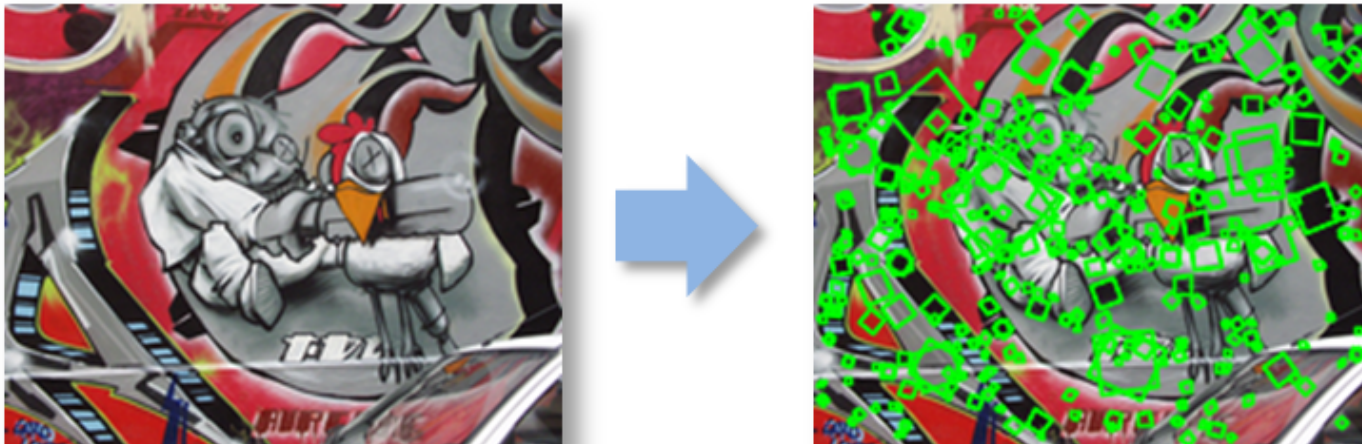
CSCI 497P/597P: Computer Vision



Lecture 9:

Image Features: ~~Overview and~~ Detection

Harris Corner

Announcements

- HW1 problems are out - deadline, etc finalized soon.

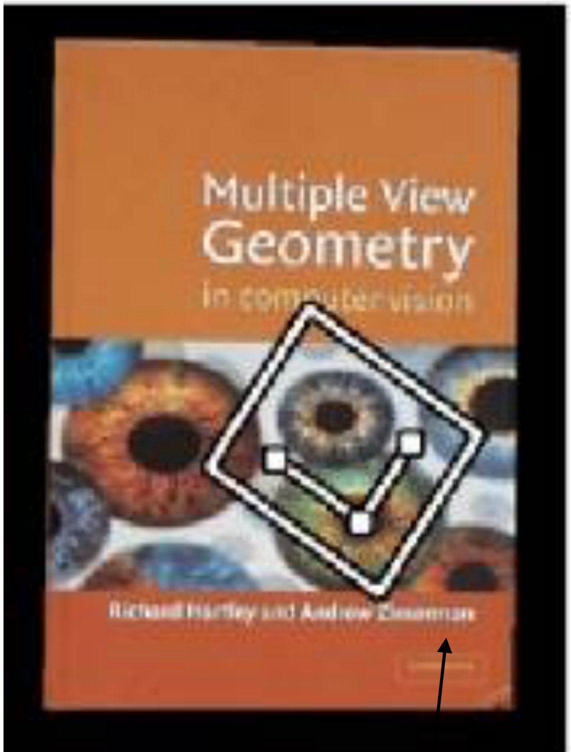
Goals

- Understand the derivation of the Harris corner detector.

Image Features

- Distinctive image elements used to help solve higher-level vision problems, like:
    - image matching
    - tracking
    - shape analysis and object recognition
- Tend to be more *compact* and more *information-dense* than raw pixels.

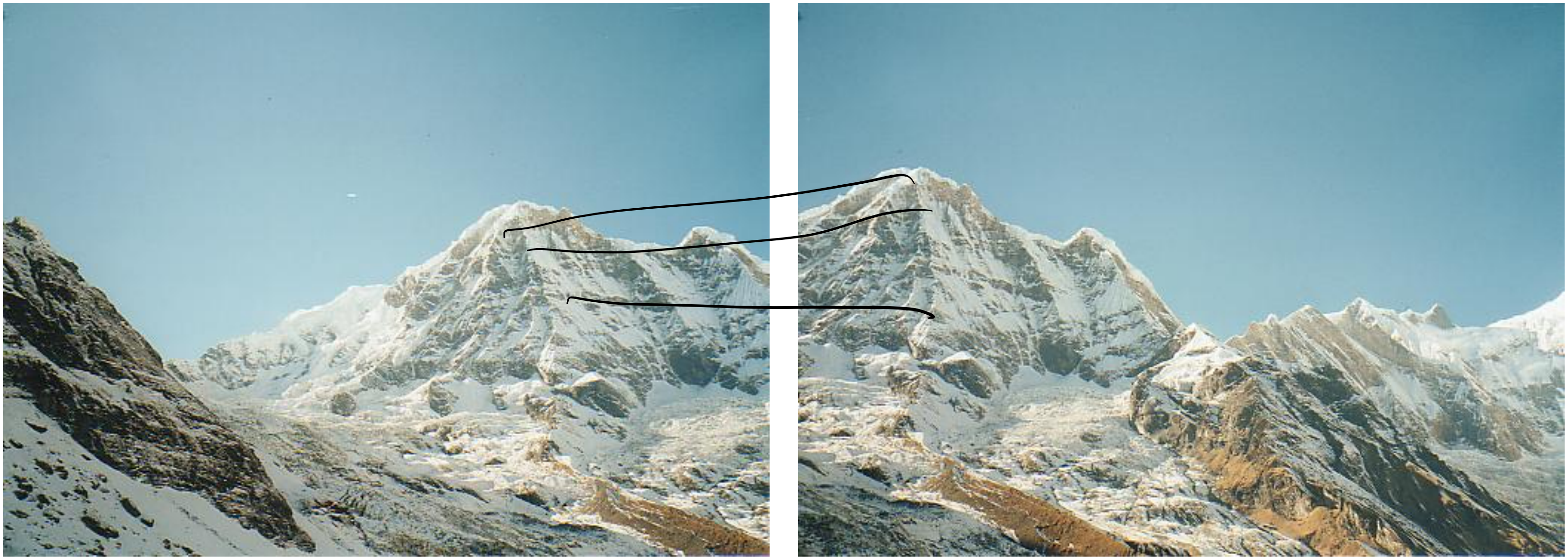
Image Matching



is this thing...

the same as this thing?

Running motivational example: Panorama Stitching



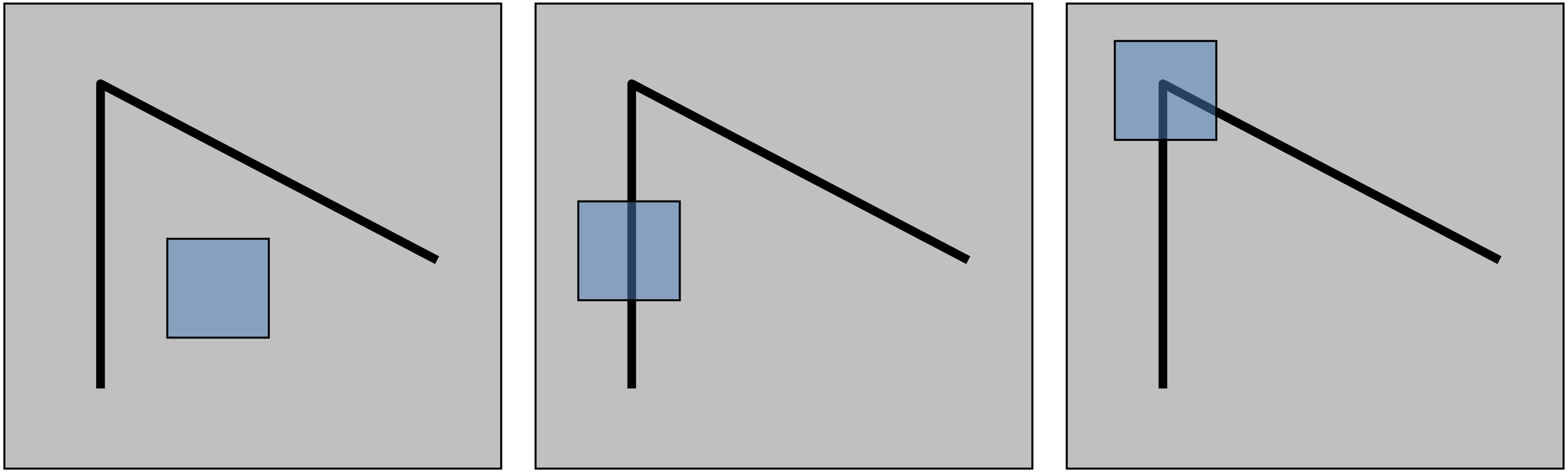
Running motivational example: Panorama Stitching

Corners as Features

- Let's use patches surrounding **corners** as features.

Corner features: a cartoon view

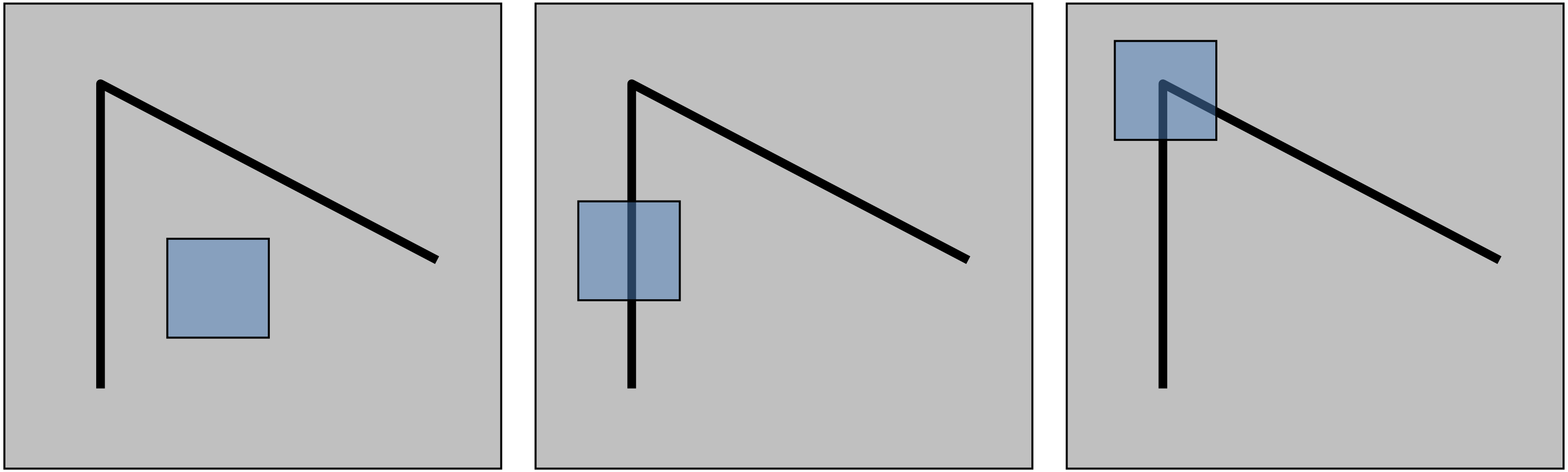
- Which of these patches is most "cornerish"?



- Can you write that down with math?

Corner features: a cartoon view

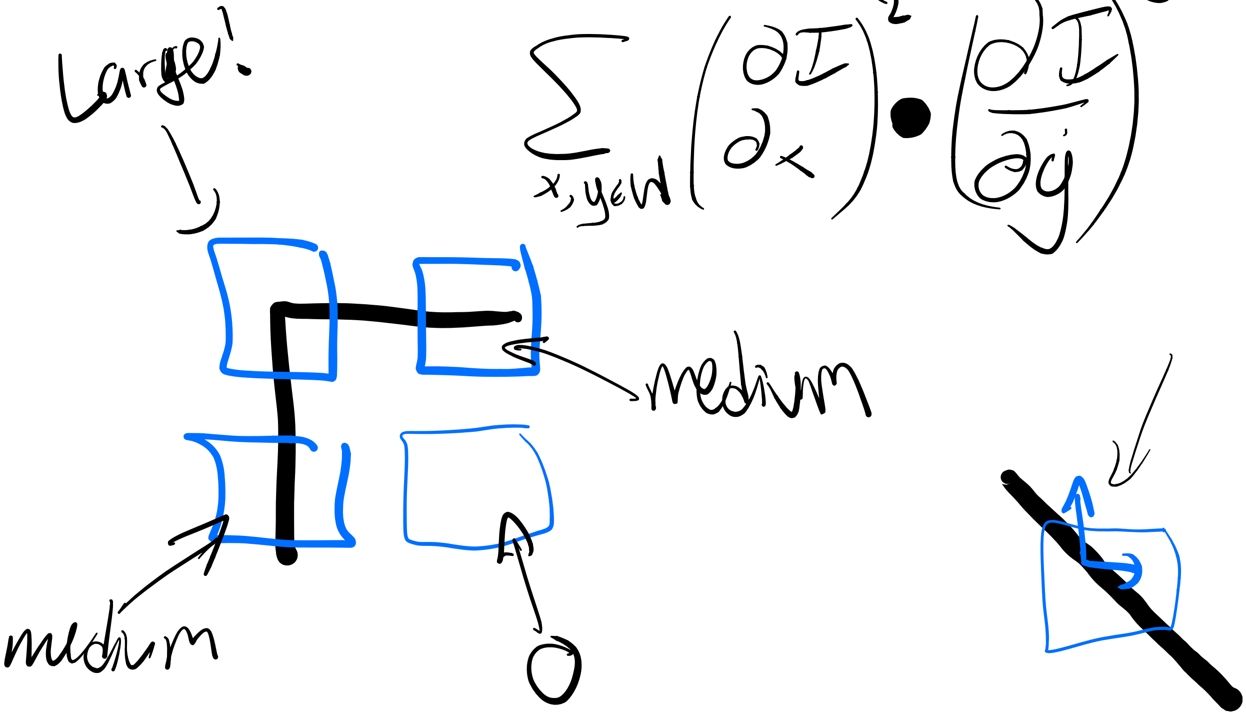
- Which of these patches is most "cornerish"?



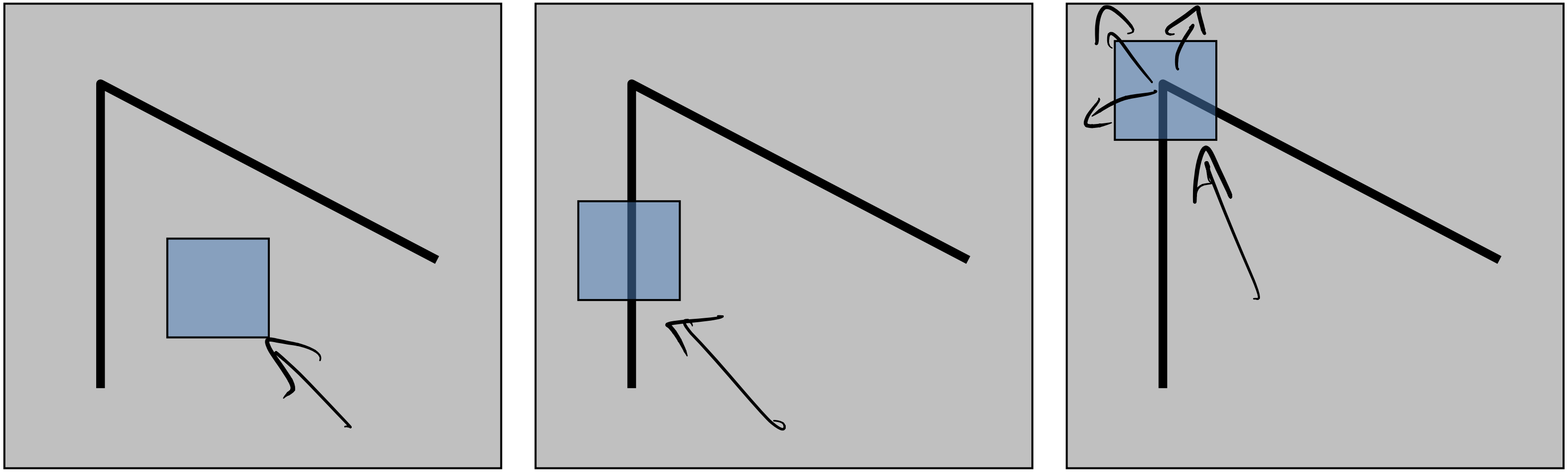
- Can you write that down with math?

Image Gradients?

$$\sum_{x,y \in W} \left(\frac{\partial I}{\partial x}\right)^2 \bullet \left(\frac{\partial I}{\partial y}\right)^2$$



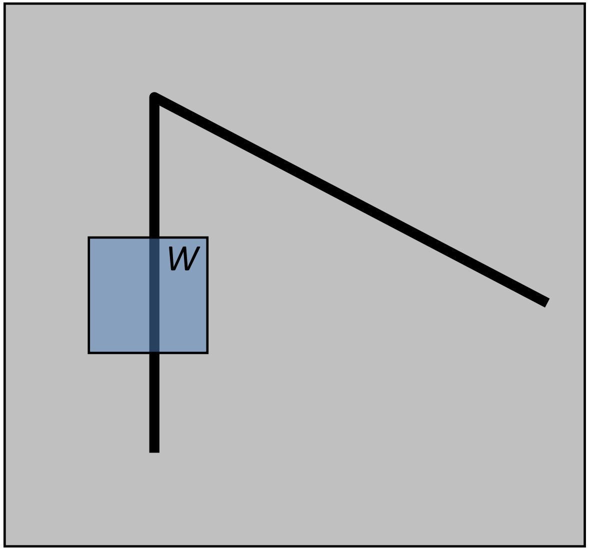
How can we measure that?



An expensive idea: compare each patch to **every other** patch

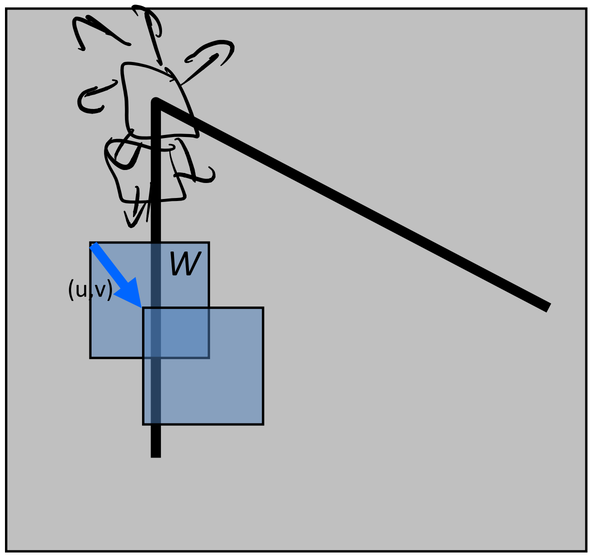
A less expensive idea: compare each patch to **nearby** patches.

In other words,



if you **nudge** the patch by (u, v), how much does its appearance change?

In other words,



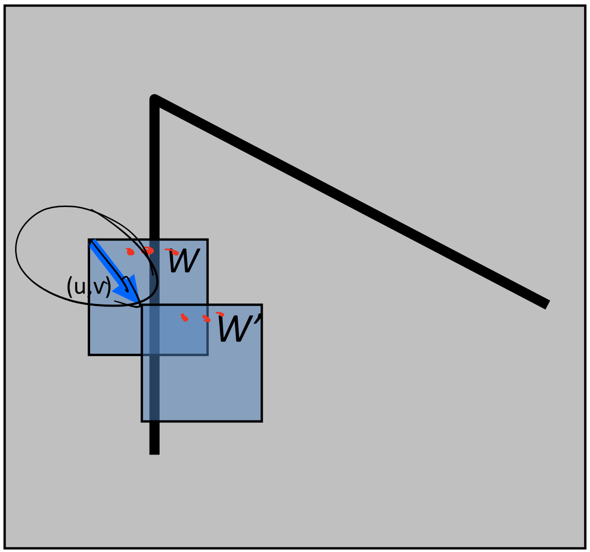
if you **nudge** the patch, how much does its appearance change?

Not at all in any direction? not unique

A lot in some directions? edge but not corner

A lot in **all** directions? corner

In mathier words,

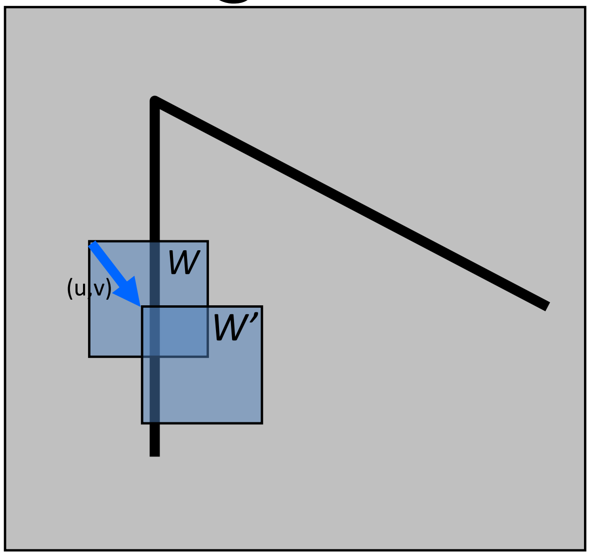


Start here: how different are W and W'?

$$E(u,v) = \sum_{x,y \in W} \left[I(x+u, y+v) - I(x,y) \right]^2$$

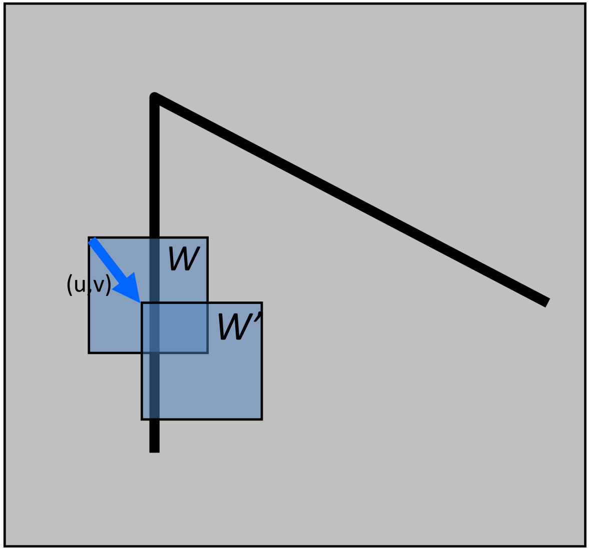
SSD

Efficiency hack: assume the image function is **locally linear**.



eh??

In mathier words,

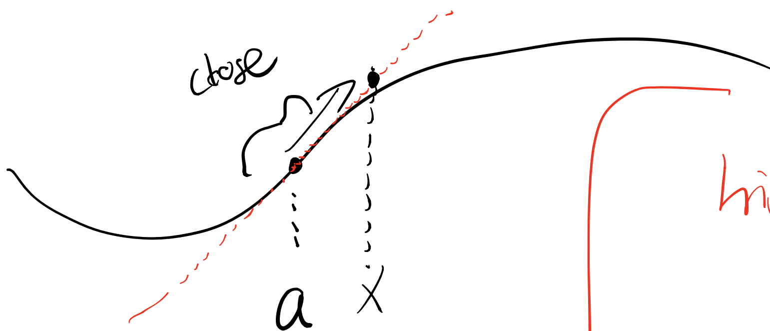


Start here: how different are W and W'?

Ultimately:

If E(u,v) is large for (u,v)s pointing in all directions,
then W is unique/distinctive/corner-like.

Taylor expansion!?



higher order terms

H.O.T.

$$f(x) = f(a) + f'(a)\cdot(x-a) + f''(a)(x-a)^2 + \cdots\cdots$$

↑ 1st derivative

$$I(x+u, y+v) \approx I(x,y) + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \text{H.O.T.}$$

locally linear model of I

$$I_x = \frac{\partial I}{\partial x} \qquad I_y = \frac{\partial I}{\partial y}$$

x deriv. y deriv

Plug the approximation into E...

...and do some good ol' algebraic manipulation

$$E(u,v) = \sum_{x,y \in W} \left[I(x+u, y+v) - I(x,y) \right]^2$$

$$\approx \sum_{x,y \in W} \left[I(x,y) + I_x u + I_y v - I(x,y) \right]^2$$

$$= \sum_{x,y \in W} \left[I_x u + I_y v \right]^2$$

$$E(u,v) = \sum_{x,y \in W} I_x^2 u^2 + 2 I_x I_y uv + I_y^2 v^2$$

$$E(u,v) = \sum_{x,y \in W} \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

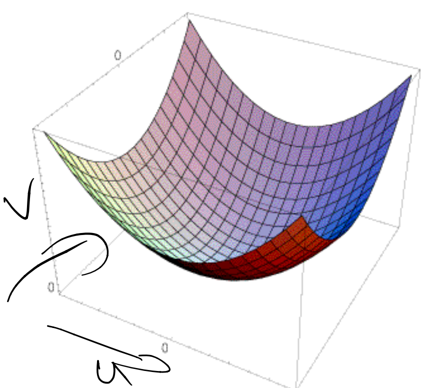
$$E(u,v) = \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

↑ structure tensor

What does this mean?

- E(u, v) tells us how much the window changes in the direction (u, v).
    - Because of our approximations, this is a **linear** model in image space, and a **quadratic model** in the error function.

Plotting E(u, v) in 2D would look something like:



Its shape is determined by the "structure tensor" matrix

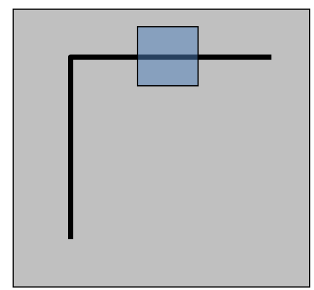
$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} A & B \\ B & C \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y) \in W} I_x^2$$

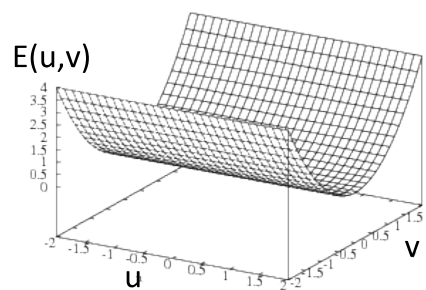
$$B = \sum_{(x,y) \in W} I_x I_y$$

$$C = \sum_{(x,y) \in W} I_y^2$$

One scenario:



$$H = \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix}$$



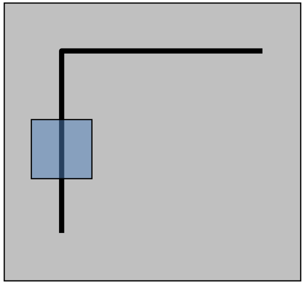
$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} A & B \\ B & C \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$A = \sum_{(x,y)\in W} I_x^2$$

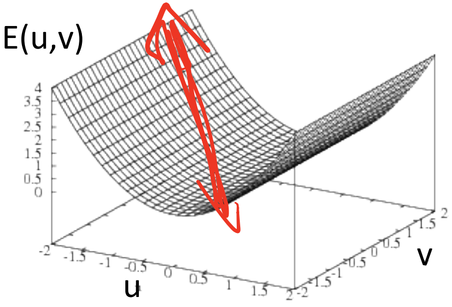
$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$

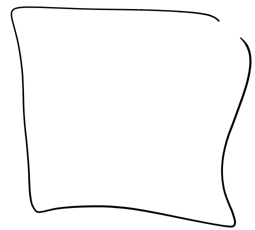
One scenario:



$$H = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$$

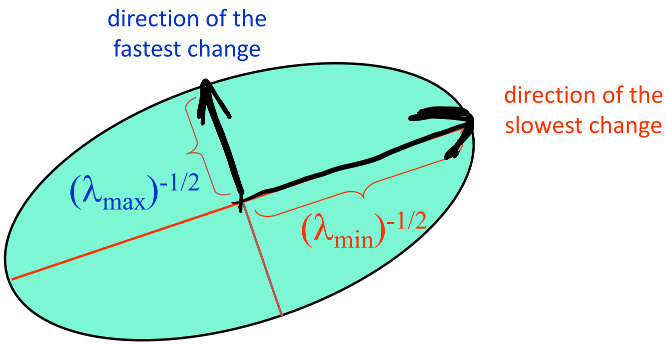


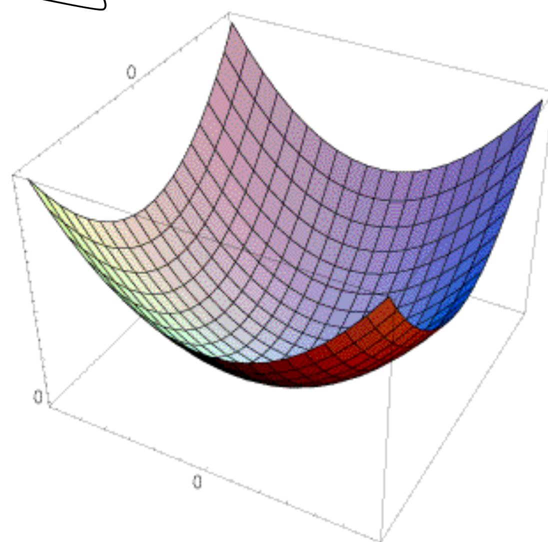
In general...



x_1 λ_1

x_2 λ_2





Eigenvectors!?

$$Ax = \lambda x$$

λ: eigval, scalar

x: eigenvector

$$\underbrace{\min_{\substack{x \\ \|x\|=1}} \|x^T A x\|}_{\lambda_{min}}$$

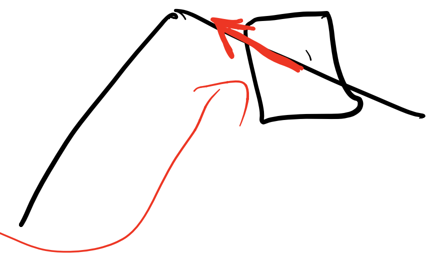
$$A = \sum I_x^2$$

$$H = \begin{bmatrix} A & B \\ B & C \end{bmatrix}$$

$\vdots$

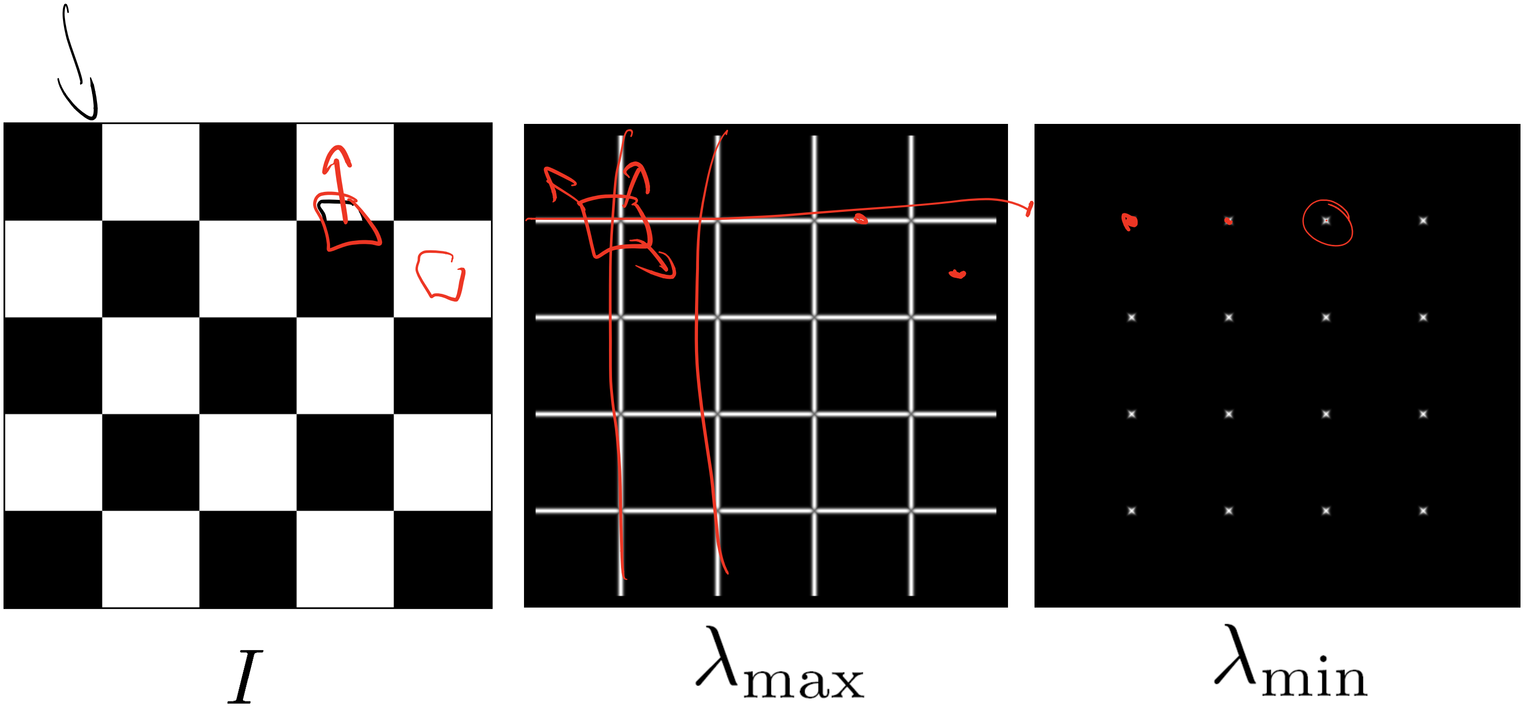
$\lambda_{min} = 0$

x_{min}



Corner Detection: Upshot

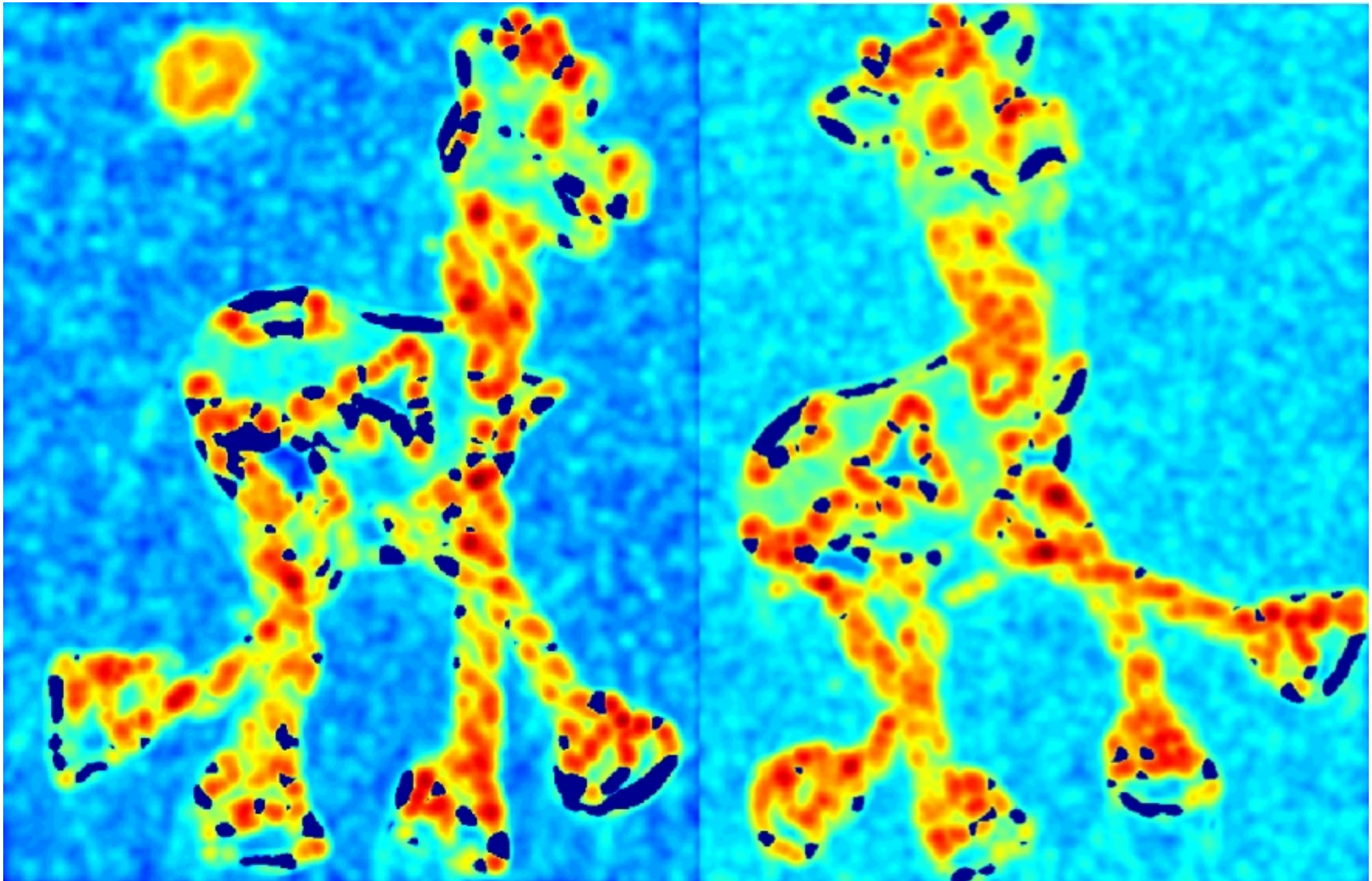
- The **smaller** eigenvalue of H is **large** when the patch is centered on a corner.



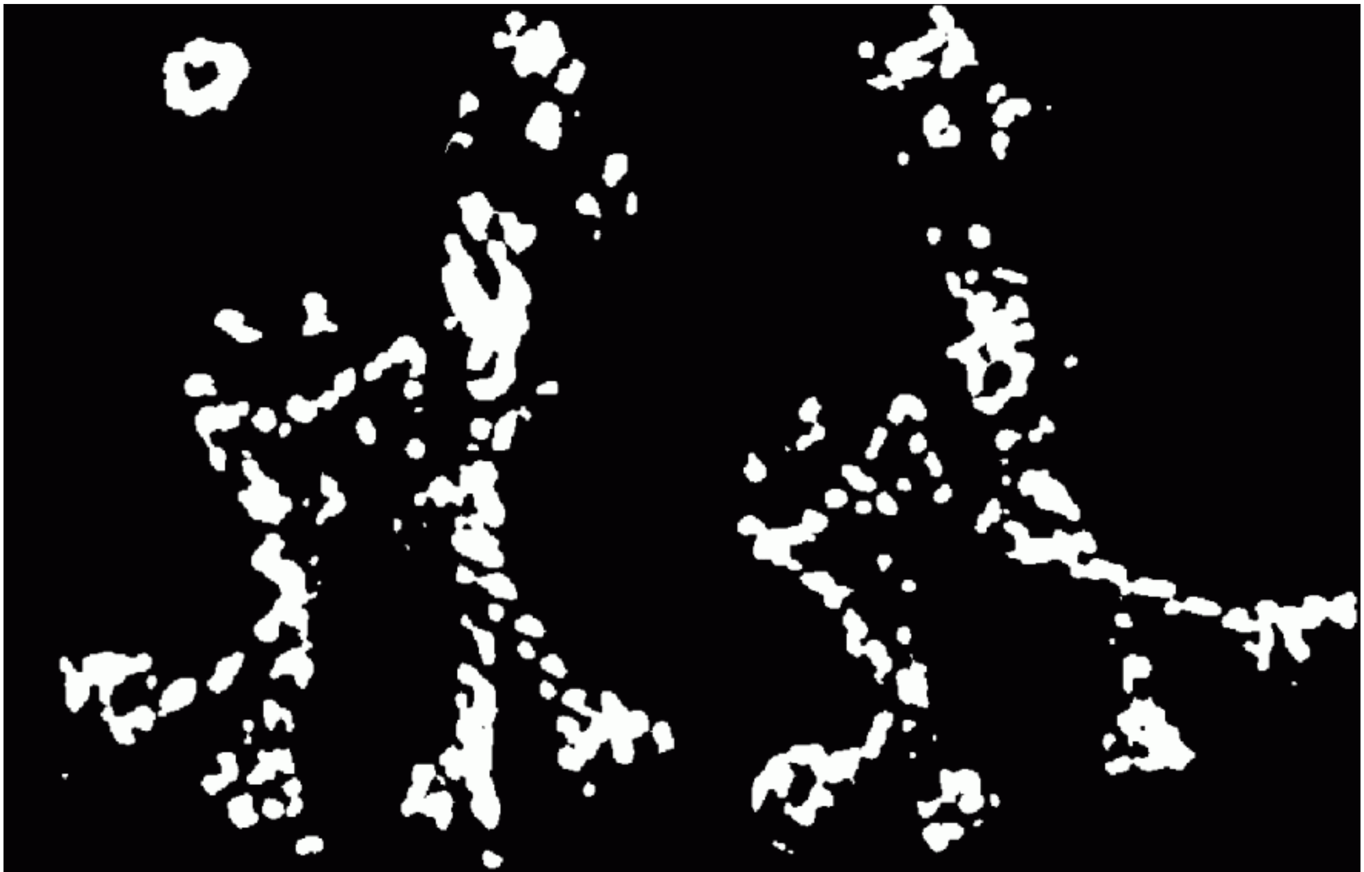
Input

red = large
blue = small

Smallest eigenvalue

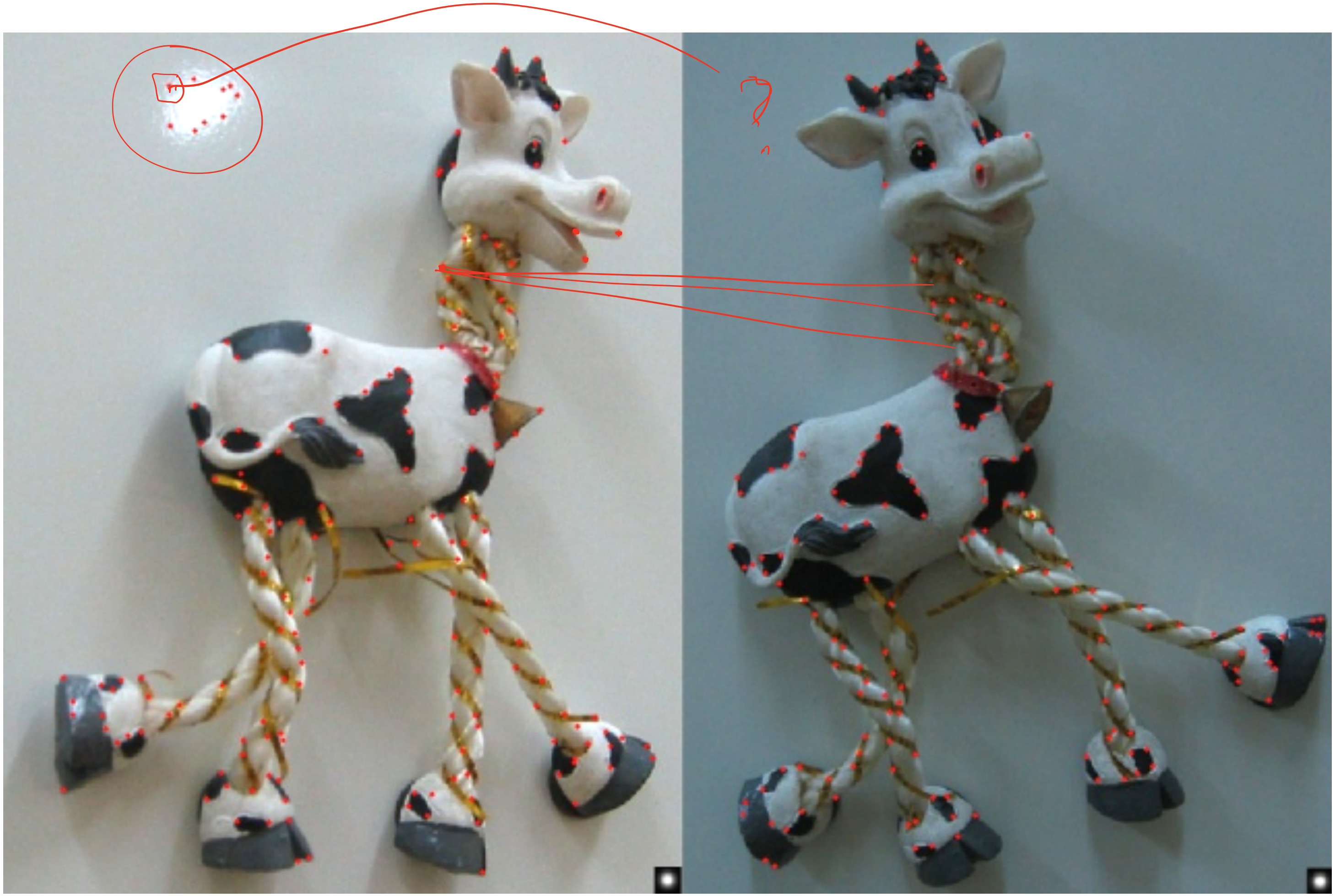


Thresholded



Keep only Local Maxima

Resulting Corners



How do you compute it?

$$A = \sum_{(x,y)\in W} I_x^2$$

$$B = \sum_{(x,y)\in W} I_x I_y$$

$$C = \sum_{(x,y)\in W} I_y^2$$