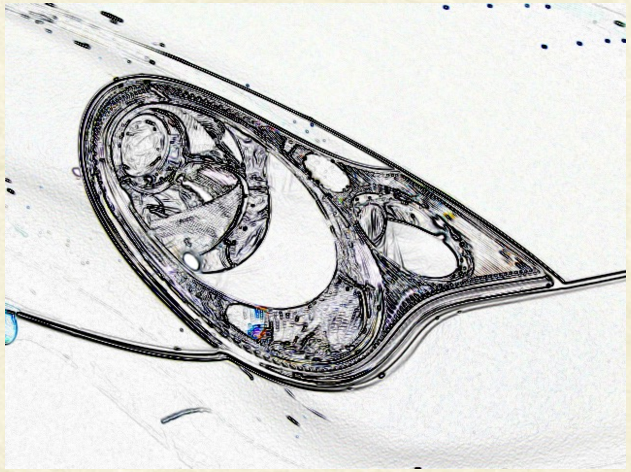
CS7.505: Computer Vision

Spring 2022: SURF Detector and Descriptor


Anoop M. Namboodiri

Biometrics and Secure ID Lab, CVIT,

IIIT Hyderabad

Key Ideas: Recap

- Hessian for Interest
  - Laplacian for Interest: Harris Corners
- DoG Approximation of LoG
  - SIFT
- Integral Image for fast box filters
  - Viola and Jones
- Scale Space Detector
  - SIFT, HoG, etc.
- Gradient Histogram from Oriented Windows
  - SIFT

Hessian Matrix:

$$H(f(x,y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial y \partial x} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

Hessian at scale σ: Second derivative of Gaussian at scale σ:

$$H(x,\sigma) = \begin{bmatrix} L_{\mathrm{xx}}(x,\sigma) & L_{\mathrm{yx}}(x,\sigma) \\ L_{\mathrm{xy}}(x,\sigma) & L_{\mathrm{yy}}(x,\sigma) \end{bmatrix}$$

Laplace operator at scale σ:

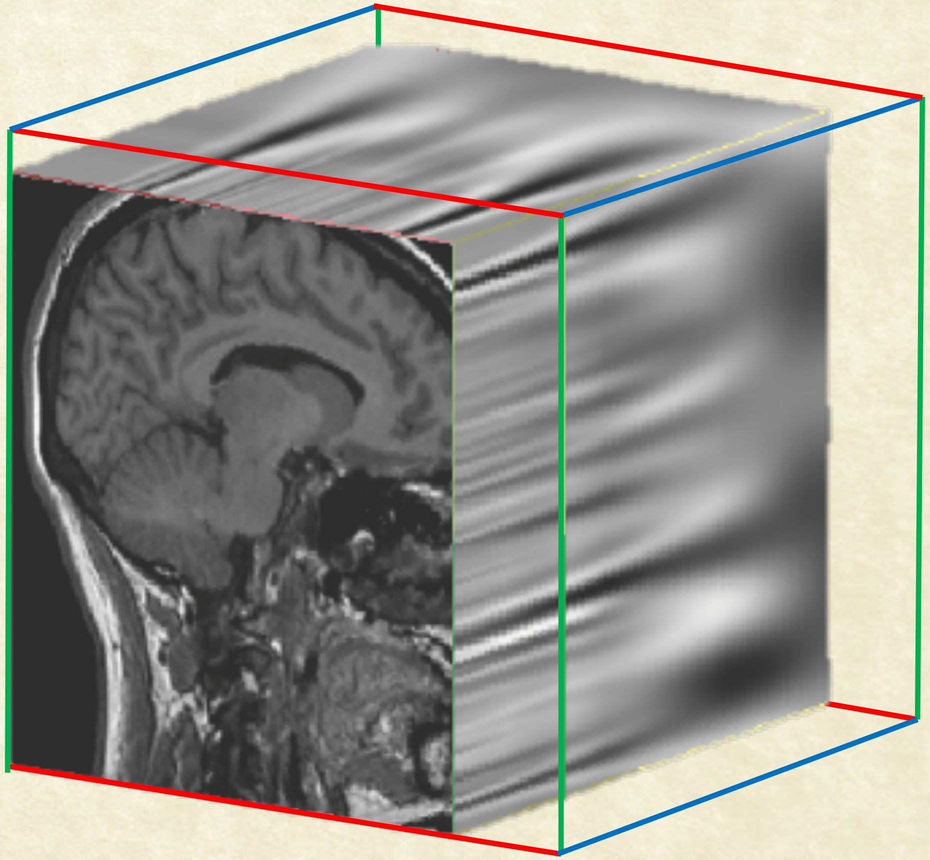
Trace of Hessian at scale σ.

Interest Operator

- Find points in the image that are:
  1. Distinctive (corners, blobs, junctions)
  2. Repeatable to find
     - Invariant to Scale, Geometric and Photometric transformations
  3. Fast to find
- What about Harris Corners?
  - Based on Gradient covariance matrix.
  - Hessian-based detectors are more robust.
  - Not scale invariant

Scale Space

- Successively smooth the image using a Gaussian with larger σ.
- Forms a 3D space with scale as the third axis.
- Can also be seen as a pyramid (3rd axis has decreasing resolution)

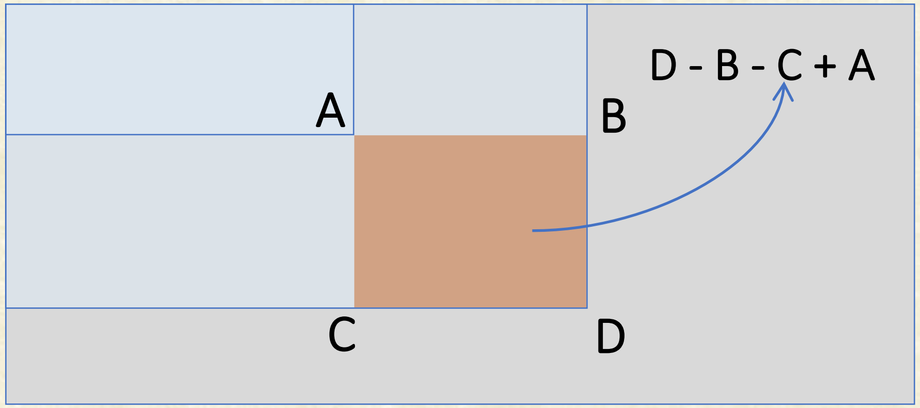


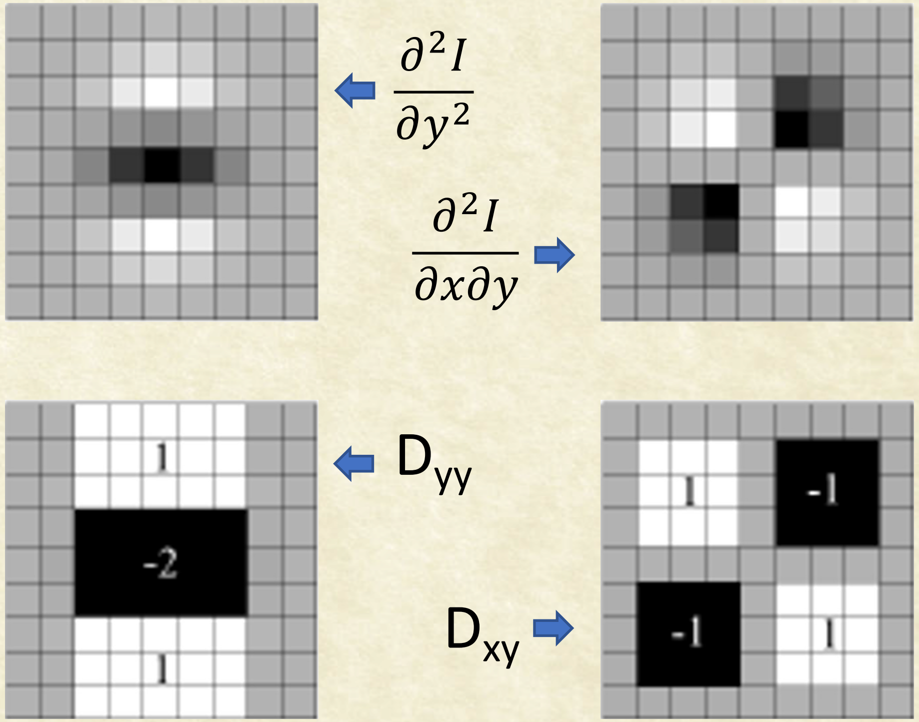
ftp://ftp.nada.kth.se/CVAP/reports/Lin08-EncCompSci.pdf

SURF Detector

4-step process

1. Compute Integral Image
2. Apply 2nd derivative (approximate) filters to image
3. Non-maximal suppression (Find local maxima in (x,y,σ) space)
4. Quadratic interpolation

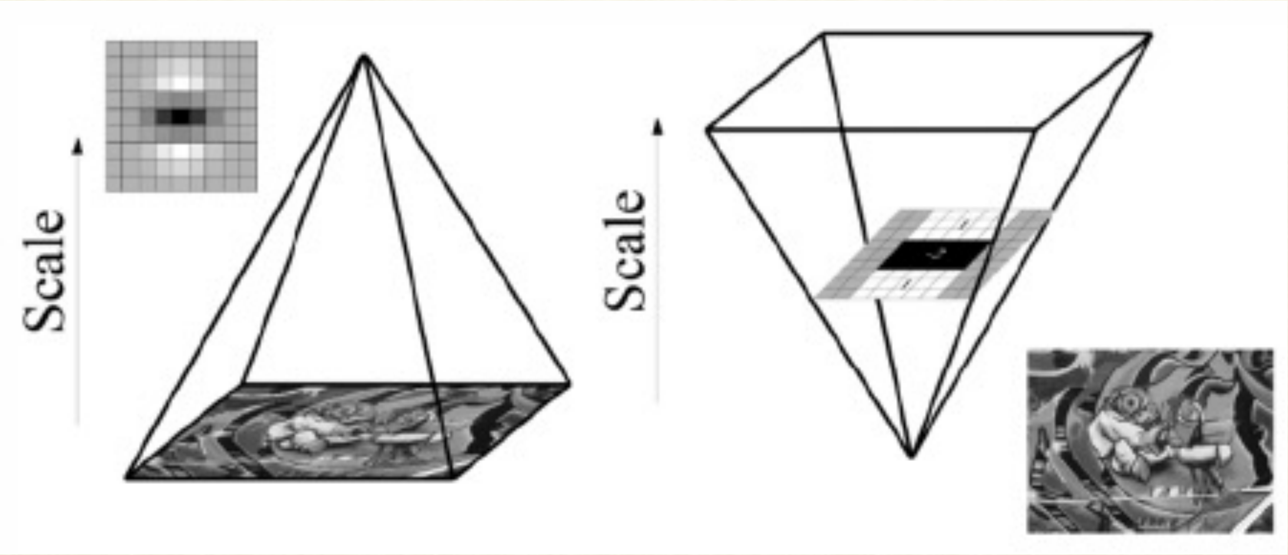


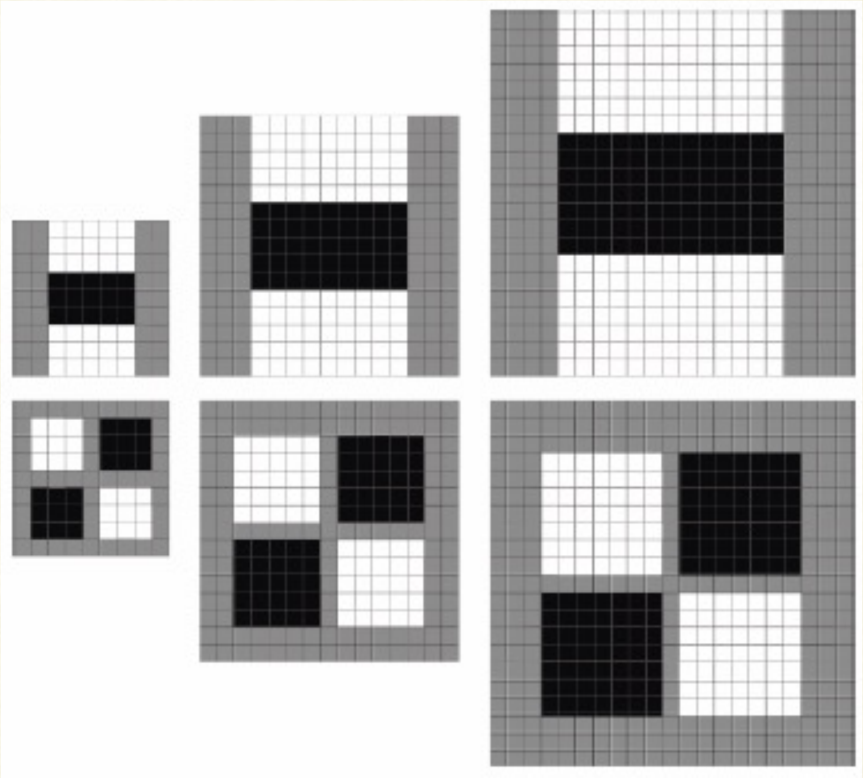


$\det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2$

Step 3: Scale Space and NMS

- Instead of downsampling the image, we scale up the filters
- Using integral image makes the computations equally efficient

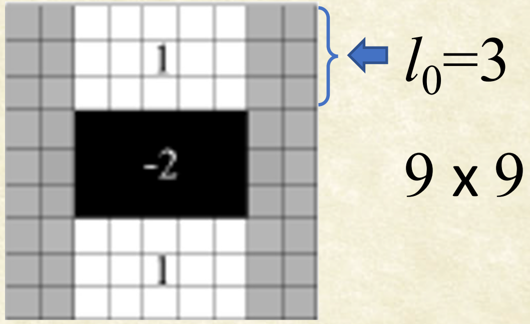


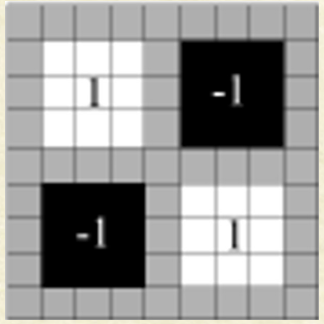


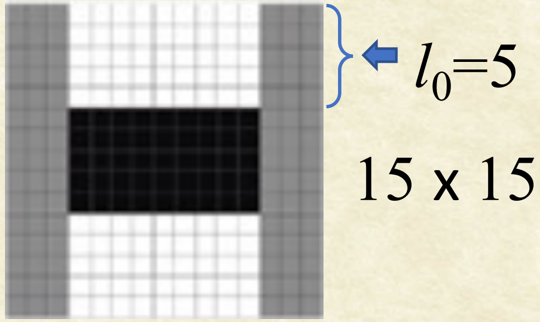
Step 3: Scale Space and NMS

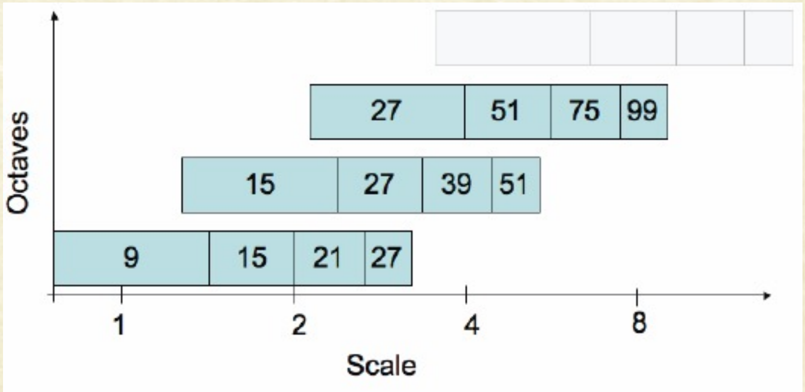
Selection of Scale

- Within an octave, select sizes that preserves central pixel
  - Let l_0 increase by 2
  - 9x9, 15x15, 21x21, 27x27
- Doubling of scale (octave) preserves discretization
- Filter Sizes:
  - 9, 15, 21, 27
  - 15, 27, 39, 51
  - 27, 51, 75, 99
- Scale is selected to match filter sizes in next octave

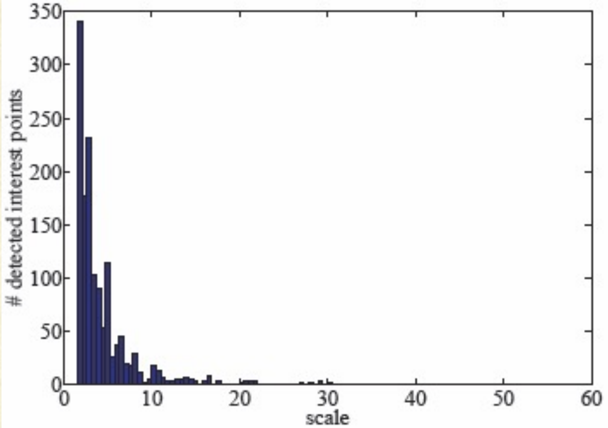








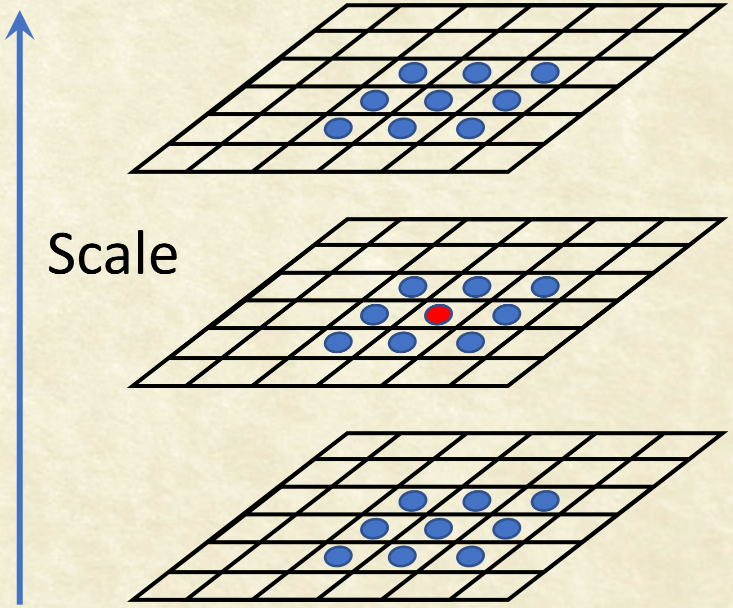
Step 3: Scale Space and NMS



- Note: Higher octaves have fewer interest points.

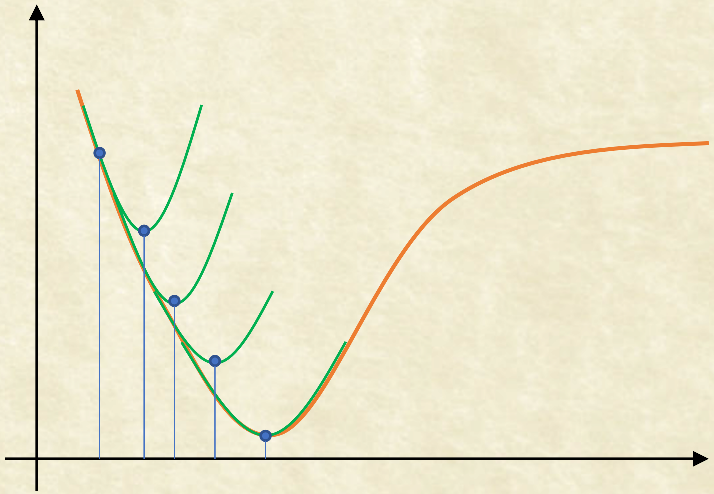
Non-Maximal Suppression

- Retain pixel only if greater than 26 neighbors in x, y, σ.



Step 4: Interpolation

- For each local maximum, the true location is affected by the discretization (especially at higher scales)
- Solution: Interpolate using known values to get true location in the 3D image-scale space
  - Take Taylor series expansion of $H(\mathbf{x}) = H + \frac{\partial H}{\partial \mathrm{x}}^T \mathrm{x} + \frac{1}{2} \mathrm{x}^T \frac{\partial^2 H}{\partial \mathrm{x}} \mathrm{x}$
  - Solve using Newton's method:
    - Done by Successive Parabolic local approximation
    - Each time, find the min/max of the parabola
- Note: Needs to be done only for few points.



SURF Descriptor

3-Step Process

1. Divide window into 4x4 (16 sub-windows)
2. Compute Haar wavelet outputs
3. Within each sub-window, compute:

$$v_{subregion} = \left[\sum dx\,, \sum dy, \sum |dx|, \sum |dy|\right]$$

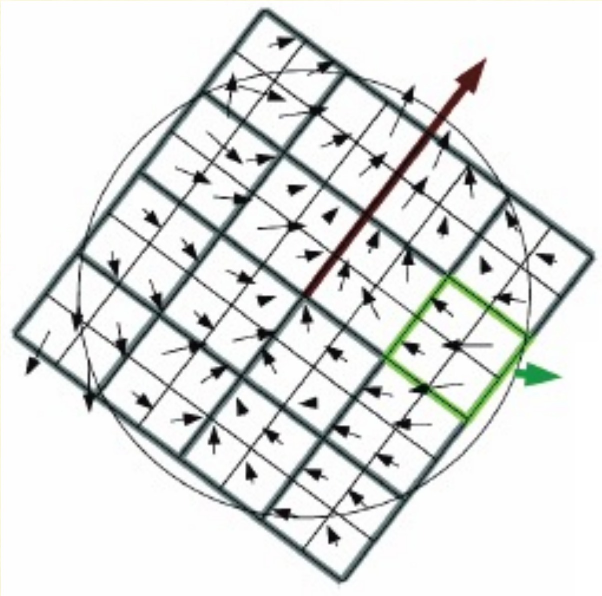
- This yields a 64-element descriptor

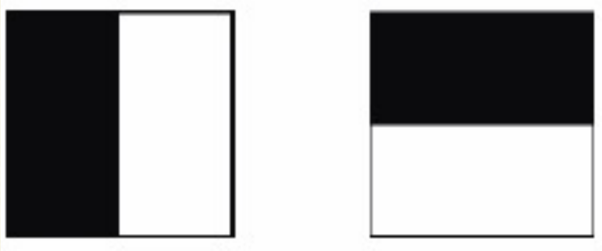
SURF Descriptor

Once an interest point is located,

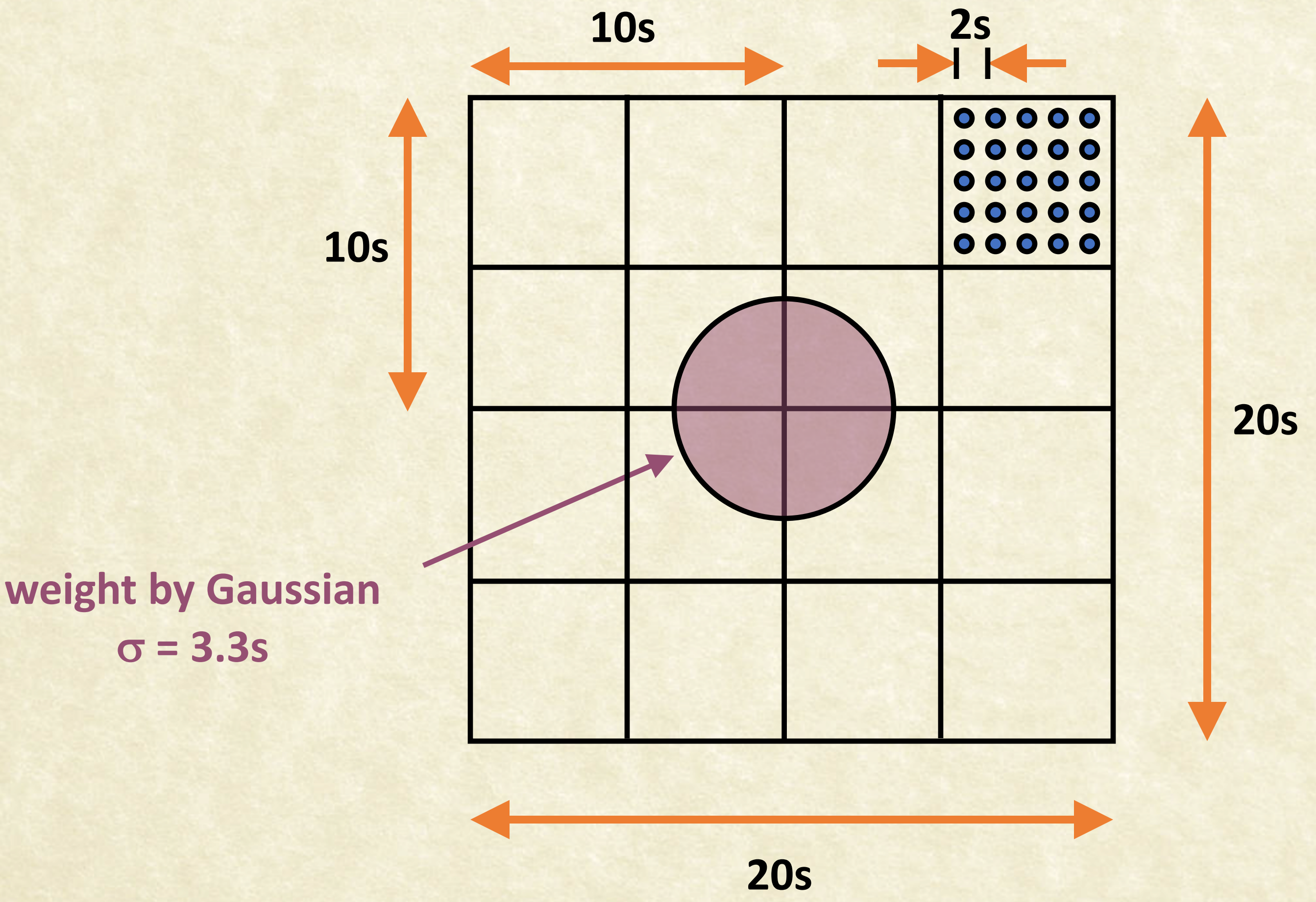
- Place a window at the point
- Divide into 4x4 sub-windows, with each sub-window having 5x5 locations
- Compute dx and dy at each location
  - Use Haar wavelets to compute these
- Sum over all 25 locations to get:

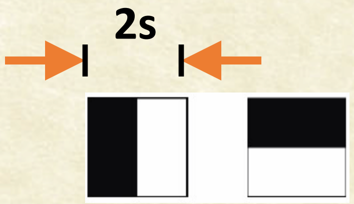
$$\sum dx, \sum dy, \sum |dx|, \sum |dy|$$



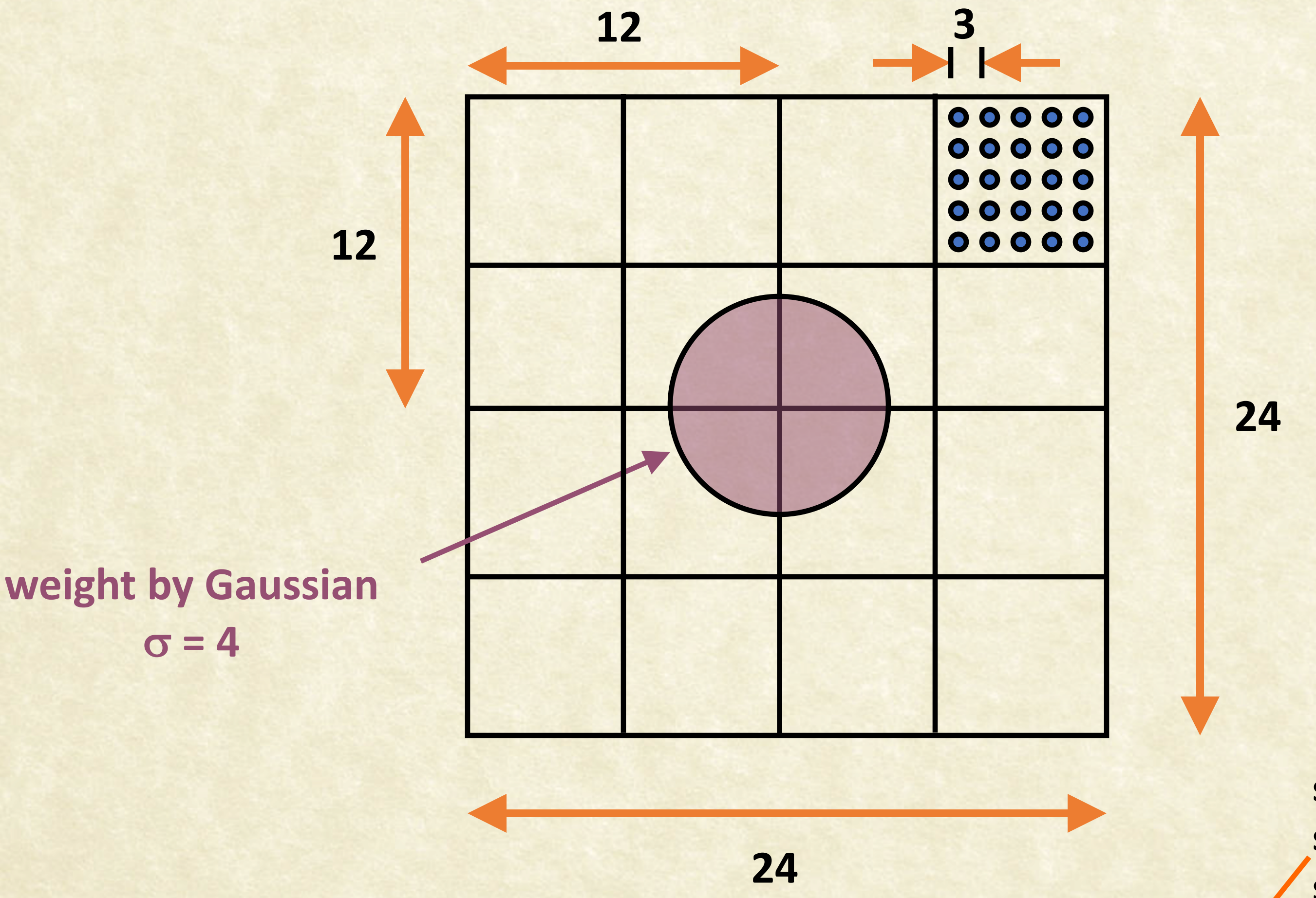


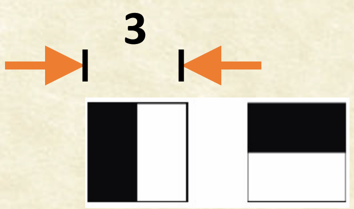
Details





with s=1.2



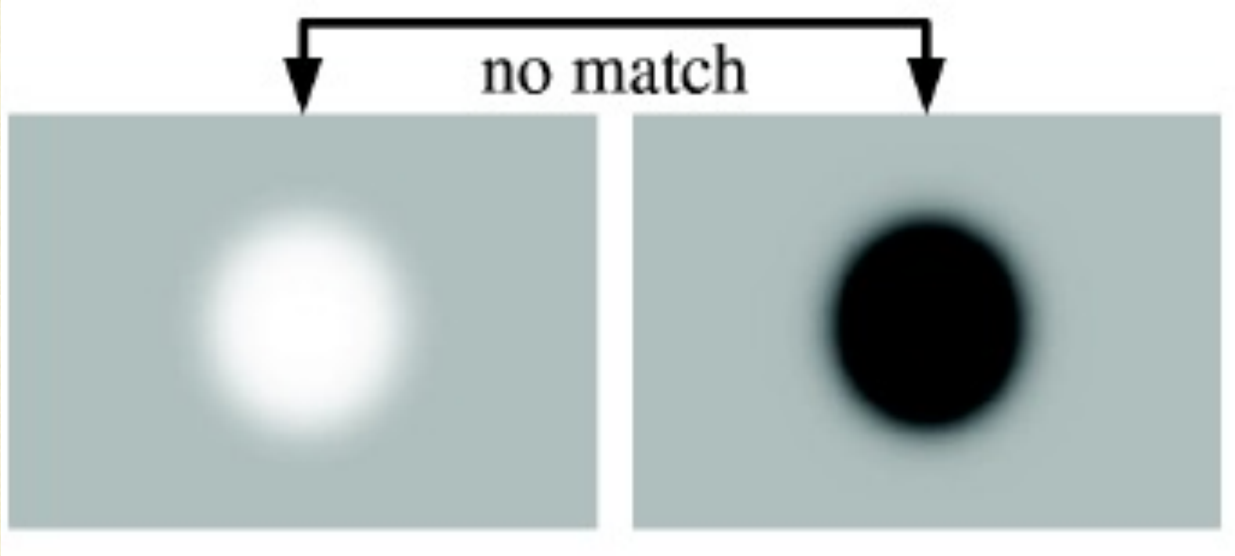


sign of LoG =
sign of trace(H_{approx}) =
sgn(Dxx + Dyy)

→ 65 values per interest point (16 x 4 + 1)

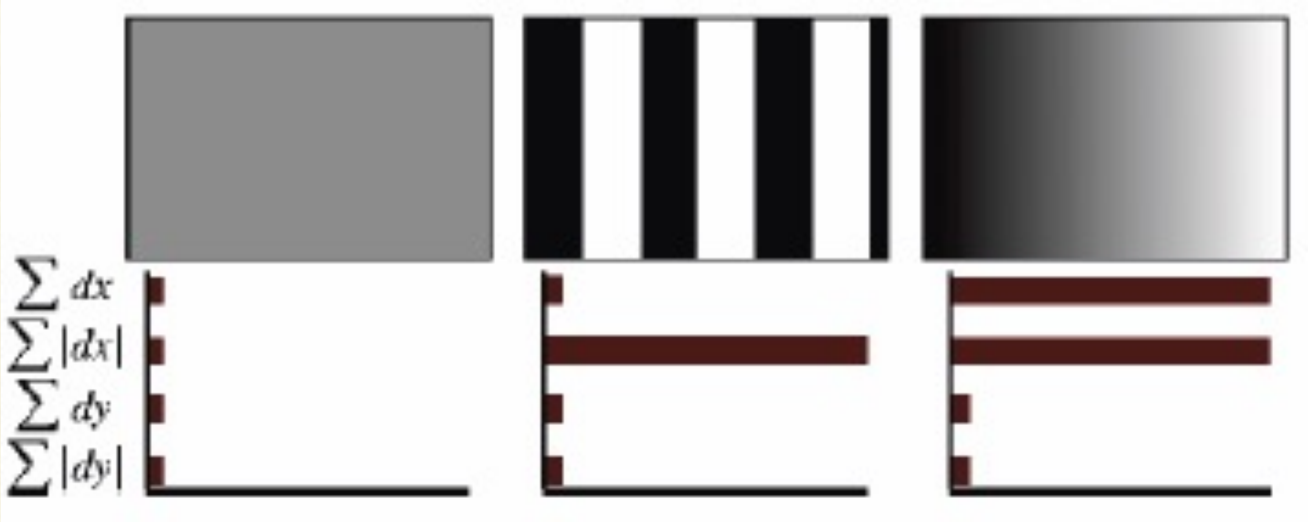
Sign of LoG

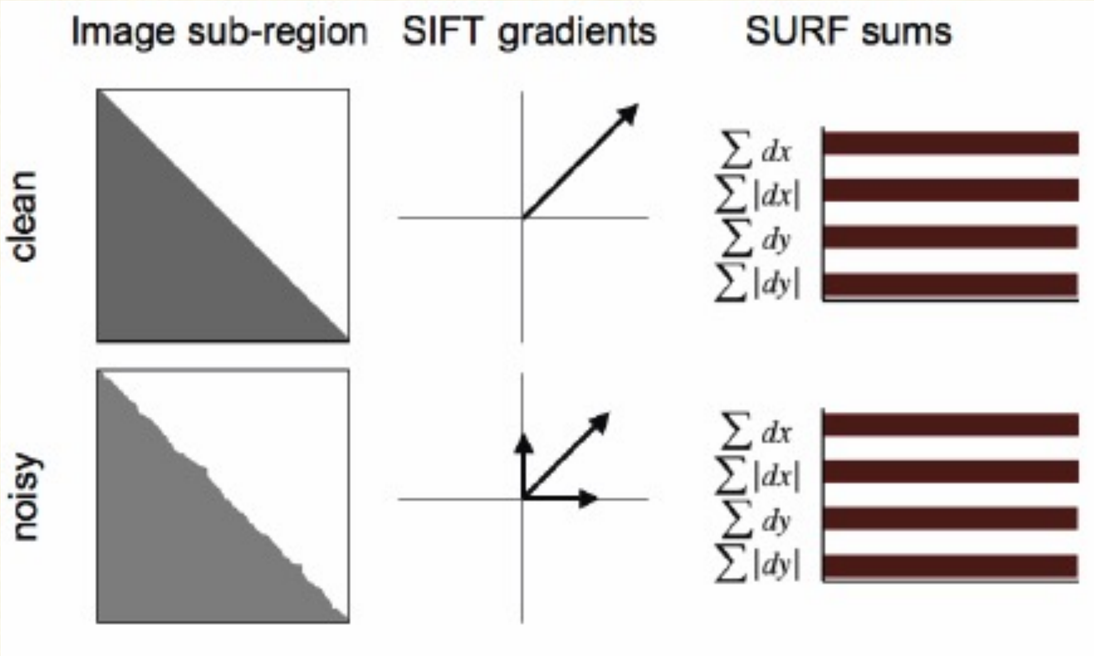
- Speeds up matching



sgn(Dxx + Dyy)

Why SURF is better than SIFT





Examples: Flowers

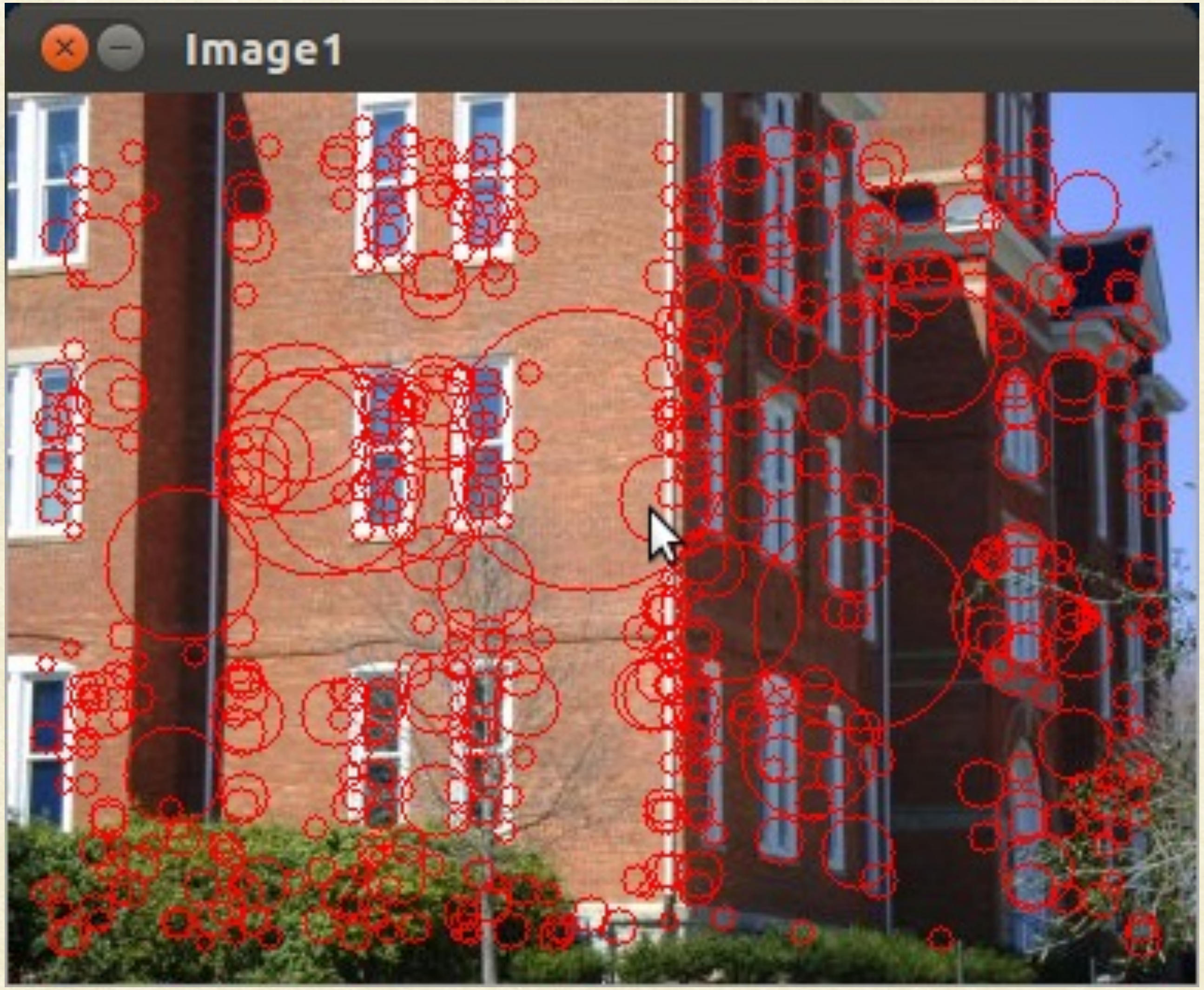
Examples: Tillman

SURF, 5 octaves, 4 scales per octave

Examples: Tillman

U-SURF, 1 octave, 4 scales per octave

Examples: Tillman



OpenCV's SURF

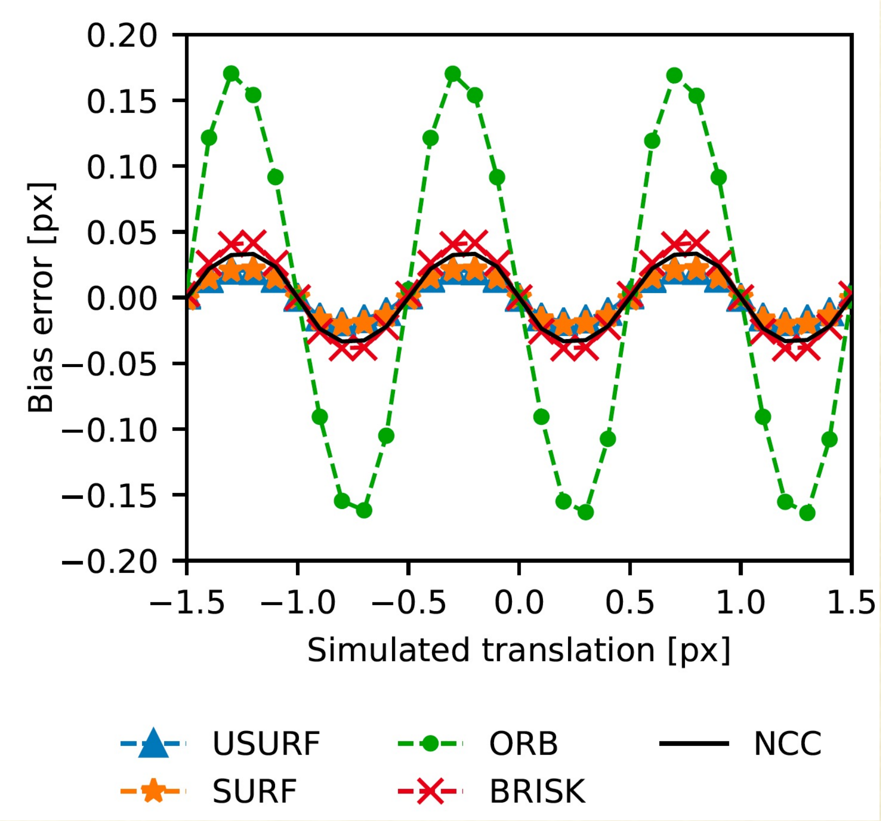
Other Alternatives

Descriptors

- Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. “ORB: An efficient alternative to SIFT or SURF”, in Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
- Leutenegger, S.; Chli, M.; Siegwart, R.Y., “BRISK: Binary Robust Invariant Scalable Keypoints”, in Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2548–2555.

Detectors

- Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. “BRIEF: Binary Robust Independent Elementary Features”, in 6314 LNCS; Springer: Berlin, Germany, 2010; pp. 778–792.
- Mair, E.; Hager, G.D.; Burschka, D.; Suppa, M.; Hirzinger, G. “Adaptive and Generic Corner Detection Based on the Accelerated Segment Test”, in Proceedings of the European Conference on Computer Vision (ECCV’10); Crete, Greece, 5–11 Sept. 2010; pp. 183–196.



Accuracy of tracking: Simulated translation

Resources

- SURF homepage
  http://www.vision.ee.ethz.ch/~surf/

- H. Bay, T. Tuytelaars, and L. V. Gool, SURF: Speeded Up Robust Features, ECCV 2006
  http://www.vision.ee.ethz.ch/~surf/eccv06.pdf

- H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, SURF: Speeded Up Robust Features (SURF), Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346–359, 2008
  ftp://ftp.vision.ee.ethz.ch/publications/articles/eth_biwi_00517.pdf

- C. Evans, OpenSURF library
  http://www.chrisevansdev.com/computer-vision-opensurf.html

Questions?