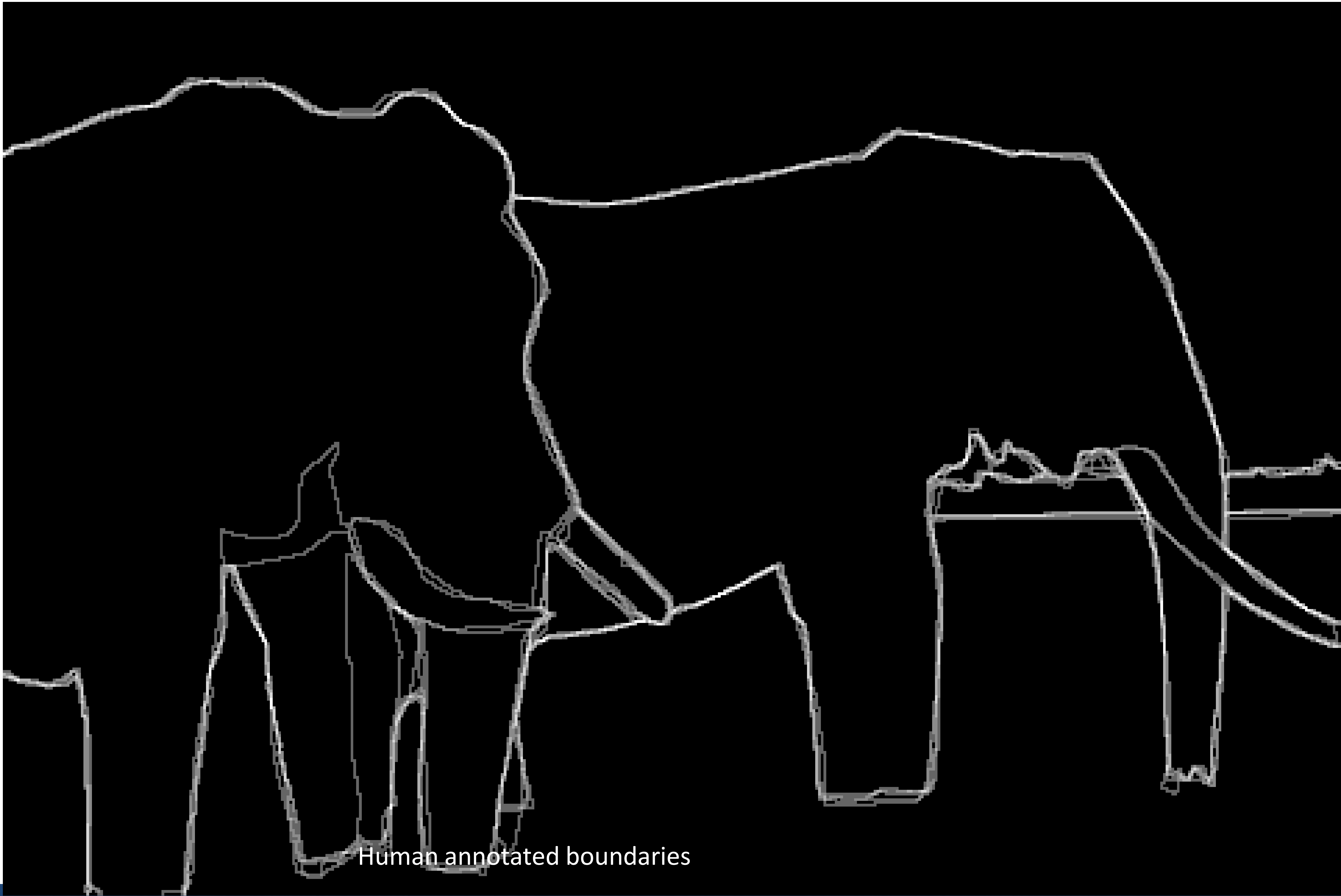
Very simple example: a sine wave

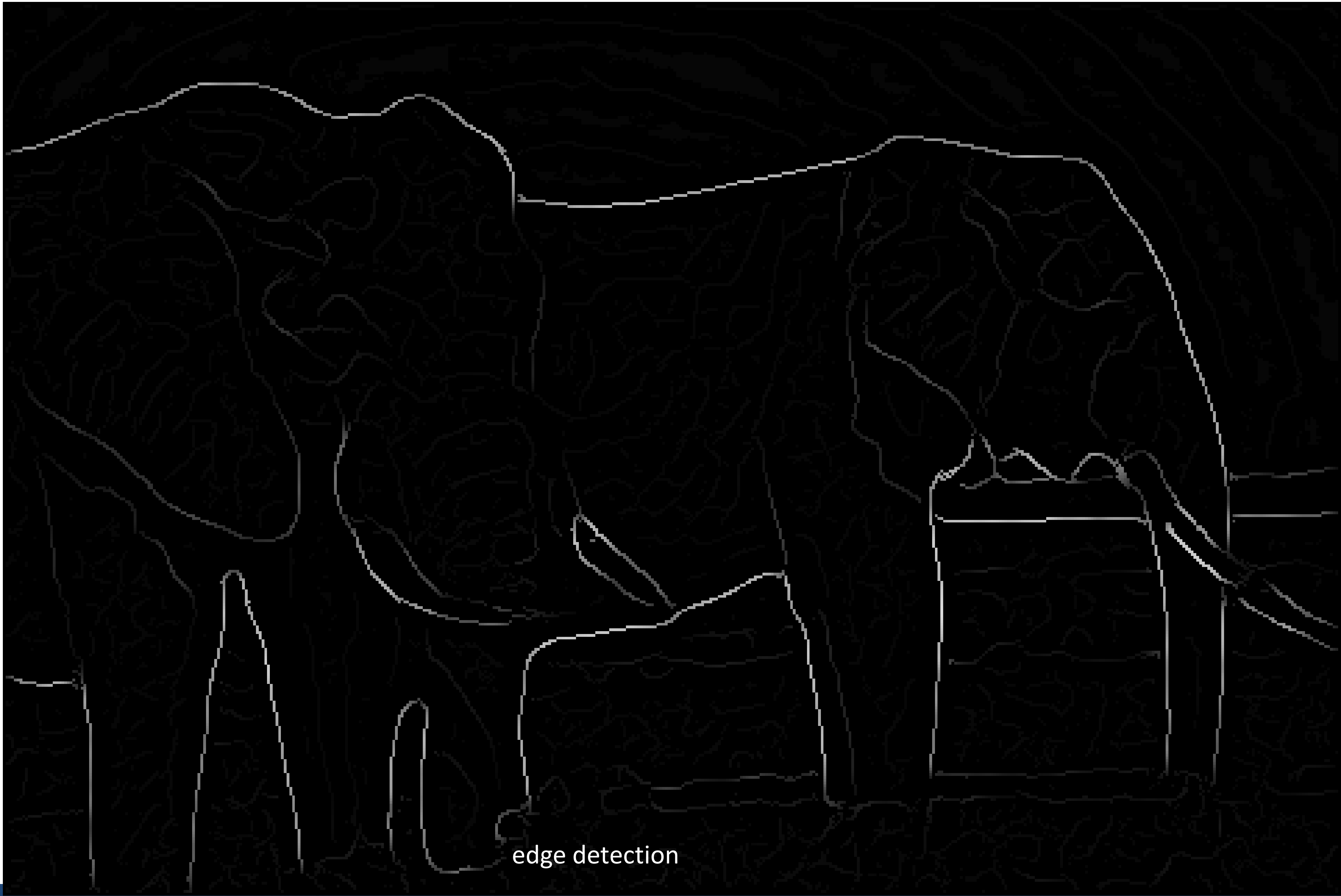


How many samples should I take?

Can I take as *many* samples as I want?


Where are the object boundaries?


Human annotated boundaries


edge detection

Where are the object boundaries?

edge detection

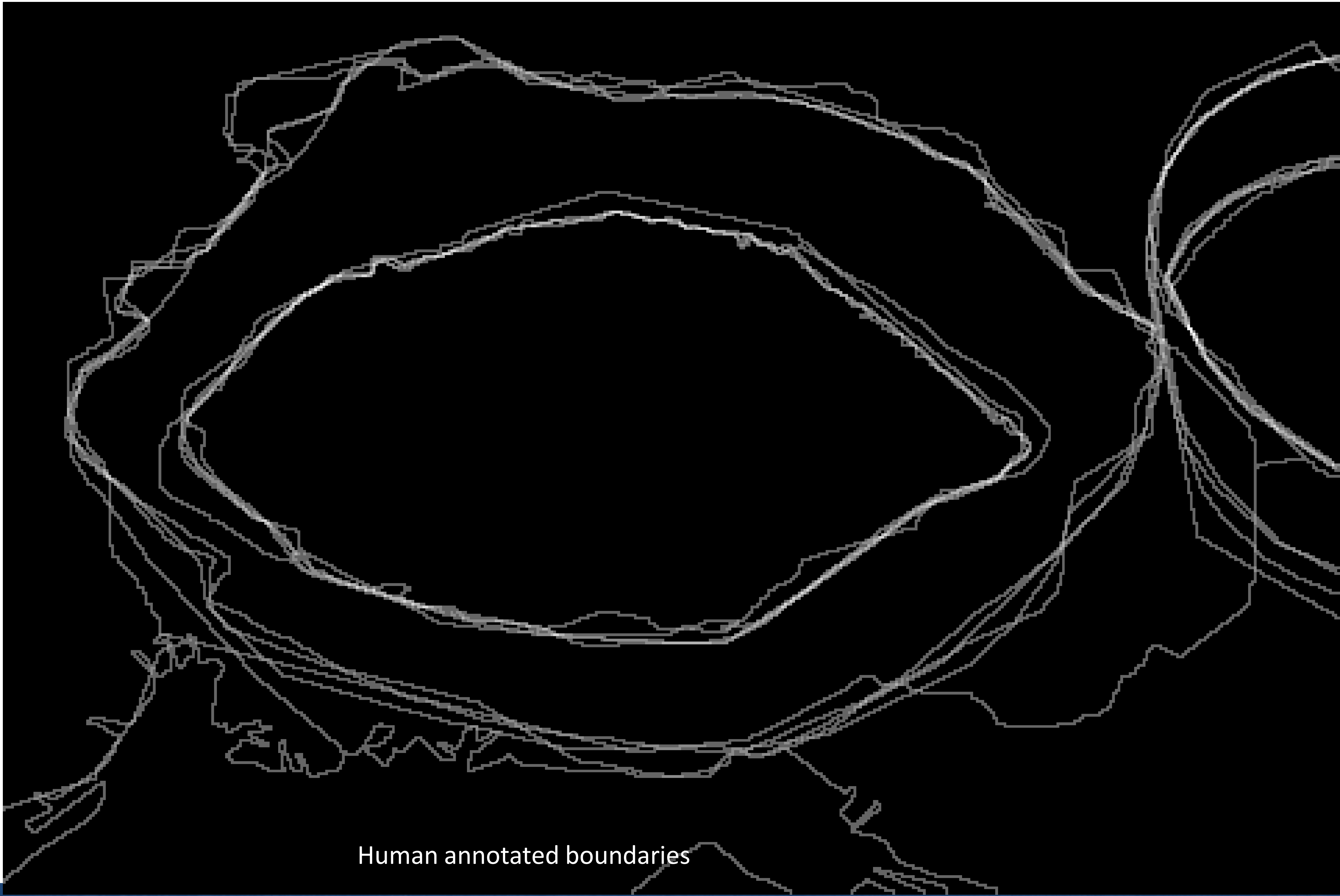
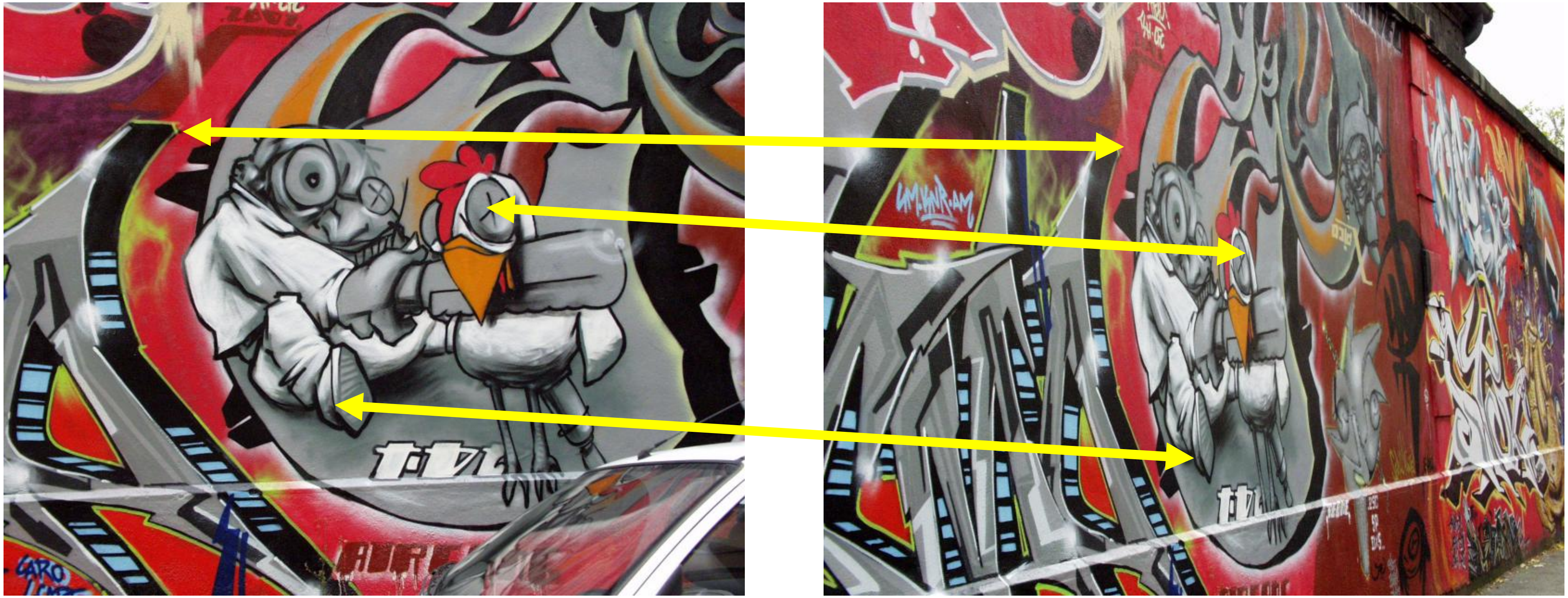
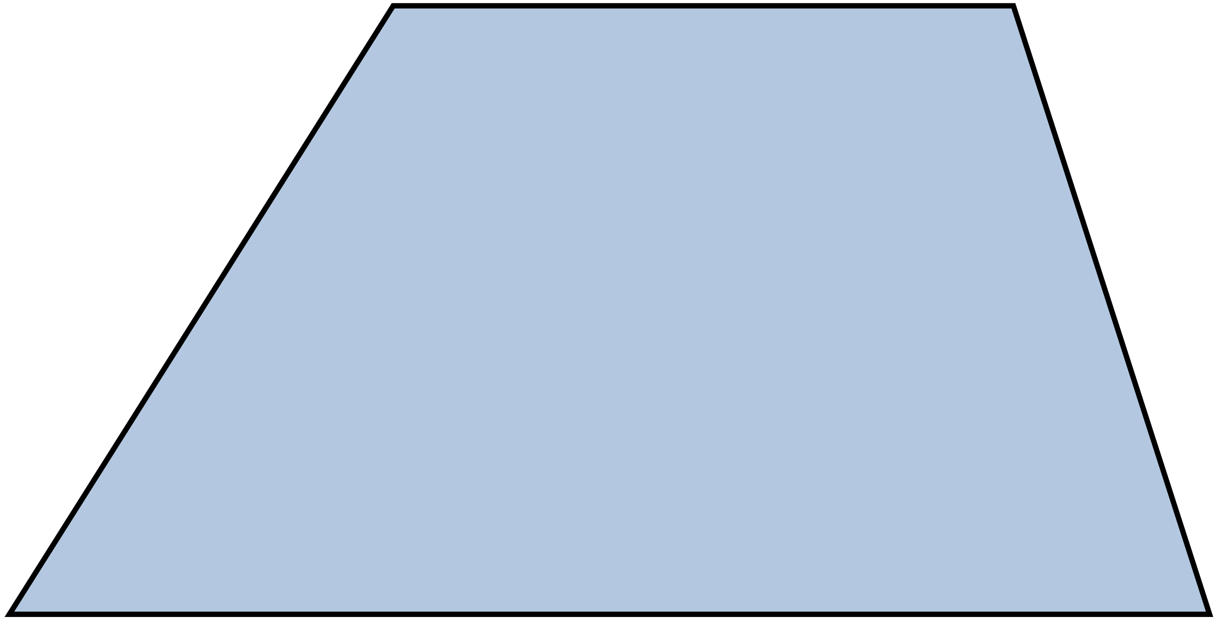

Human annotated boundaries

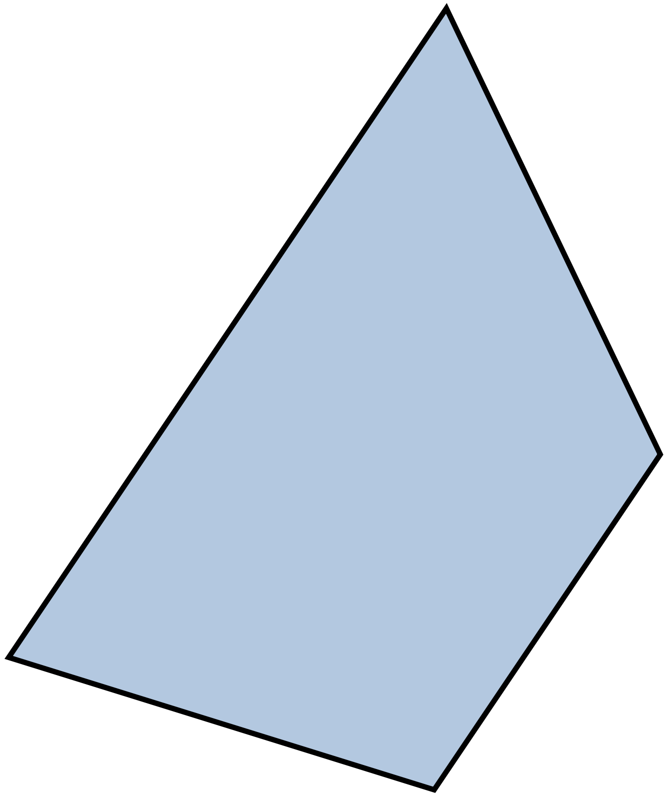
Image matching





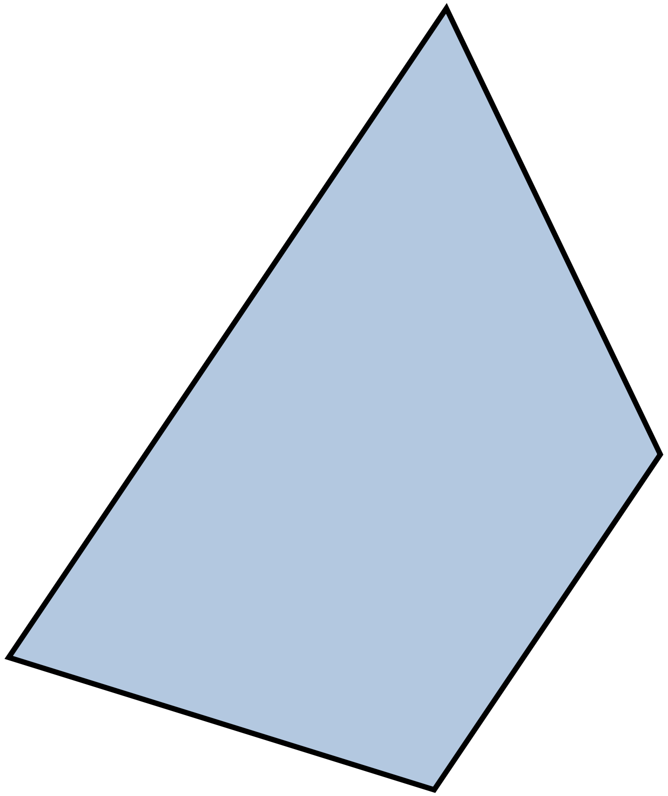
Pick a point in the image.
Find it again in the next image.

What type of feature would you select?



Pick a point in the image.
Find it again in the next image.

What type of feature would you select?



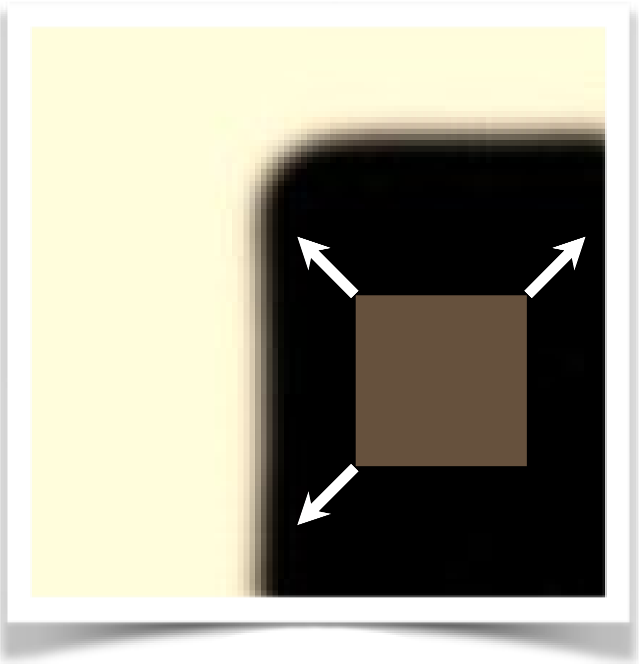
corner

Pick a point in the image.
Find it again in the next image.

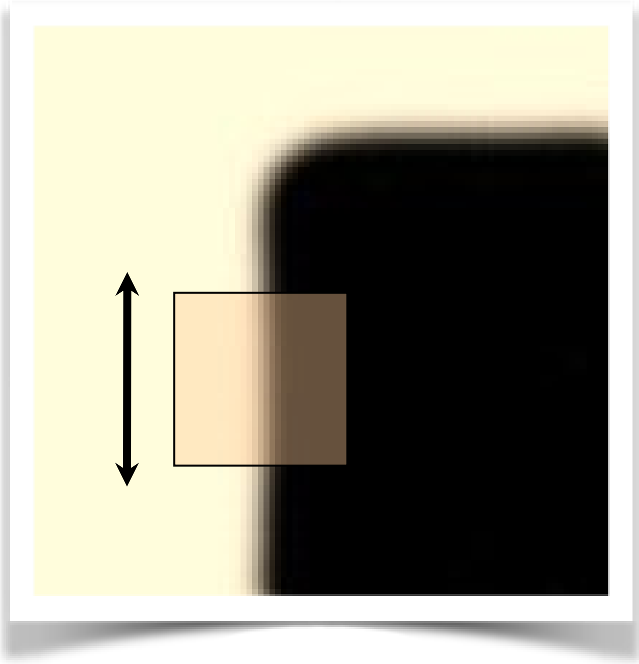
What type of feature would you select?

Easily recognized by looking through a small window

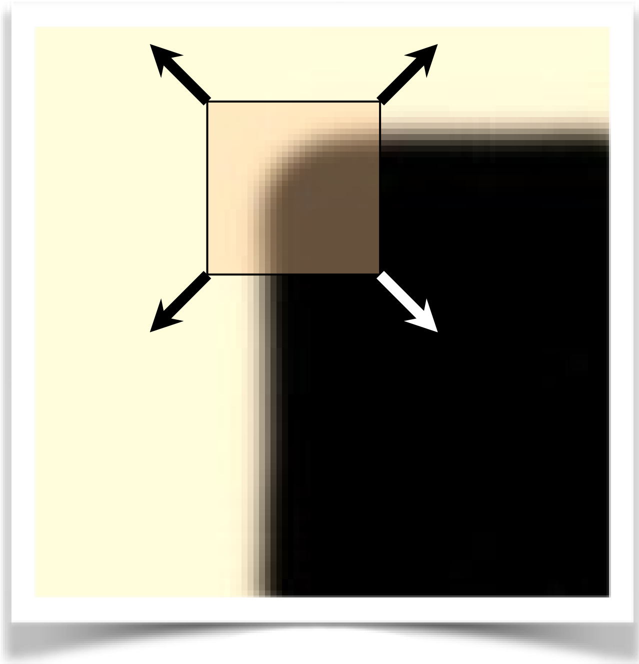
Shifting the window should give large change in intensity



"flat" region:
no change in all
directions



"edge":
no change along the edge
direction

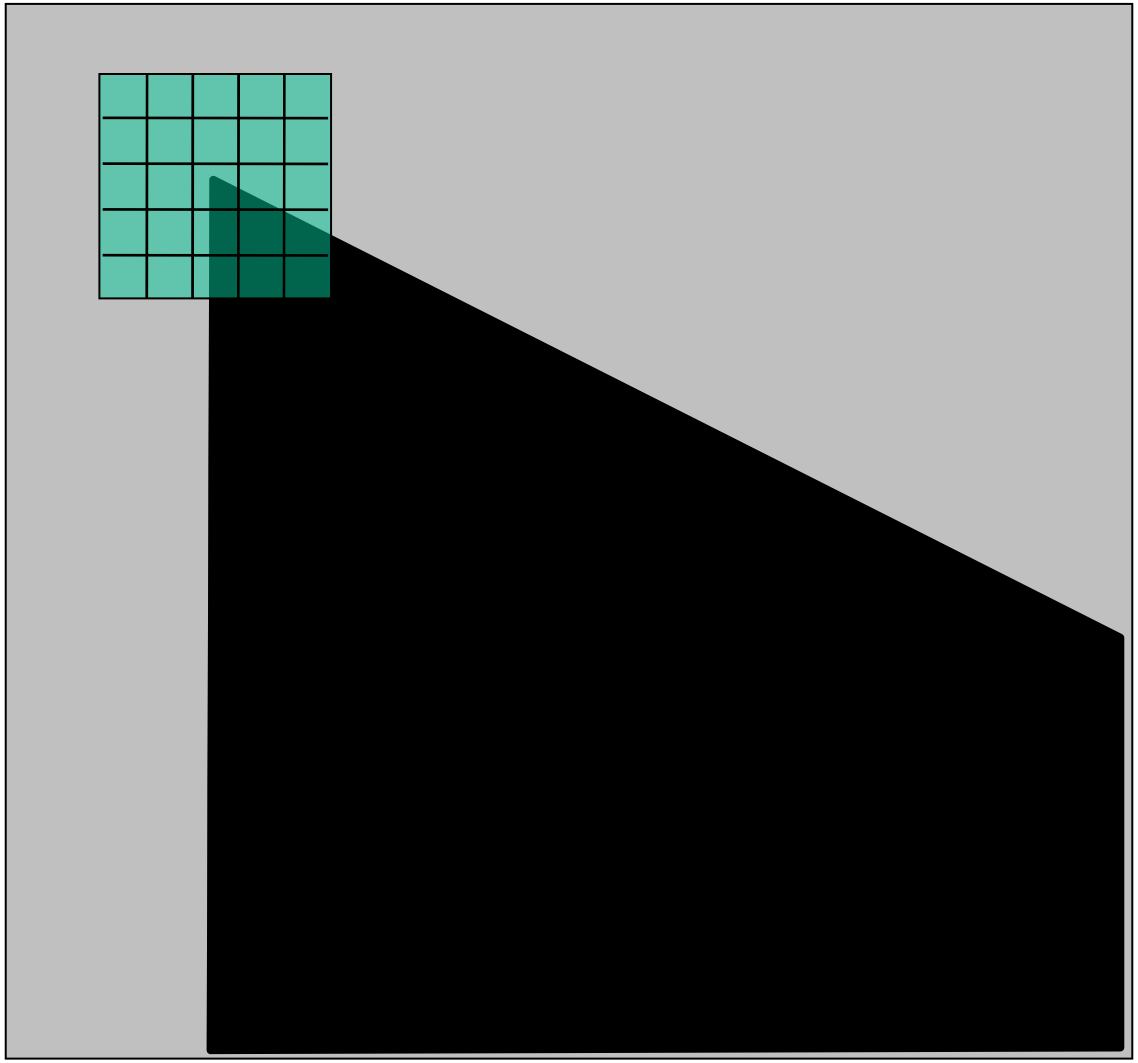


"corner":
significant change in all
directions

[Moravec 1980]

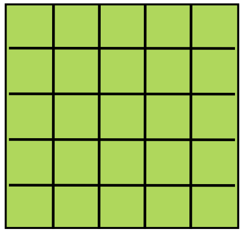
1. Compute image gradients over a small region

(not just a single pixel)



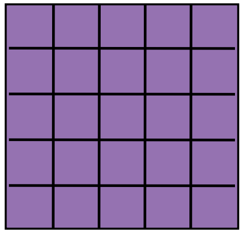
array of x gradients

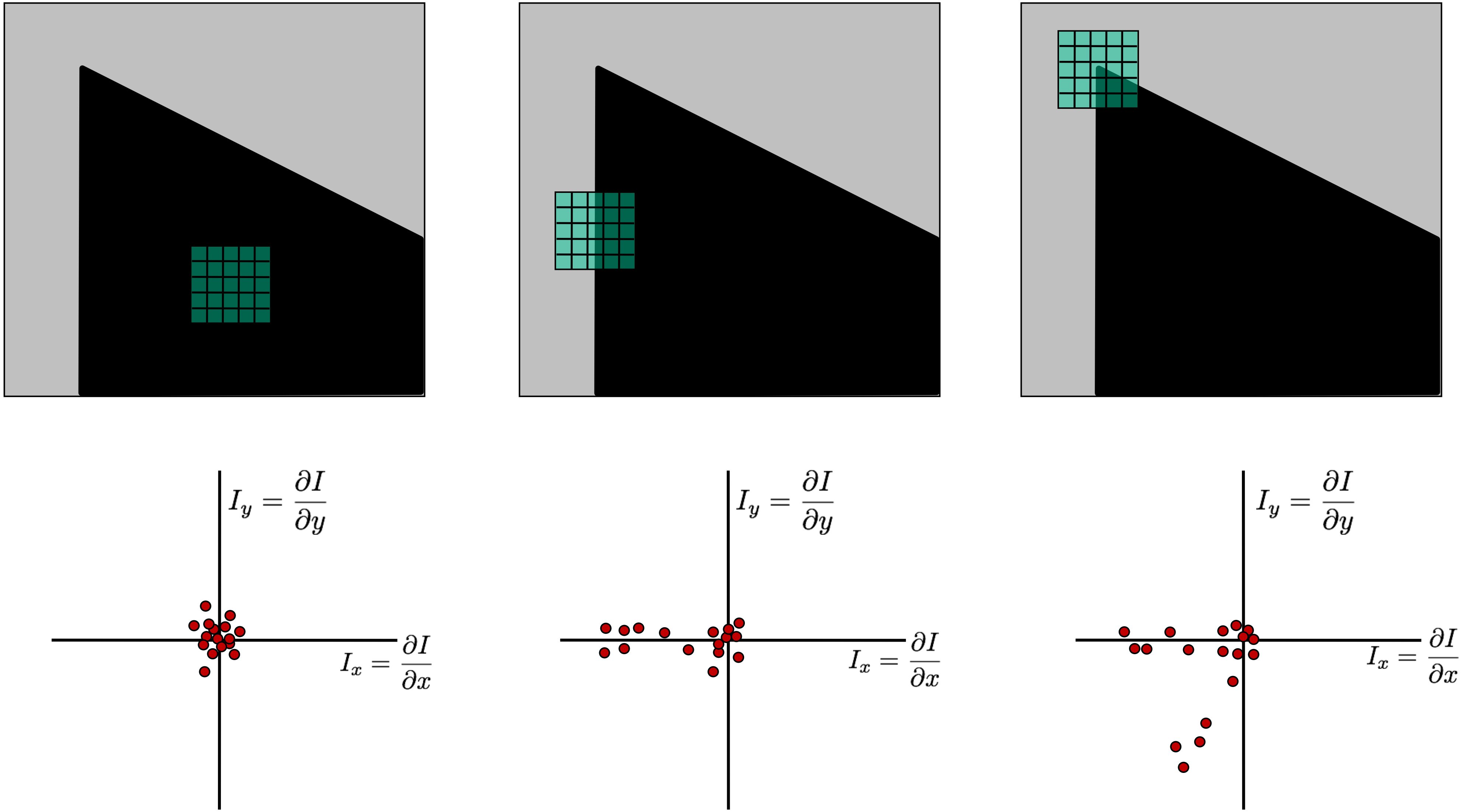
$$I_x = \frac{\partial I}{\partial x}$$



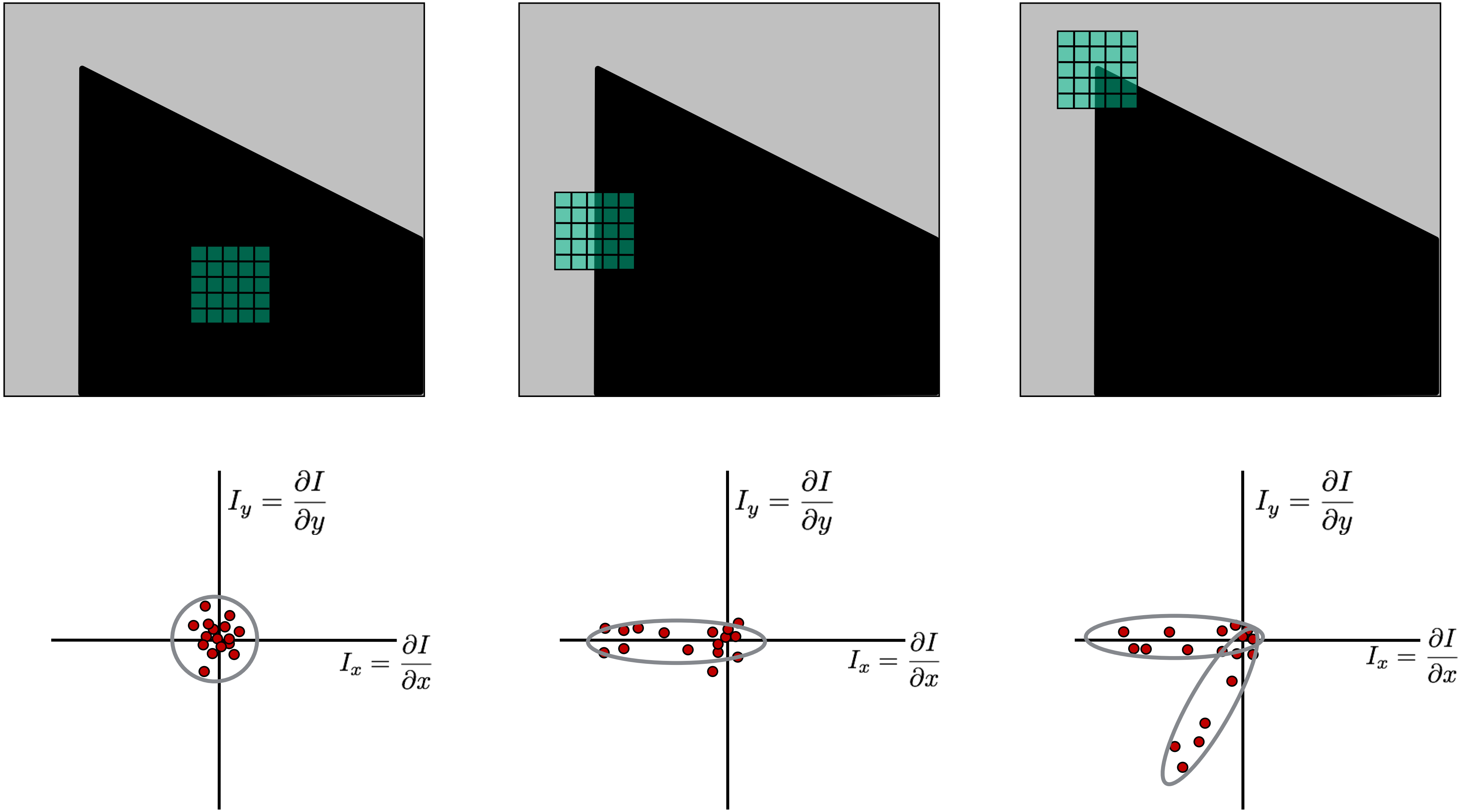
array of y gradients

$$I_y = \frac{\partial I}{\partial y}$$



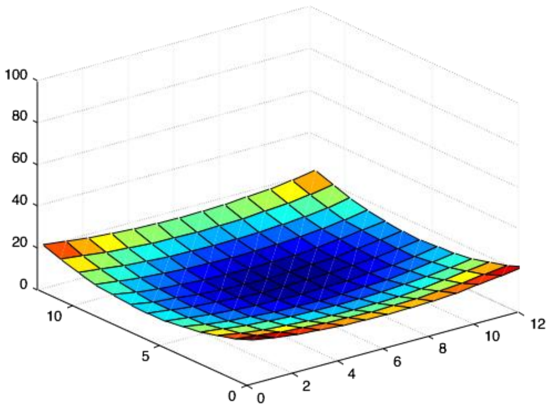


What does the distribution tell you about the region?

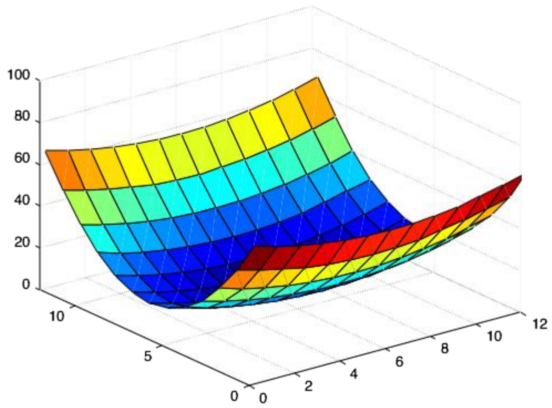


distribution reveals edge orientation and magnitude

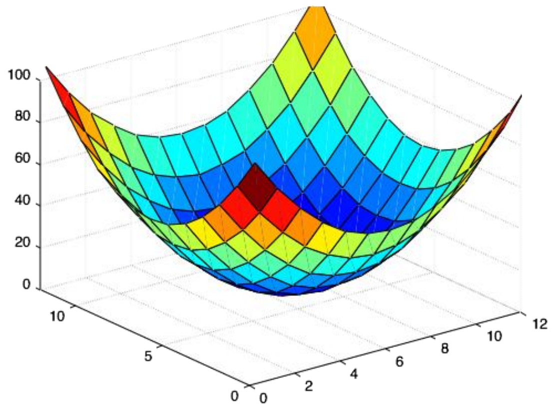
Which error surface indicates a good image feature?



flat



edge
‘line’



corner
‘dot’

Compute eigenvalues and eigenvectors

eigenvalue

$$\boldsymbol{M e} = \lambda \boldsymbol{e}$$

eigenvector

$$(M - \lambda I)\boldsymbol{e} = 0$$

Compute eigenvalues and eigenvectors

eigenvalue

$$M\boldsymbol{e} = \lambda\boldsymbol{e}$$

eigenvector

$$(M - \lambda I)\boldsymbol{e} = 0$$

1. Compute the determinant of $M - \lambda I$

(returns a polynomial)

Compute eigenvalues and eigenvectors

eigenvalue

$$\boldsymbol{M}\boldsymbol{e} = \lambda\boldsymbol{e}$$

eigenvector

$$(M - \lambda I)\boldsymbol{e} = 0$$

1. Compute the determinant of $M - \lambda I$

(returns a polynomial)

2. Find the roots of polynomial $\det(M - \lambda I) = 0$

(returns eigenvalues)

Compute eigenvalues and eigenvectors

eigenvalue

$$\mathbf{M}\boldsymbol{e} = \lambda\boldsymbol{e}$$

eigenvector

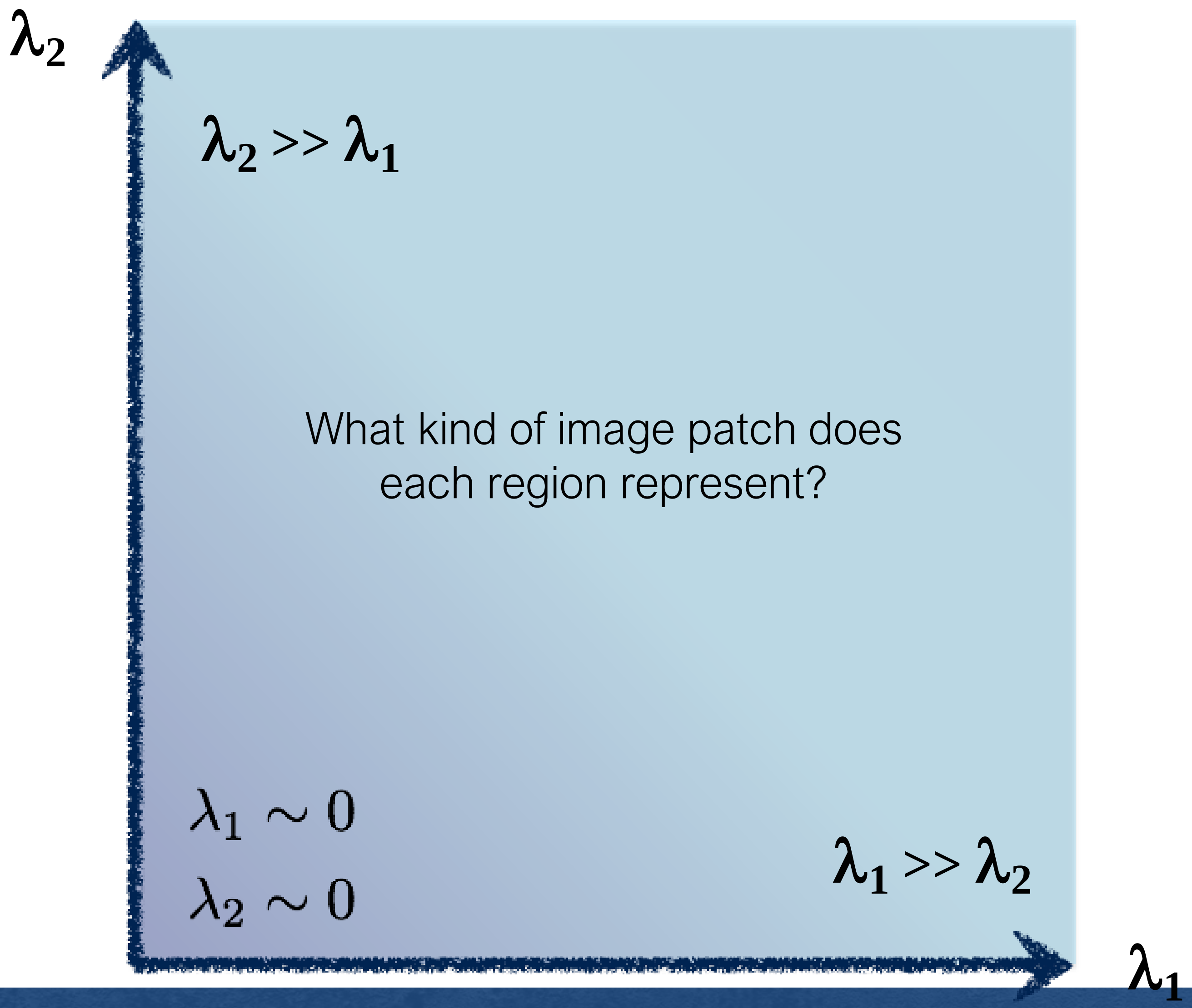
$$(M - \lambda I)\boldsymbol{e} = 0$$

1. Compute the determinant of $M - \lambda I$
(returns a polynomial)

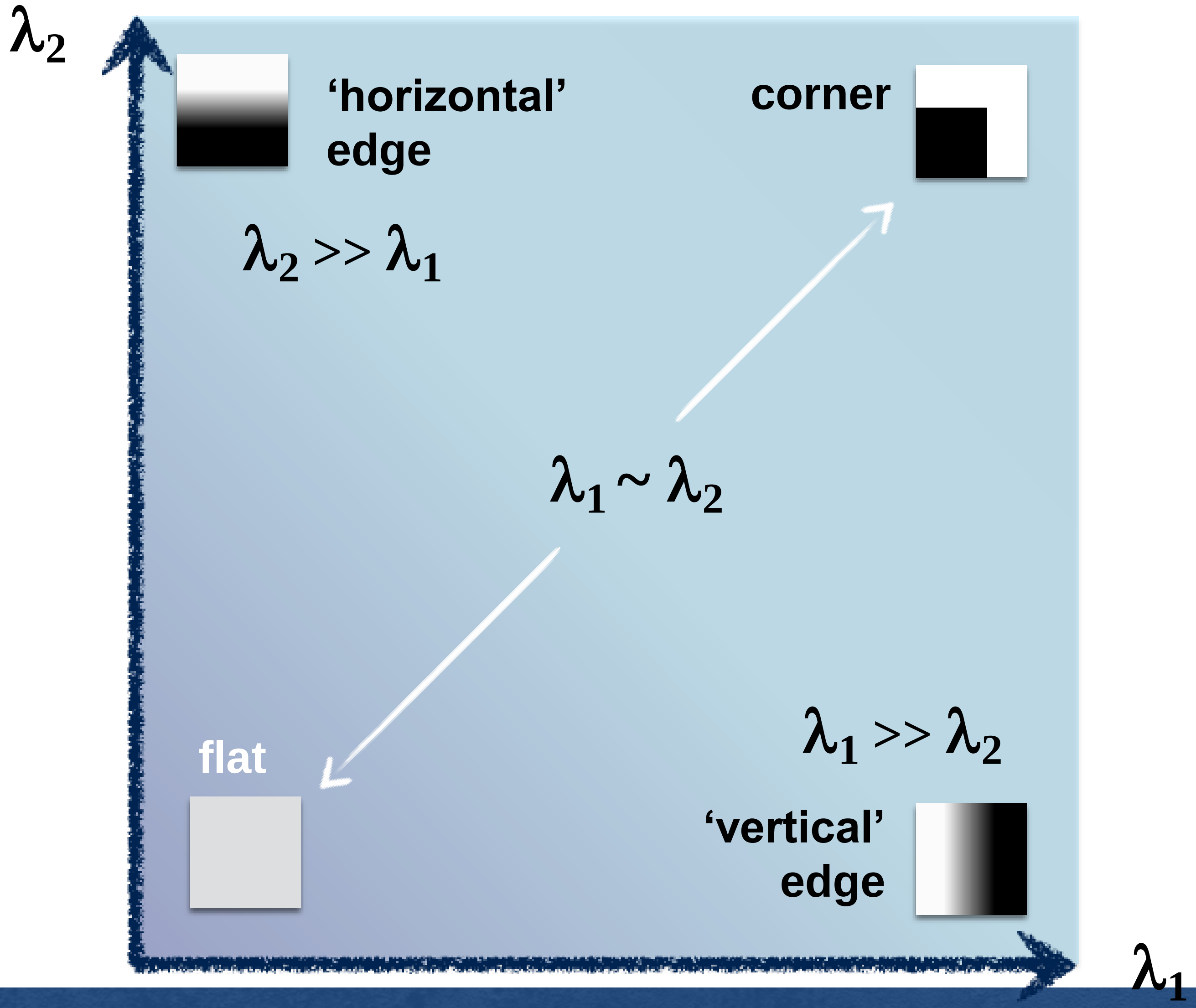
2. Find the roots of polynomial $\det(M - \lambda I) = 0$
(returns eigenvalues)

3. For each eigenvalue, solve $(M - \lambda I)\boldsymbol{e} = 0$
(returns eigenvectors)

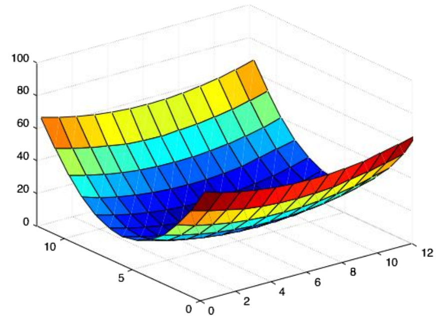
interpreting eigenvalues

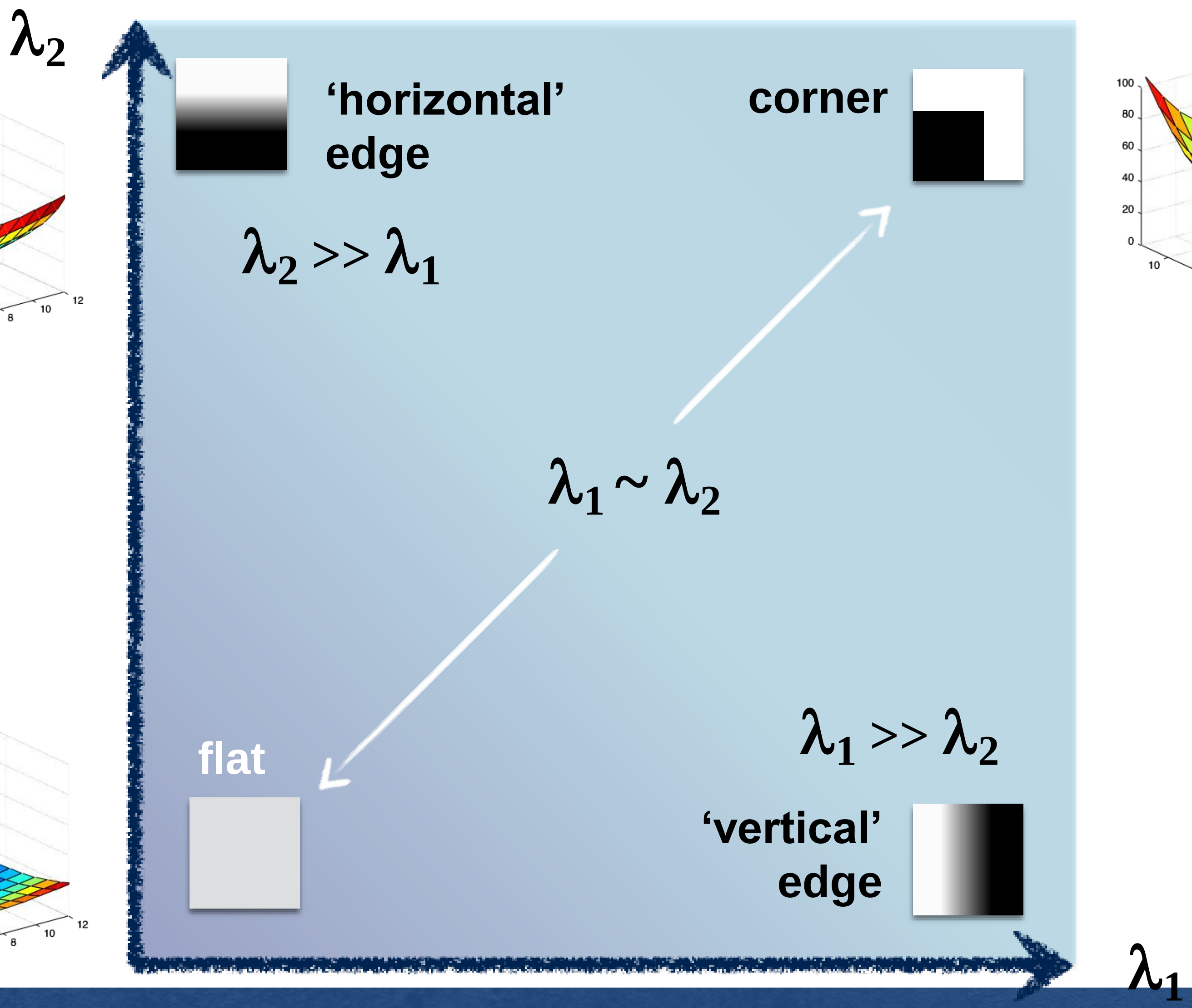


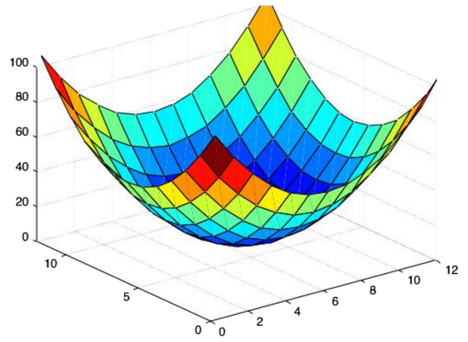
interpreting eigenvalues

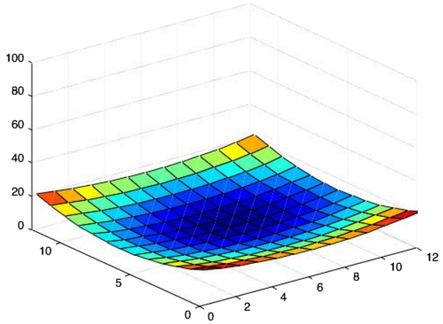


interpreting eigenvalues



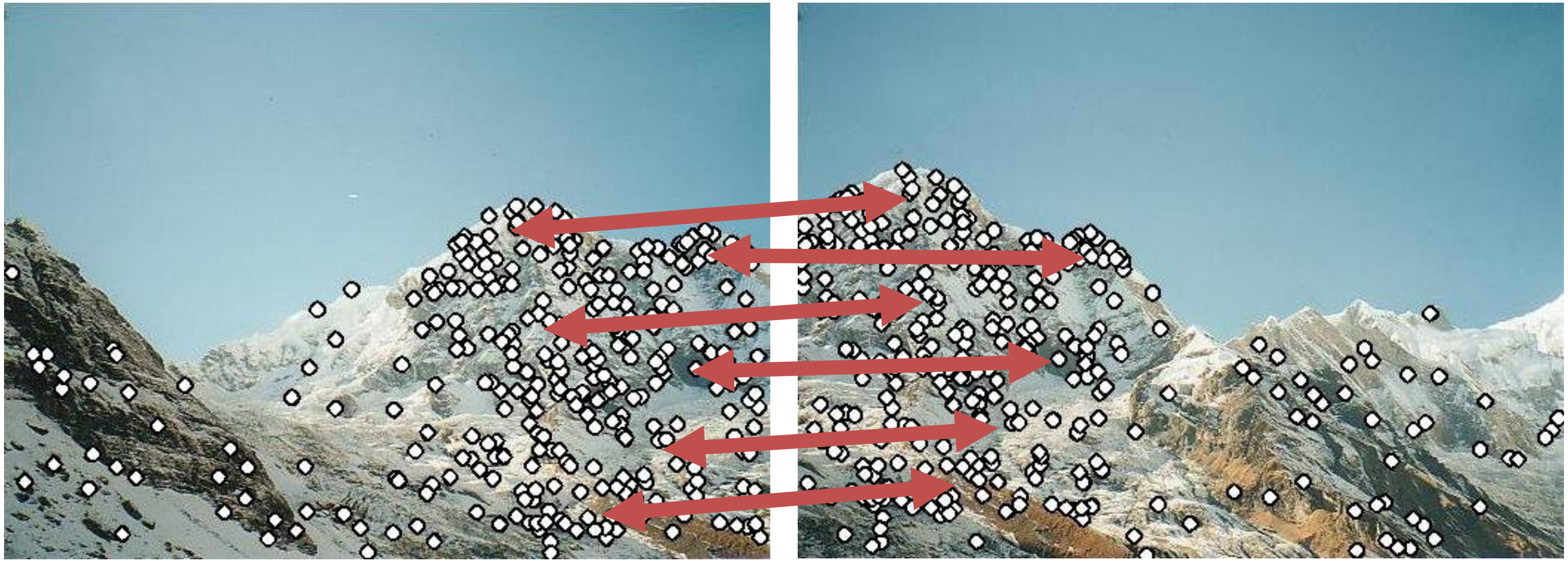






Finding + Matching

Finding and Matching



1: find corners+features

2: match based on local image data

Slide Credit: S. Lazebnik, original figure: M. Brown, D. Lowe