D. Lowe, [Distinctive image features from scale-invariant keypoints](https://www.cvlibs.net/publications/IJCV60 Lowe2004.pdf), *IJCV* 60 (2), pp. 91-110, 2004

Slides from S. Lazebnik.
Keypoint detection with scale selection

- We want to extract keypoints with characteristic scales that are covariant w.r.t. the image transformation.

Source: L. Lazebnik
Basic idea

- Convolve the image with a “blob filter” at multiple scales and look for extrema of filter response in the resulting scale space

T. Lindeberg, Feature detection with automatic scale selection, IJCV 30(2), pp 77-116, 1998
Blob detection

Find maxima and minima of blob filter response in space and scale

Source: L. Lazebnik  
Source: N. Snavely
Blob filter

Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

\[ \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]

Source: L. Lazebnik
Recall: Edge detection

Edge = maximum of derivative

\[ f \]

\[ \frac{d}{dx}g \]

\[ f * \frac{d}{dx}g \]

Source: S. Seitz

Source: L. Lazebnik
Edge detection, Take 2

Edge = zero crossing of second derivative

Source: L. Lazebnik

Source: S. Seitz
From edges to blobs

- Edge = ripple
- Blob = superposition of two ripples

Spatial selection: the magnitude of the Laplacian response will achieve a maximum at the center of the blob, provided the scale of the Laplacian is “matched” to the scale of the blob.

Source: L. Lazebnik
Scale selection

• We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response.

• However, Laplacian response decays as scale increases:

Source: L. Lazebnik
Scale normalization

- The response of a derivative of Gaussian filter to a perfect step edge decreases as $\sigma$ increases:

- To keep response the same (scale-invariant), must multiply Gaussian derivative by $\sigma$
- Laplacian is the second Gaussian derivative, so it must be multiplied by $\sigma^2$

Source: L. Lazebnik
Effect of scale normalization

Original signal

Unnormalized Laplacian response

Scale-normalized Laplacian response

Source: L. Lazebnik
Blob detection in 2D

- **Scale-normalized** Laplacian of Gaussian:

\[
\nabla^2_{\text{norm}} g = \sigma^2 \left( \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right)
\]

Source: L. Lazebnik
Blob detection in 2D

- At what scale does the Laplacian achieve a maximum response to a binary circle of radius $r$?

Source: L. Lazebnik
Blob detection in 2D

- At what scale does the Laplacian achieve a maximum response to a binary circle of radius \( r \)?
- To get maximum response, the zeros of the Laplacian have to be aligned with the circle.
- The Laplacian is given by (up to scale):
  \[
  (x^2 + y^2 - 2\sigma^2) e^{-(x^2+y^2)/2\sigma^2}
  \]
- Therefore, the maximum response occurs at \( \sigma = r / \sqrt{2} \).

Source: L. Lazebnik
Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales

Source: L. Lazebnik
Scale-space blob detector: Example

Source: L. Lazebnik
Scale-space blob detector: Example

Source: L. Lazebnik
Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales
2. Find maxima of squared Laplacian response in scale-space

Source: L. Lazebnik
Scale-space blob detector: Example

Source: L. Lazebnik
Efficient implementation

- Approximating the Laplacian with a difference of Gaussians:

\[
L = \sigma^2 \left( G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)
\]

(Laplacian)

\[
DoG = G(x, y, k\sigma) - G(x, y, \sigma)
\]

(Difference of Gaussians)

Source: L. Lazebnik
Efficient implementation


Source: L. Lazebnik
Eliminating edge responses

- Laplacian has strong response along edges

Source: L. Lazebnik
Eliminating edge responses

• Laplacian has strong response along edges

• Solution: filter based on Harris response function over neighborhoods containing the “blobs”

Source: L. Lazebnik
From feature detection to feature description

• To recognize the same pattern in multiple images, we need to match appearance “signatures” in the neighborhoods of extracted keypoints
  • But corresponding neighborhoods can be related by a scale change or rotation
  • We want to *normalize* neighborhoods to make signatures invariant to these transformations

Source: L. Lazebnik
Finding a reference orientation

- Create histogram of local gradient directions in the patch
- Assign reference orientation at peak of smoothed histogram

Source: L. Lazebnik
SIFT features

- Detected features with characteristic scales and orientations:

![Image of SIFT features]


Source: L. Lazebnik
From keypoint detection to feature description

Detection is covariant:

\[ \text{features}(\text{transform(image)}) = \text{transform(features(image))} \]

Description is invariant:

\[ \text{features}(\text{transform(image)}) = \text{features(image)} \]

Source: L. Lazebnik
SIFT descriptors

- Inspiration: complex neurons in the primary visual cortex

D. Lowe, Distinctive image features from scale-invariant keypoints, *IJCV* 60 (2), pp. 91-110, 2004

Source: L. Lazebnik
Properties of SIFT

Extraordinarily robust detection and description technique

• Can handle changes in viewpoint
  – Up to about 60 degree out-of-plane rotation
• Can handle significant changes in illumination
  – Sometimes even day vs. night
• Fast and efficient—can run in real time
• Lots of code available