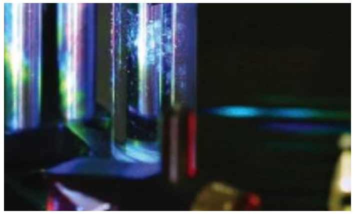
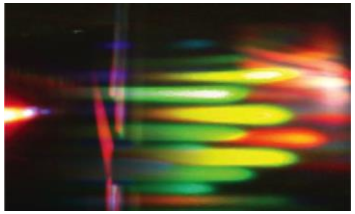
Imperial College London





COMP70058 Computer Vision

Lecture 9 – Interest Point Detection

Stamatia (Matina) Giannarou, PhD
The Hamlyn Centre for Robotic Surgery

stamatia.giannarou@imperial.ac.uk

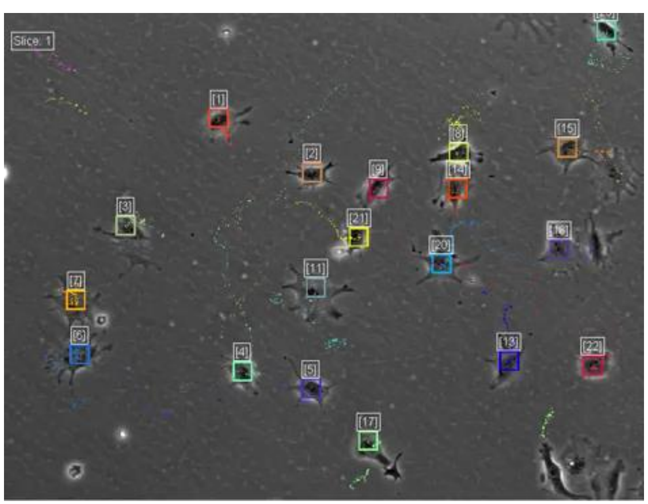
The Hamlyn Centre
for Robotic Surgery

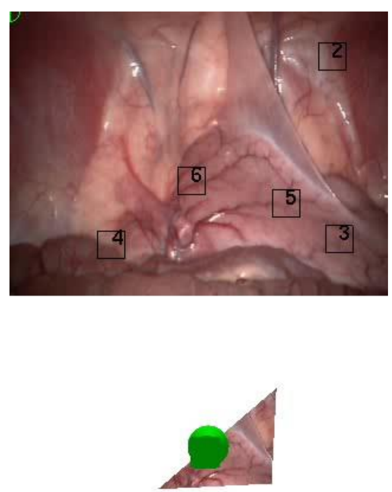
Contents

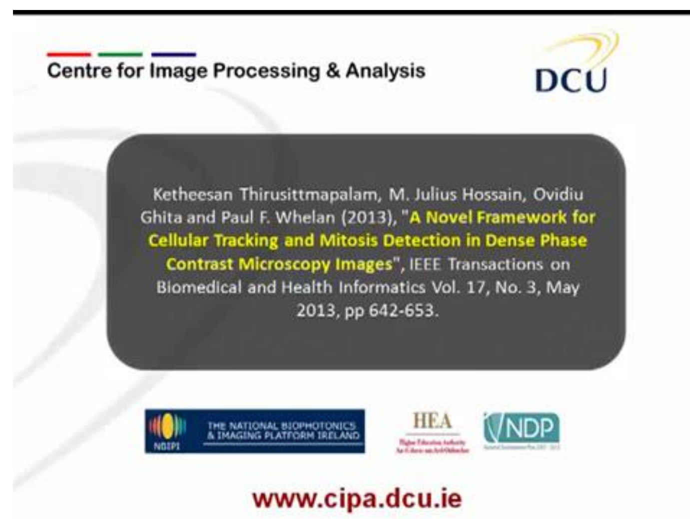
- Feature Detection
- Interest Points in Vision
- Corner Detection and Harris Corner Detector
- Shi-Tomasi Corner Detector
- Automatic Scale Selection

Applications of Image Sequence Processing

- Image alignment and stitching
- Object recognition
- 3D reconstruction and modelling
- Motion tracking
- Indexing and content-based retrieval
- Robot mapping and navigation
- Gesture recognition



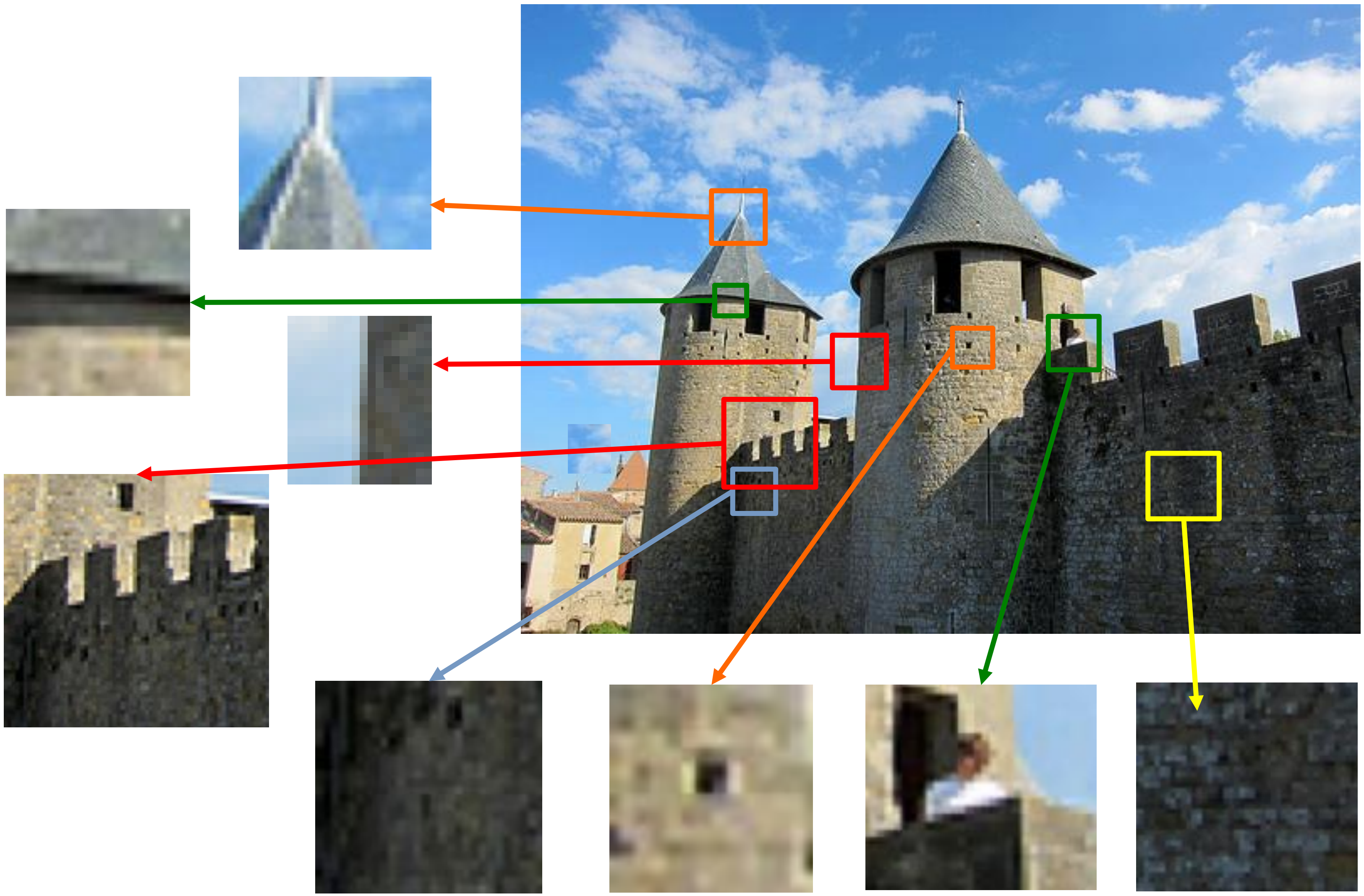




Feature Detection

- What is a feature?
  - An interesting part of an image
- Why local features?
  - Locality – local features are associated with object features
  - Generalisability – local features are more generalisable for scene representation
  - Abundance – many of them, often with redundancy (numerically attractive)
  - Efficiency – fast to calculate, thus good for real-time performance.
- Types of features
  - Edges
  - Corners (when there is high curvature in image gradient)
  - Blobs
  - Ridges

What Features are Good for Tracking



Interest Points in Vision

- Well defined position in images;
- Incorporate useful local image context (e.g., distinctive texture patch);
- Stable under local or global changes (e.g., illumination, contrast);
- Immune to geometrical changes (e.g., scale, translation, rotation);
- Generalisable to natural scenes and preferably with good mathematical representation.

Edges	Corners	Blobs	Ridges
Sobel Prewitt Roberts Robinson Canny Canny-Deriche	**Harris** **Shi & Tomasi** Yang & Chabat Level Curvature Susan AST/FAST	LoG DoG DoH (Determinant of Hessian) MSER (Maximally stable extremal regions) PCBR (Principal curvature based region)	Ridge sets Valley sets Relative critical sets

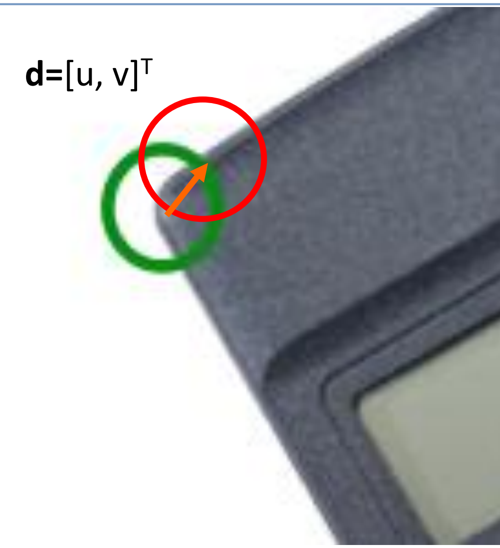
Interest Points in Vision

Why Corners

- Movement in uniform region results in no change in pixel distribution within the window
- Movement along an edge also results in no change in pixel distribution within the window
- For corners, any movement will result in significant changes

Harris Corner Detector

- Let's look at a small window W, when shifted by **d**=[u, v]^T
- The changes in intensity distribution due to **d** can be expressed as the squared difference of I(*x*+*u*, *y*+*v*)-I(*x*,*y*) within the window



$$\varepsilon(u,v) = \sum_{x,y \in W} \Big(I(x+u, y+v) - I(x,y) \Big)^2$$

For uniform patches, this will be near 0 and for distinctive patches such as corners, it will be large, therefore we are interested in maximising ε(u,v) while searching for corners

By Taylor series expansion $I(x+u, y+v) = I(x,y) + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \ldots$

$$\varepsilon(u,v) \approx \sum_{x,y \in W} \left(I(x,y) + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v - I(x,y) \right)^2, \text{ define } I_x \equiv \frac{\partial I}{\partial x} \text{ and } I_y \equiv \frac{\partial I}{\partial y} \text{ we have}$$

$$\varepsilon(u,v) \approx \sum_{x,y \in W} \left(uI_x + vI_y \right)^2 = \sum_{x,y \in W} \left(u^2 I_x^2 + 2uvI_x I_y + v^2 I_y^2 \right)$$

Harris Corner Detector

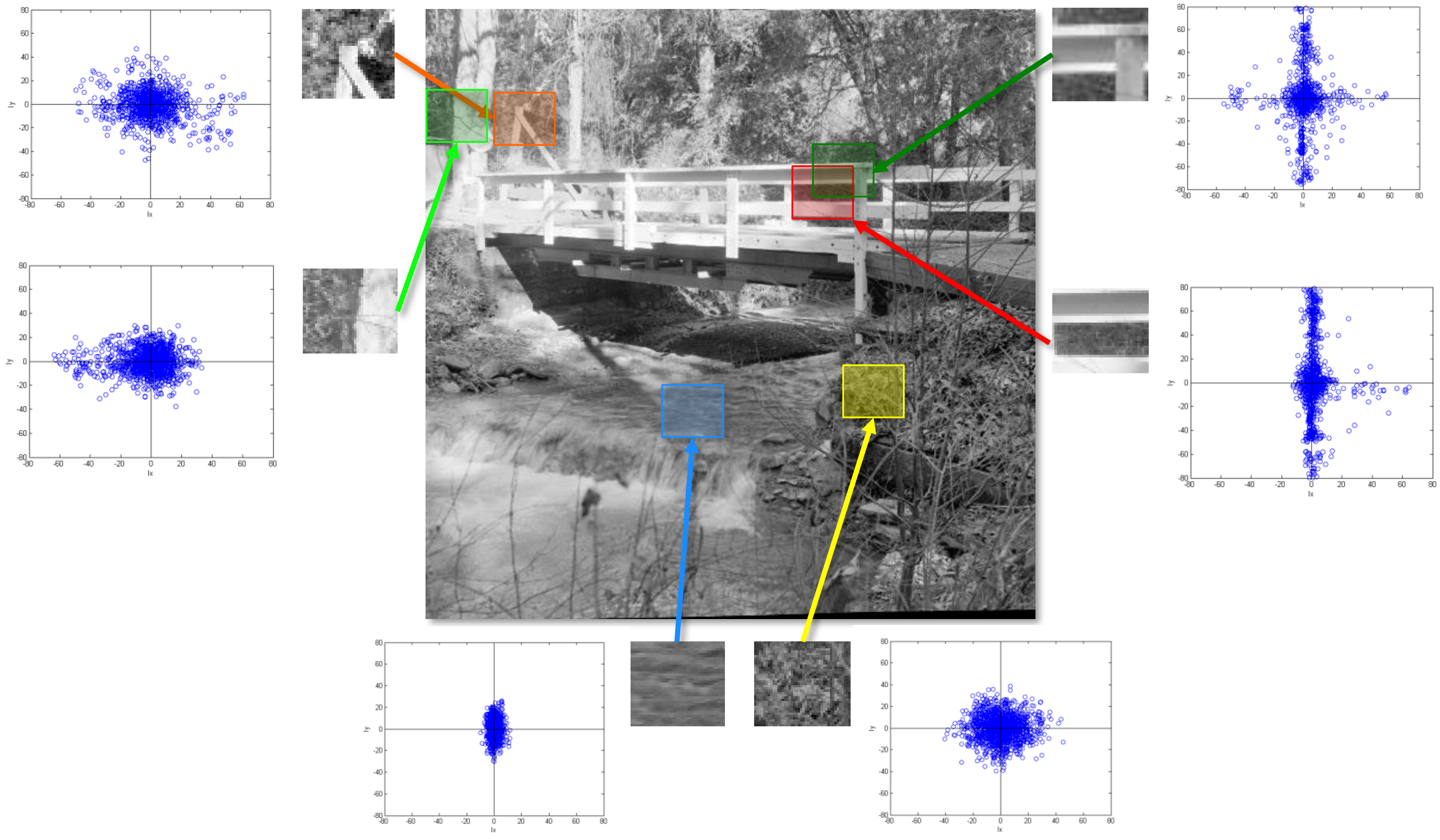
$$\varepsilon(u,v) \approx \sum_{x,y\in W} \left(uI_x + vI_y\right)^2 = \sum_{x,y\in W} \left(u^2 I_x^2 + 2uvI_x I_y + v^2 I_y^2\right)$$

- When expressed in matrix form, we have the following equivalent expression

$$\varepsilon(u,v) \approx [u,v] \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$\varepsilon(u,v) \approx \mathbf{d}^T \mathbf{M} \mathbf{d}$$

- We will get a large $E(u, v)$, where image derivatives are large.
- M is a 2x2 matrix which can provide information about the directions of change.



Ix
Iy

Harris Corner Detector

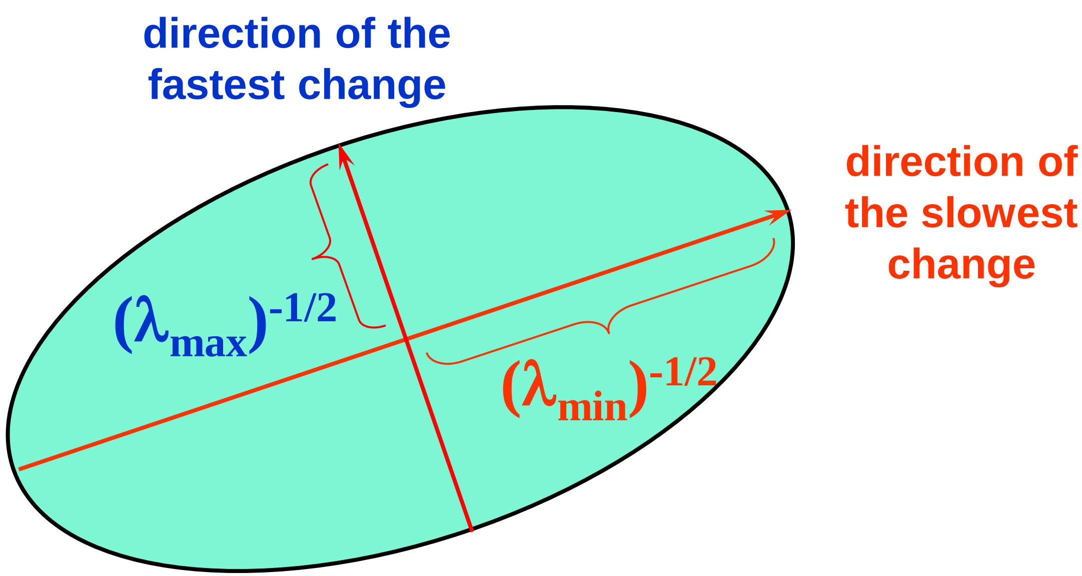
- Since M is a real symmetric matrix, we can decompose M as:

$$M = P \Lambda P^T$$

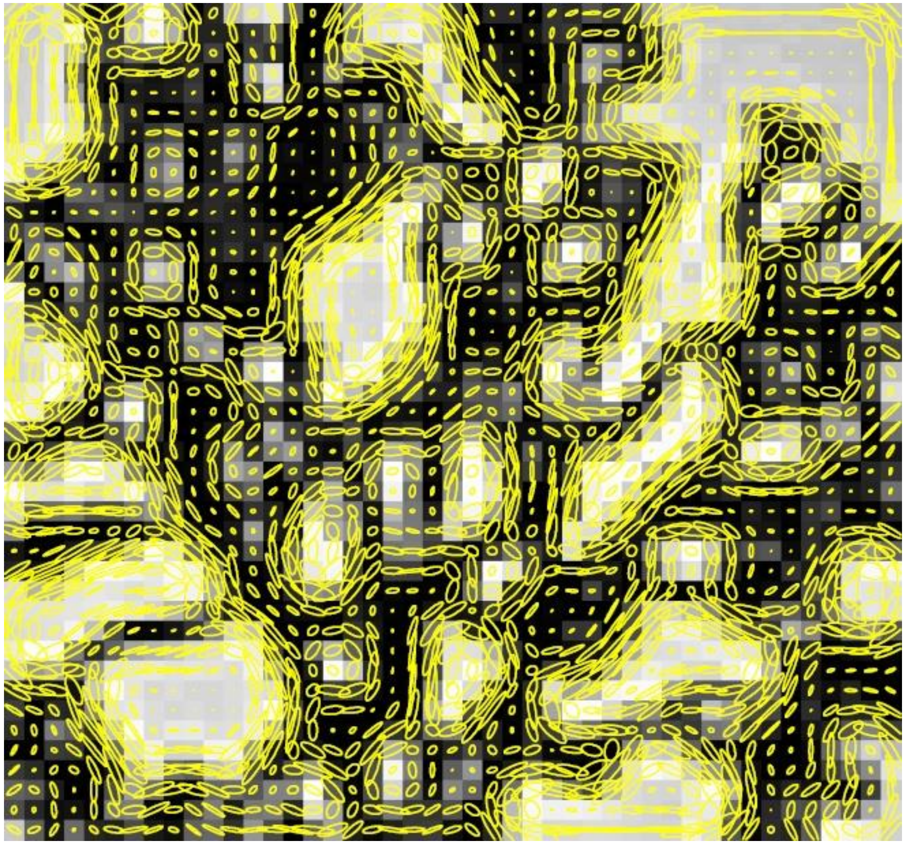
each column is an eigenvector (for P)

diagonal matrix with eigenvalues $\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$

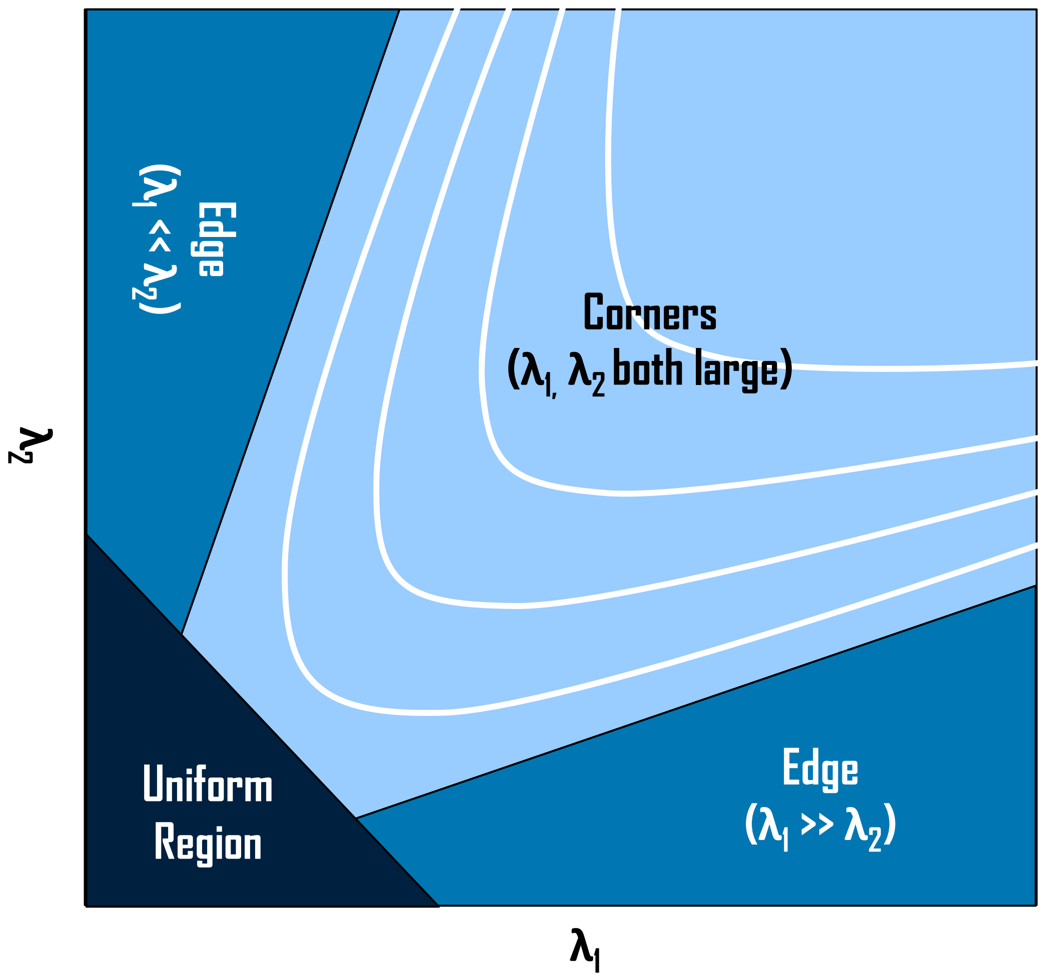
each row is an eigenvector (for P^T)



Corner Response Map



Corner Response Map



Harris Corner Detector

$$\varepsilon(u,v) \approx \sum_{x,y\in W} \left(uI_x + vI_y\right)^2 = \sum_{x,y\in W} \left(u^2 I_x^2 + 2uvI_x I_y + v^2 I_y^2\right)$$

- When expressed in matrix form, we have the following equivalent expression

$$\varepsilon(u,v) \approx [u,v] \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$\varepsilon(u,v) \approx \mathbf{d}^T \mathbf{M} \mathbf{d}$$

- Measure of corner response (where λ1, λ2 are eigenvalues of M)

$$R = \det \mathbf{M} - k(trace\mathbf{M})^2$$

$$\det \mathbf{M} = \lambda_1 \lambda_2$$

Empirically determined constant

$$trace\mathbf{M} = \lambda_1 + \lambda_2$$

$$k \in [0.04, 0.06]$$

Algorithm Design – Harris Corner Detection

- Compute x and y derivatives of image (can convolve image with derivatives of Gaussians);
- Compute products of derivatives at every pixel;
- Compute the sums of the products of derivatives;
- Construct the M matrix;
- Calculate the corner response value R;
- Threshold on value R, suppress non-maximum value.

$$I_x = G_x(x, y, \sigma) * f(x, y)$$
$$I_y = G_y(x, y, \sigma) * f(x, y)$$

$$\mathbf{M} = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$

Empirically determined constant

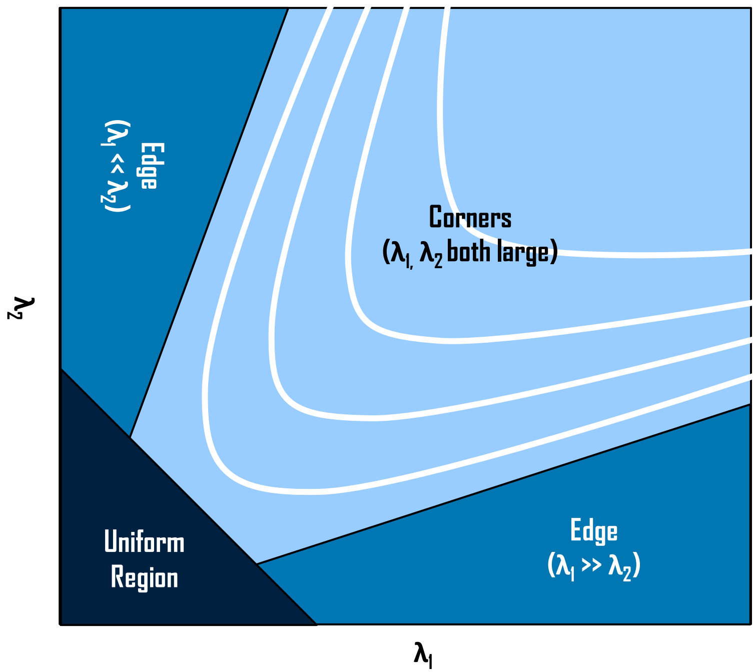
$$R = \det \mathbf{M} - k(trace\mathbf{M})^2$$

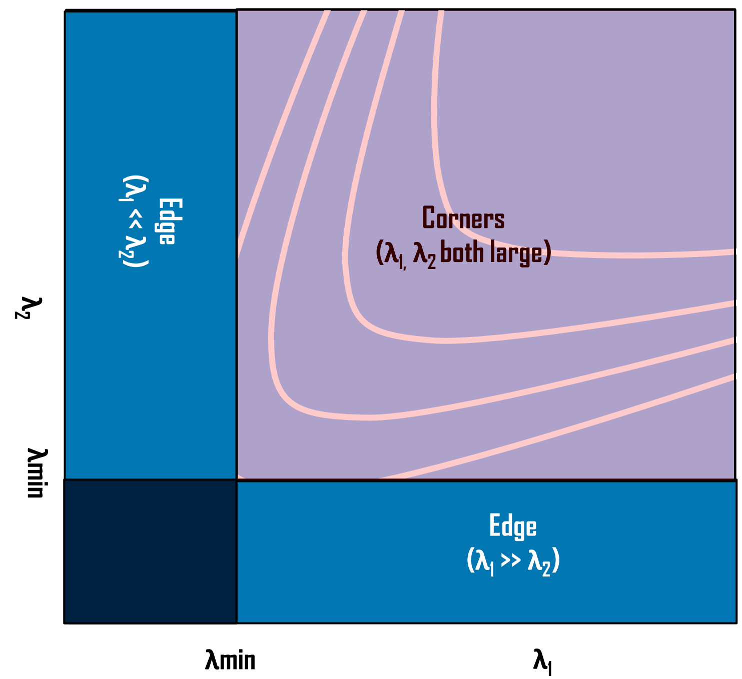
$$\det \mathbf{M} = \lambda_1 \lambda_2$$

$$trace\mathbf{M} = \lambda_1 + \lambda_2$$

Harris Corner Detector in Action

Shi-Tomasi Corner Detector





Empirically determined constant

$$R = \det \mathbf{M} - k(trace\mathbf{M})^2$$

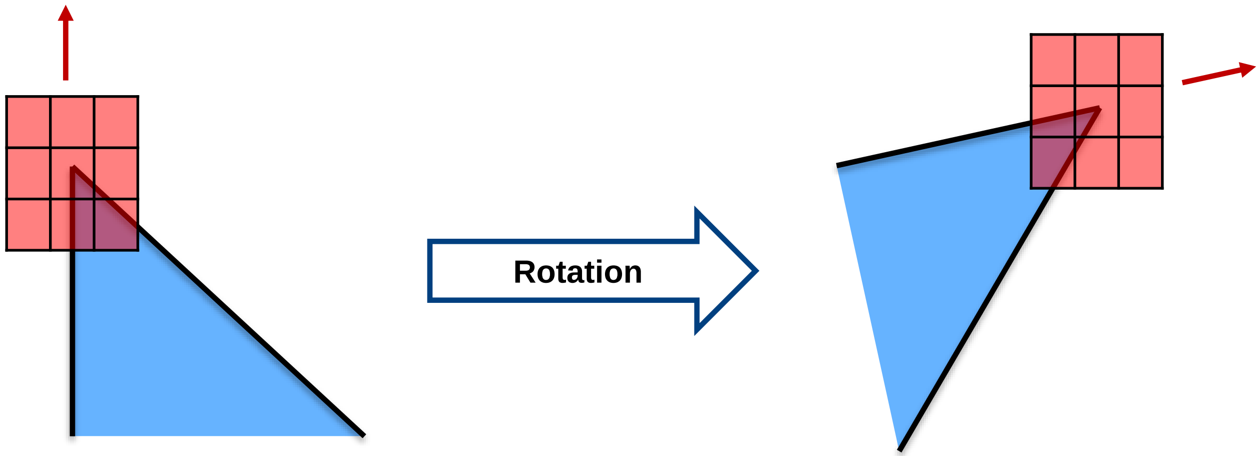
$$\det \mathbf{M} = \lambda_1 \lambda_2$$

$$trace\mathbf{M} = \lambda_1 + \lambda_2$$

$$R = \min(\lambda_1, \lambda_2)$$

Corner Detector Invariance Properties

- If the corner has rotated, you can still get the same change of intensities when you shift the window. However, you need to shift along a different direction.
- The eigenvalues of M do not change. But the eigenvectors change.
- Therefore, the ellipse representing the feature will rotate but its shape will not change.



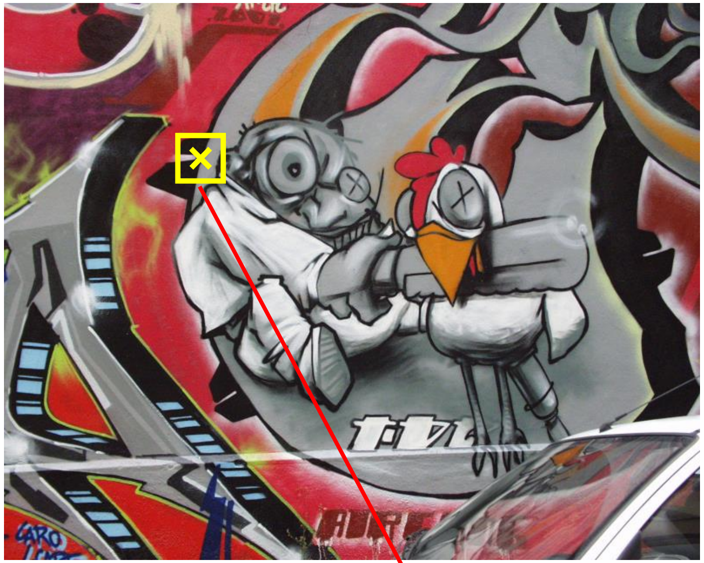
Corner Detector Invariance Properties

- When the image scale changes, the corner response will change.
- However, we can apply a corner detector at multiple scales and find the response at the most suitable scale.
- How do we normally get multi-scale images?
  - Gaussian smoothing with different σ
  - Sampling at different pixel resolutions

$\sigma = 1$ $\sigma = 3$ $\sigma = 5$ $\sigma = 7$

Image convolved with Gaussian kernels of different σ provide information at different scale.

Automatic Scale Selection



$$f(I_{i_1 \ldots i_m}(x,\sigma)) = f(I_{i_1 \ldots i_m}(x',\sigma'))$$

- We need to find the patch size for which the *f* response will be equal.

https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Response of some function *f*

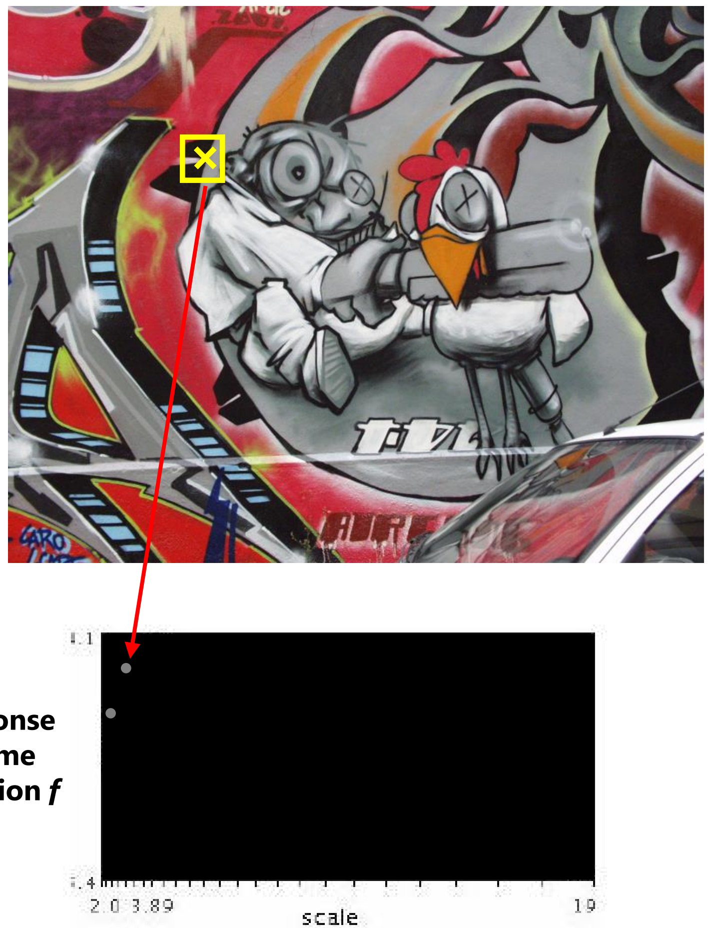
$$f(I_{i_1 \ldots i_m}(x, \sigma))$$

$$f(I_{i_1 \ldots i_m}(x', \sigma'))$$

https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

- Function responses for increasing scale (scale signature)



Response of some function *f*

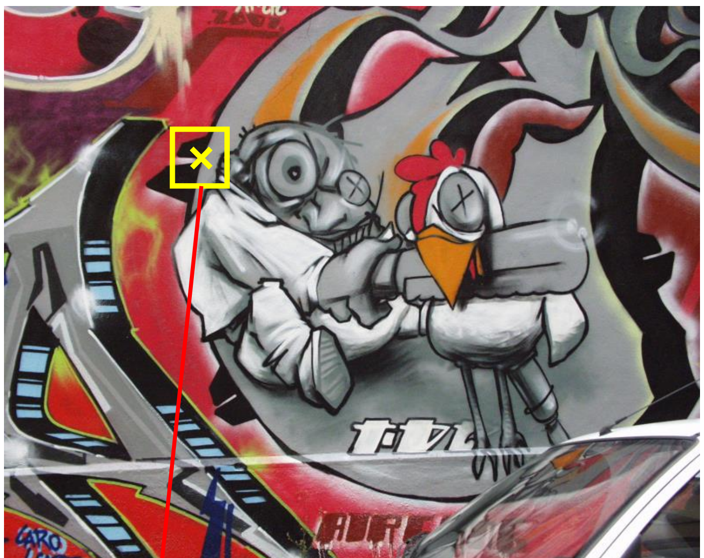
$$f(I_{i_1 \dots i_m}(x, \sigma))$$

$$f(I_{i_1 \dots i_m}(x', \sigma'))$$

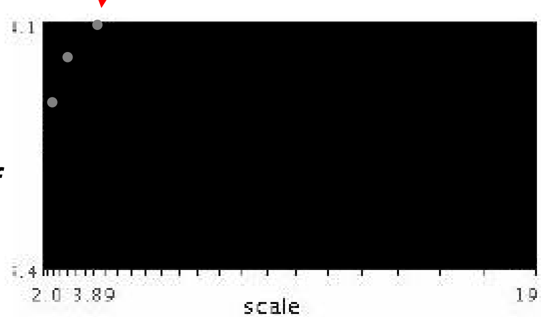
https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

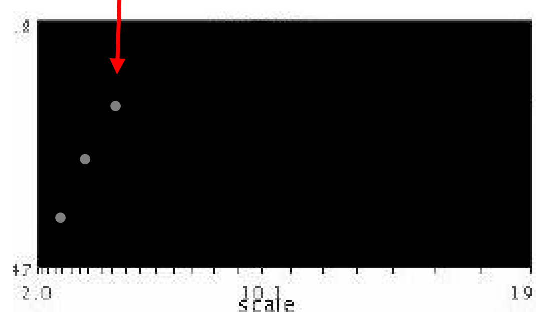
- Function responses for increasing scale (scale signature)



Response of some function *f*



$$f(I_{i_1 \ldots i_m}(x, \sigma))$$



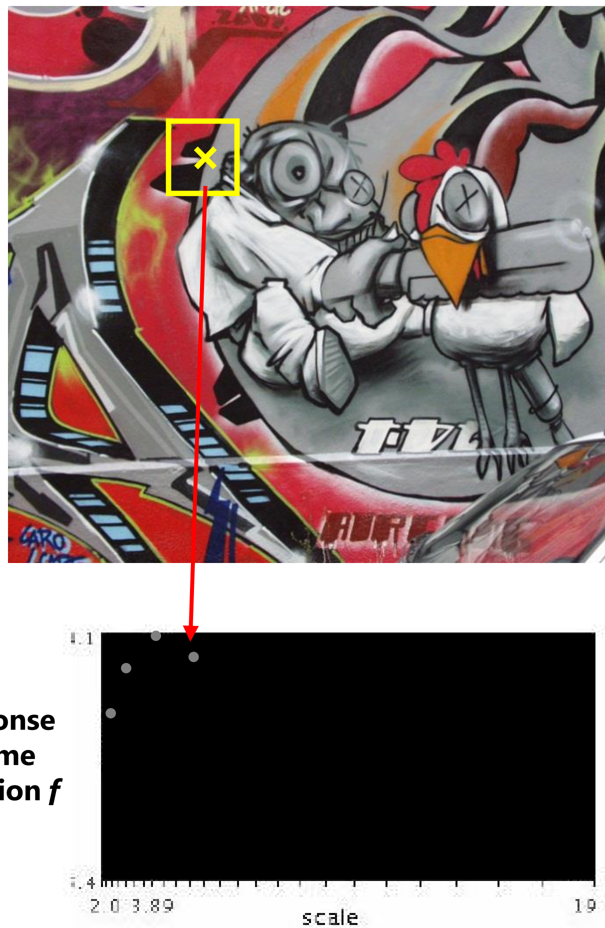
$$f(I_{i_1 \ldots i_m}(x', \sigma'))$$

https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Response of some function *f*



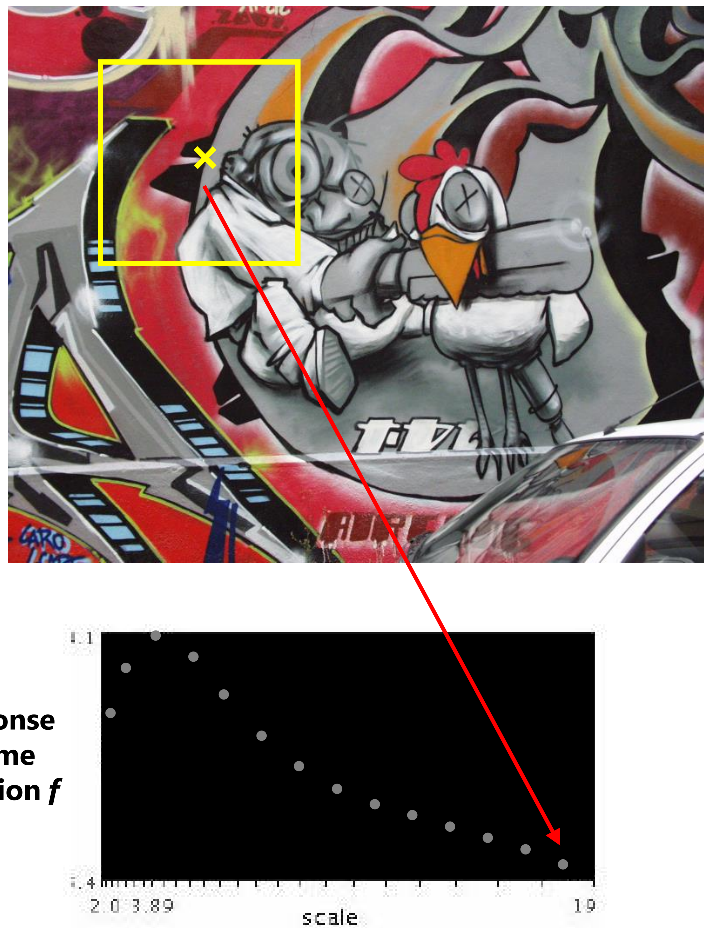
$$f(I_{i_1 \dots i_m}(x, \sigma))$$

$$f(I_{i_1 \dots i_m}(x', \sigma'))$$

https://cs.brown.edu/courses/csci1430/

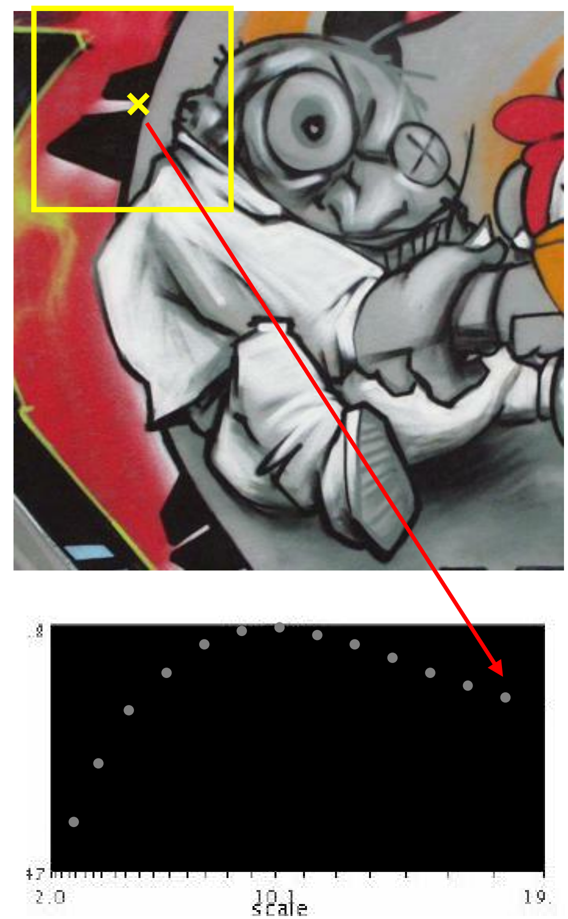
Automatic Scale Selection

- Function responses for increasing scale (scale signature)



Response of some function *f*

$$f(I_{i_1 \ldots i_m}(x, \sigma))$$



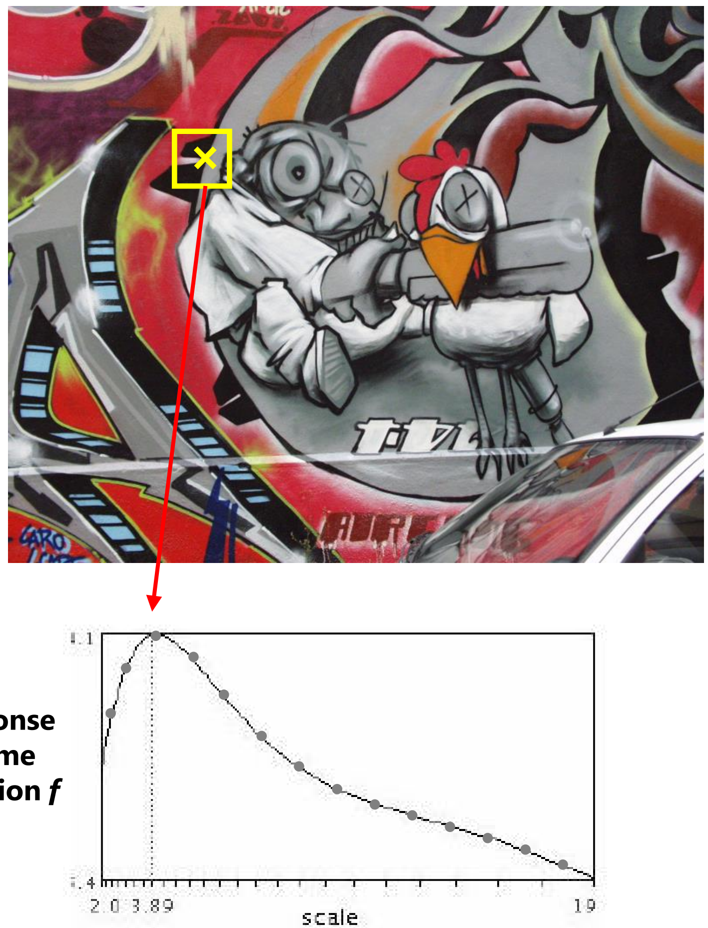
$$f(I_{i_1 \ldots i_m}(x', \sigma'))$$

https://cs.brown.edu/courses/csci1430/

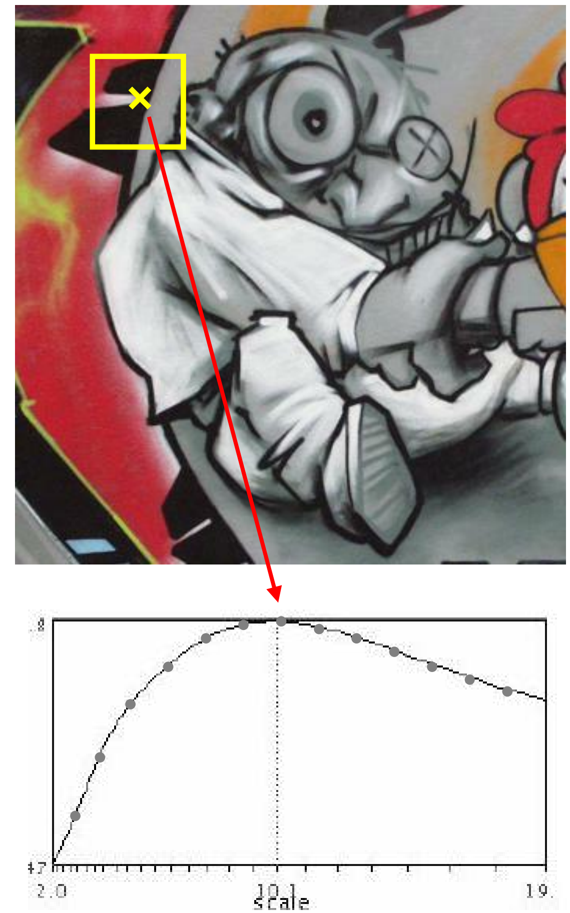
Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Response of some function *f*



$$f(I_{i_1 \ldots i_m}(x, \sigma))$$

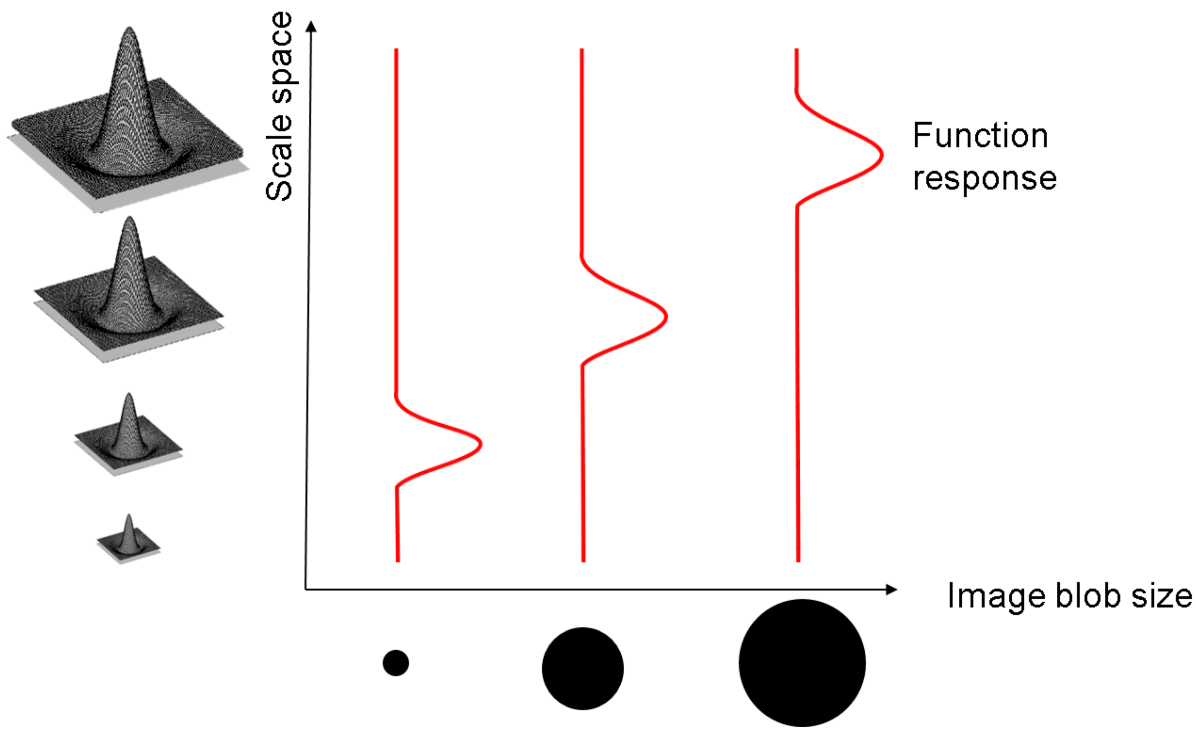


$$f(I_{i_1 \ldots i_m}(x', \sigma'))$$

https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

- The Laplacian of Gaussian (LoG) which is the second derivative of Gaussian is a suitable function to estimate the characteristic scale of image structures such as blobs and corners.
- When the size of the LoG kernel matches the size of an image structure the response attains an extremum.

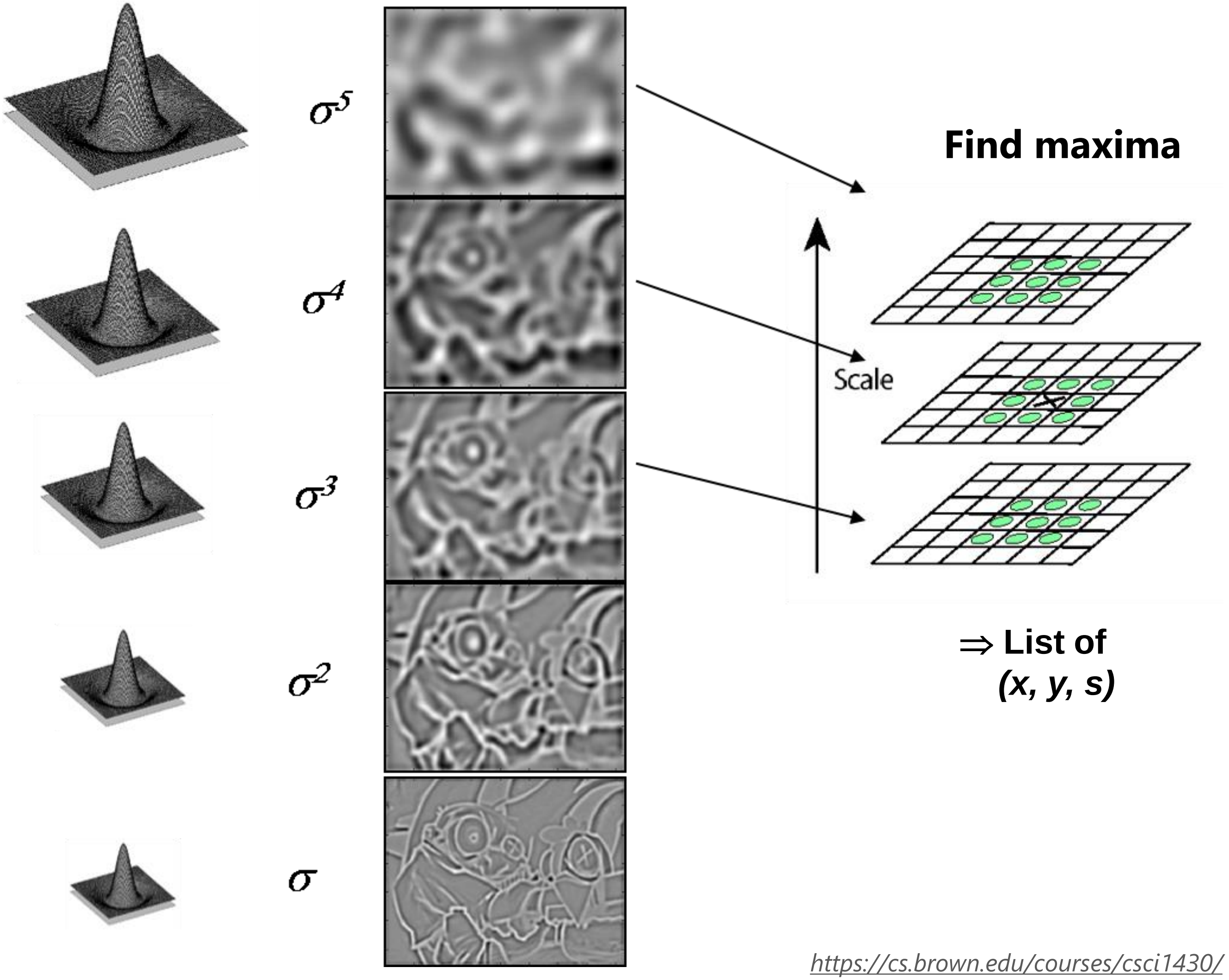


K. Mikolajczyk et al. Scale & affine invariant interest point detectors, IJCV 2004.

https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

- Find local maxima in position-scale space of LoG



$\Rightarrow$ **List of** ***(x, y, s)***

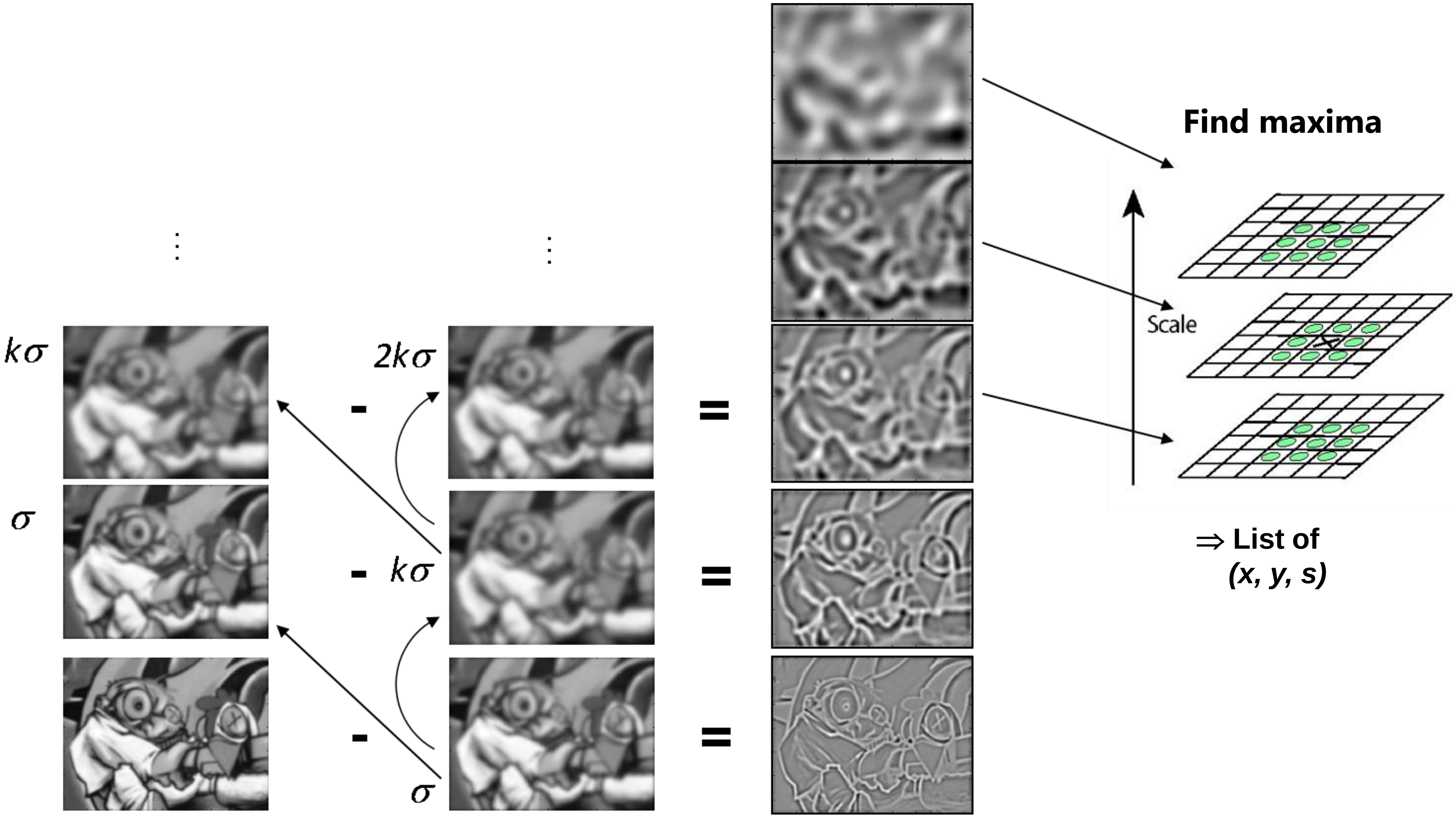
https://cs.brown.edu/courses/csci1430/

Automatic Scale Selection

Approximate LoG with Difference-of-Gaussian (DoG).

1. Blur image with σ Gaussian kernel
2. Blur image with *k* σ Gaussian kernel
3. Subtract 2. from 1.

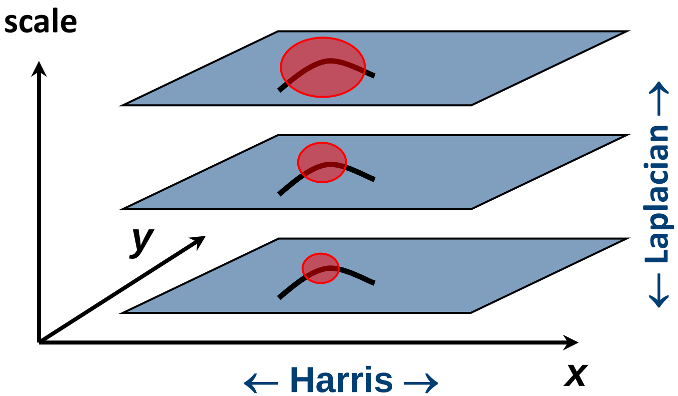
Automatic Scale Selection



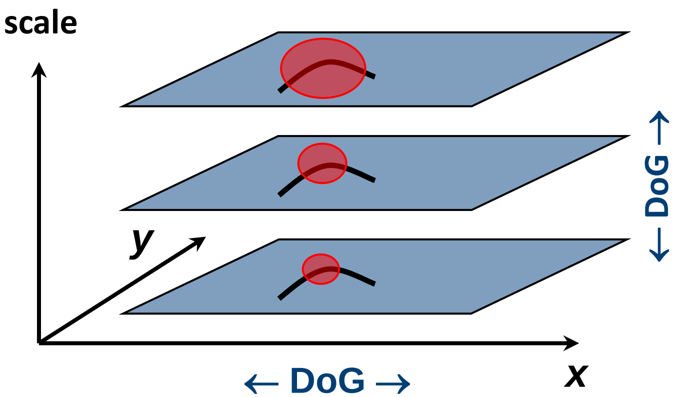
https://cs.brown.edu/courses/csci1430/

Scale invariant detectors

- Harris-Laplace detector [1]
  - Find the optimal scale σ using Laplacian response
  - Find the local maximum in space using Harris detector response



- SIFT [2]
  - Find the optimal scale σ using DoG response
  - Find the local extremum in space using DoG response



[1] K. Mikolajczyk et al. Scale & affine invariant interest point detectors, IJCV 2004.
[2] D. Lowe. Distinctive image features from scale-invariant keypoints, IJCV 2004.

Comparison of Keypoint Detectors

Table 7.1 Overview of feature detectors.

Feature Detector	Corner	Blob	Region	Rotation invariant	Scale invariant	Affine invariant	Repeatability	Localization accuracy	Robustness	Efficiency
Harris	√			√			+++	+++	+++	++
Hessian		√		√			++	++	++	+
SUSAN	√			√			++	++	++	+++
Harris-Laplace	√	(√)		√	√		+++	+++	++	+
Hessian-Laplace	(√)	√		√	√		+++	+++	+++	+
DoG	(√)	√		√	√		++	++	++	++
SURF	(√)	√		√	√		++	++	++	+++
Harris-Affine	√	(√)		√	√	√	+++	+++	++	++
Hessian-Affine	(√)	√		√	√	√	+++	+++	+++	++
Salient Regions	(√)	√		√	√	(√)	+	+	++	+
Edge-based	√			√	√	√	+++	+++	+	+
MSER			√	√	√	√	+++	+++	++	+++
Intensity-based			√	√	√	√	++	++	++	++
Superpixels			√	√	(√)	(√)	+	+	+	+

Tuytelaars, T and Mikolajczyk, K (2008) Local invariant feature detectors: A survey Foundations and Trends in Computer Graphics and Vision, 3 (3). pp. 177-280.

Conclusions

- Feature Detection
- Interest Points in Vision
- Corner Detection and Harris Corner Detector
- Shi-Tomasi Corner Detector
- Automatic Scale Selection