UNIVERSITÀ DEGLI STUDI DI PADOVA

An overview of existing features

Stefano Ghidoni

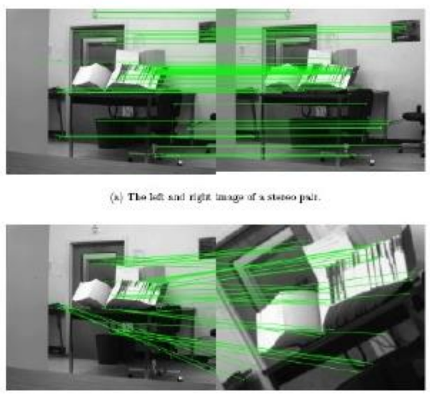
Agenda

- SIFT-based features
- Features for analyzing shapes
- Binary features
- Feature comparison

UNIVERSITÀ DEGLI STUDI DI PADOVA

Beyond SIFT

IAS-LAB



(a) The left and right image of a stereo pair.

(b) An image rotated 30°, cropped and scaled by a factor 2 in respect to the original image on the left side.

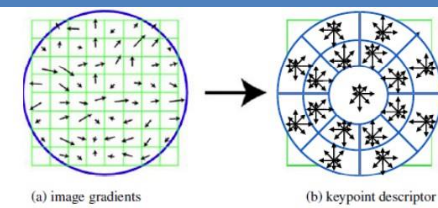


Figure 4.19 The gradient location-orientation histogram (GLOH) descriptor uses log-polar bins instead of square bins to compute orientation histograms (Mikolajczyk and Schmid 2005).

PCA-SIFT	SURF	GLOH

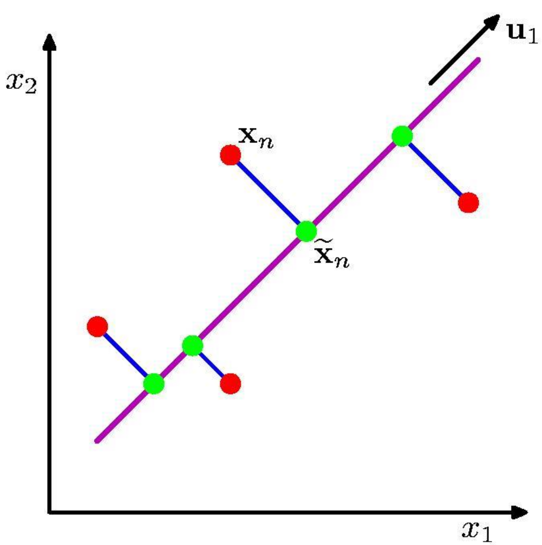
Principal Component Analysis (PCA)

- What is PCA?

Principal Component Analysis (PCA)

- What is PCA?
- A dimensionality reduction technique
- Reducing dimensions can be useful to
  - Work on more compact representations
  - Reduce computational workload
  - Highlight the most important information hiding the details

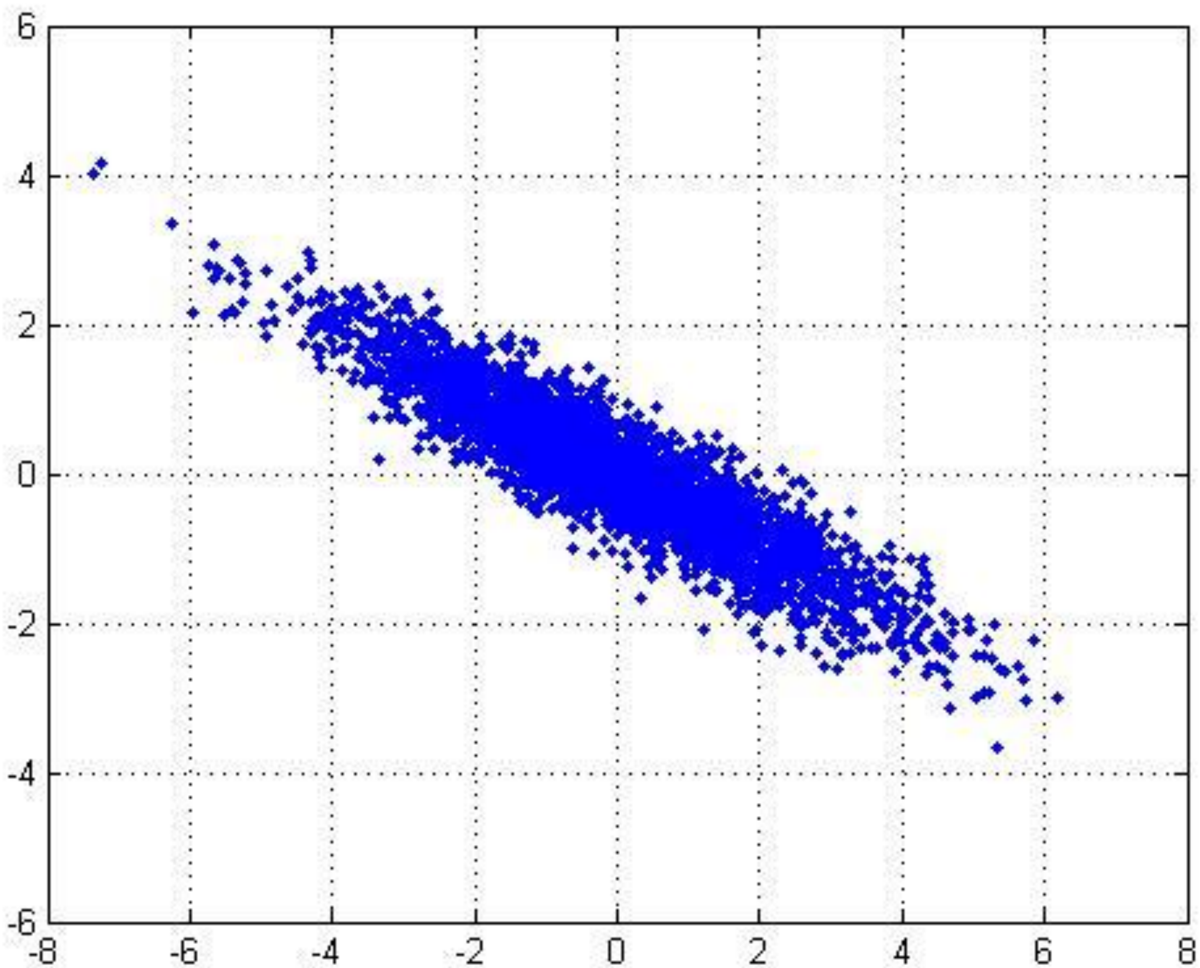
- Recall: PCA is an orthogonal projection of data onto a lower dimensional linear space that
  - Maximizes variance of projected data (purple line)
  - Minimizes mean squared distance between original data points and their projection (sum of blue lines)



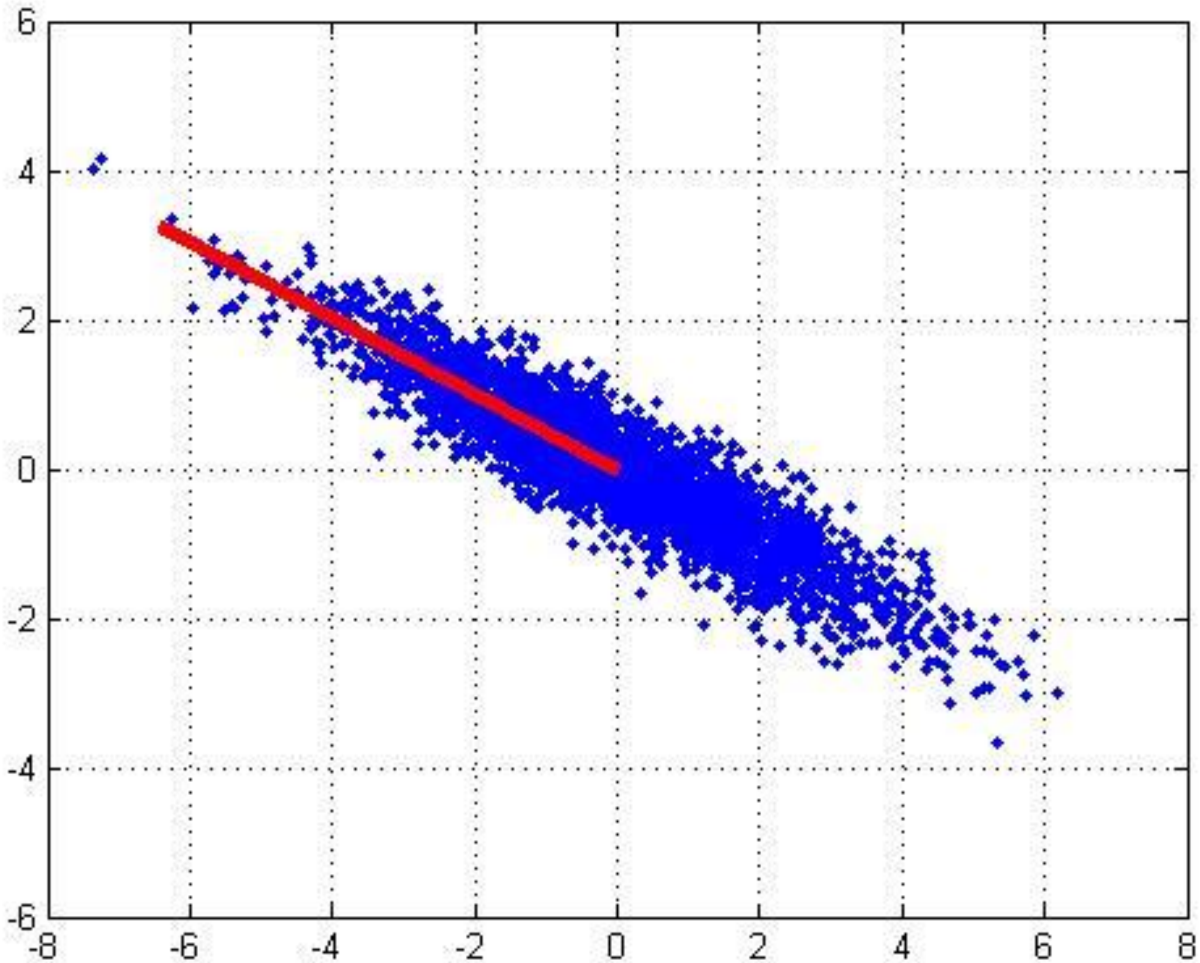
Principal Component Analysis (PCA)

- Principal component #1 points in the direction of largest variance
- Each other principal component
  - Is orthogonal to the previous ones
  - Points in the direction of largest variance of the residual subspace

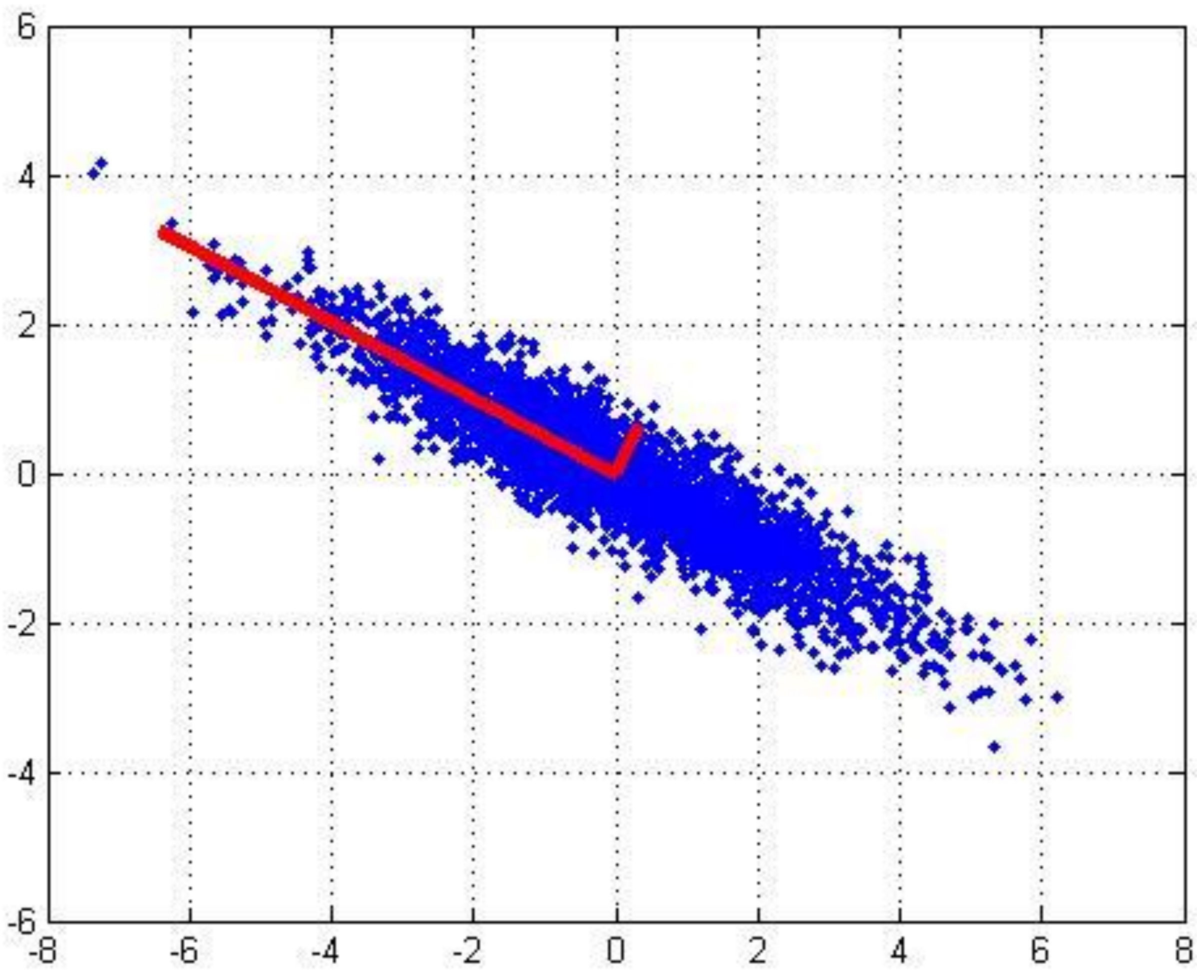
Example: data



Example: 1st axis

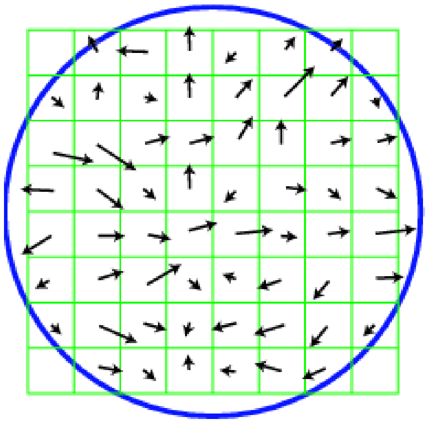


Example: 2nd axis



PCA-SIFT

- Steps 1-3: same as SIFT
- Modify step 4 (descriptor calculation)
- Refer to a 41x41 patch at the given scale, centered around the keypoint, normalized to a canonical direction



Yan Ke and Rahul Sukthankar, “PCA-SIFT: A More Distinctive Representation for Local Image Descriptors”, Computer Vision and Pattern Recognition, 2004

PCA-SIFT

- SIFT: weighted histograms
- PCA-SIFT: concatenate horizontal and vertical gradients (39x39) into a long vector
  - Length: 2 directions x 39x39 = 3042
- Normalize the vector to unit norm
- Reduce the vector using PCA
  - E.g., from 3042 to 36

SURF

SURF & the integral image

- Speeded Up Robust Features (SURF)
- Speed-up computations by fast approximation of hessian matrix and descriptors using integral images

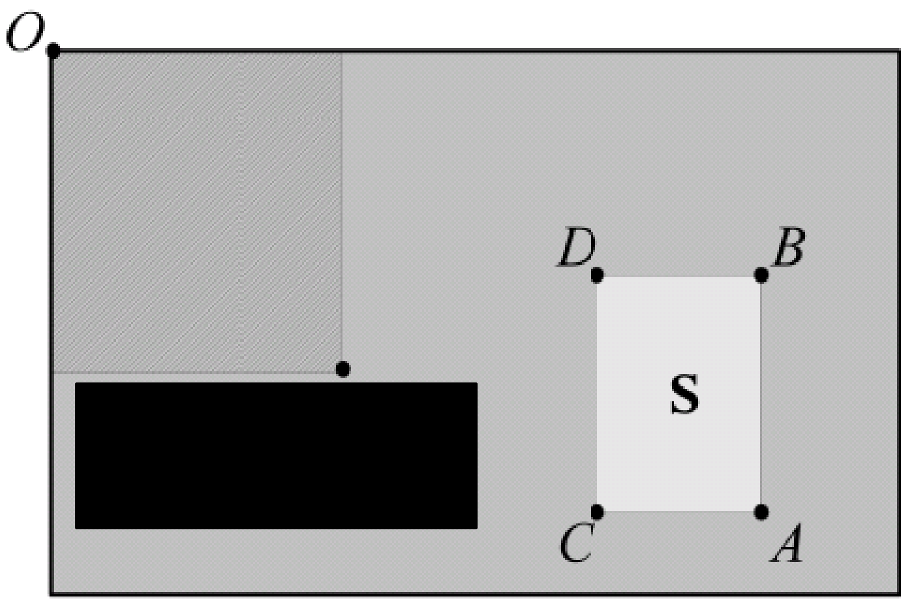
Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, “SURF: Speeded Up Robust Features”, European Computer Vision Conference (ECCV), 2006

SURF & the integral image

- SURF uses the integral image $I_{\Sigma}(x, y)$
  - Each pixel represents the sum of all pixels in the rectangle between $(0,0)$ and that pixel

$$I_{\Sigma}(x_i, y_i) = \sum_{i=0}^{x_i-1} \sum_{j=0}^{y_i-1} I(i, j)$$

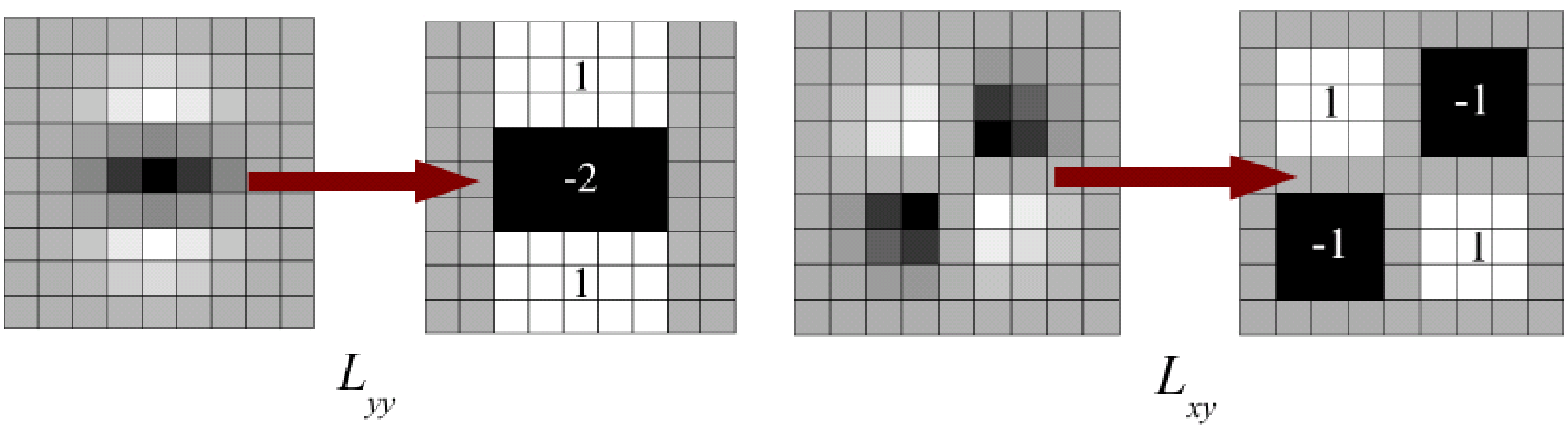
- S=A+D-B-C



Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, “SURF: Speeded Up Robust Features”, European Computer Vision Conference (ECCV), 2006

Box filters

- A box filter is composed of black and white rectangles
  - White: positive weight
  - Black: negative weight
- Box filters exploit the integral image
  - Calculation is constant-time irrespective of the filter size

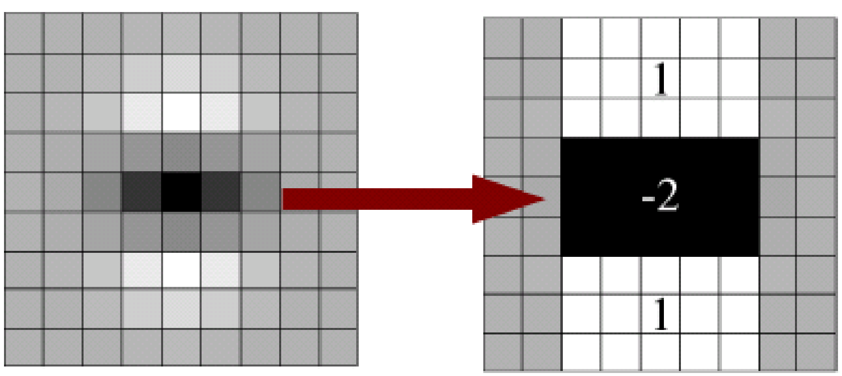


Box filters

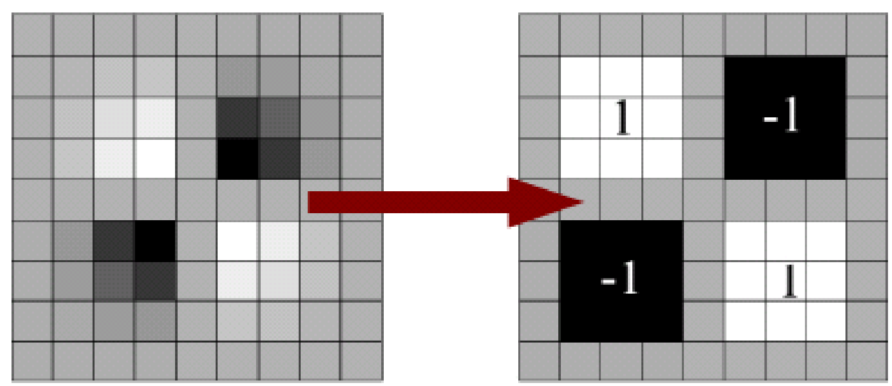
- Keypoint detection: compute the Hessian matrix in scale space

$$H = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix}$$

- Approximate L using box filters (fast calculation)
- Find the maximum of the determinant



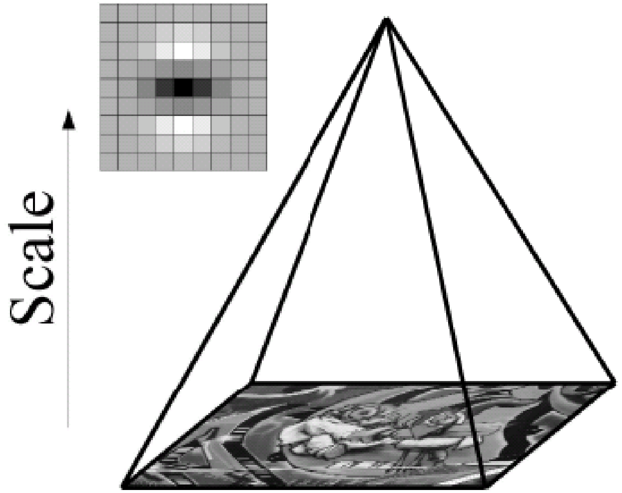
L_{yy}

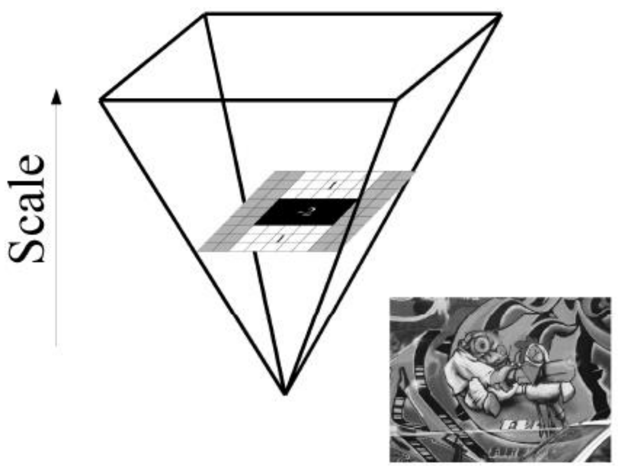


L_{xy}

Scale space with SIFT & SURF

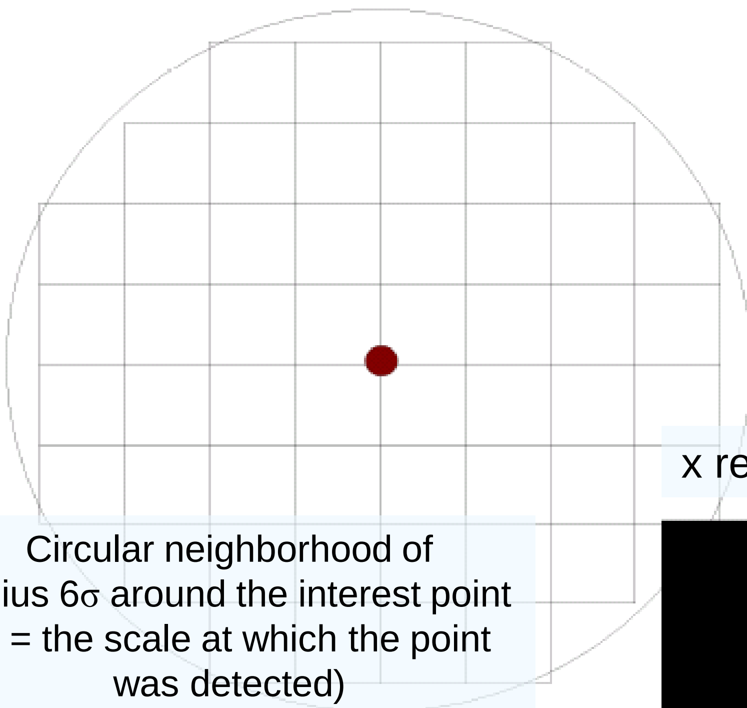
- SIFT: images are smoothed with a gaussian and subsampled to move along the pyramid
- SURF: use filters of increasing size
  - Computational time does not depend on filter size using the integral image!





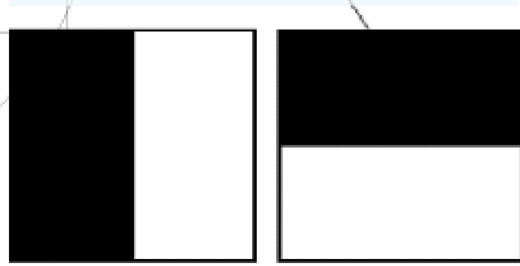
Orientation

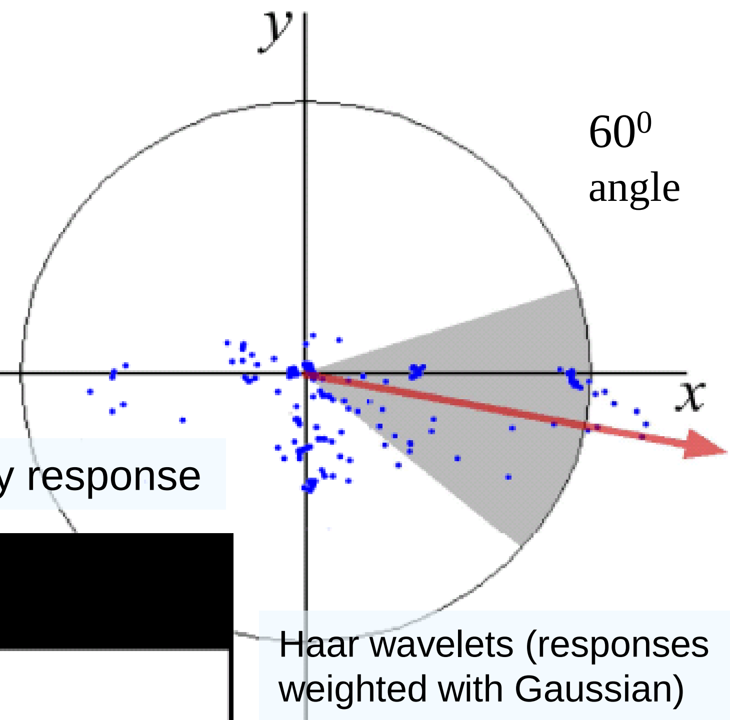
- Consider a neighborhood of size $6s$
  - s being the scale where the keypoint was found
- Evaluate gradients in x and y using the Haar wavelets
  - Responses smoothed using a gaussian



Circular neighborhood of radius 6σ around the interest point (σ = the scale at which the point was detected)

x response y response

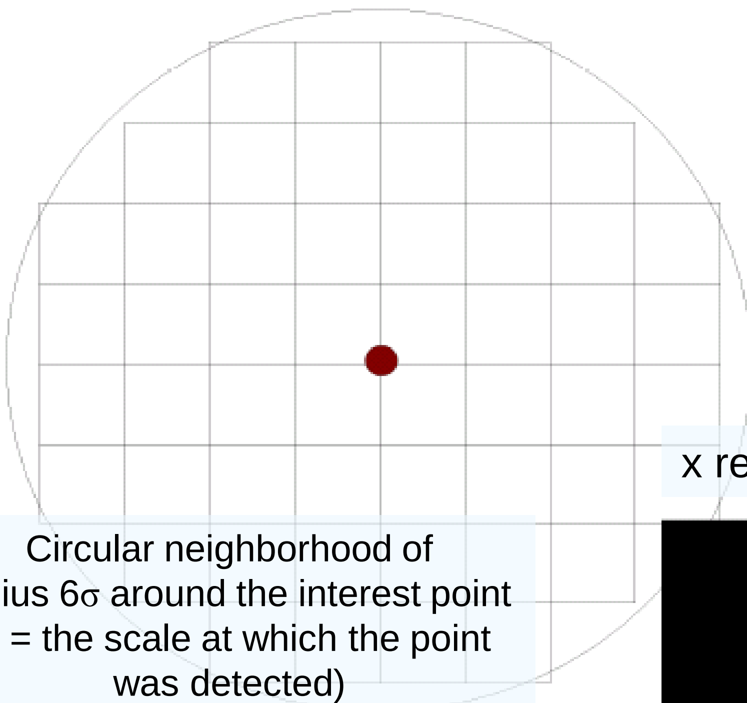




Haar wavelets (responses weighted with Gaussian)

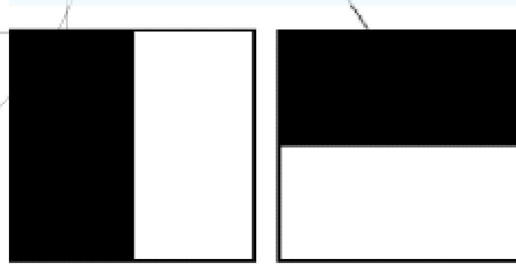
Orientation

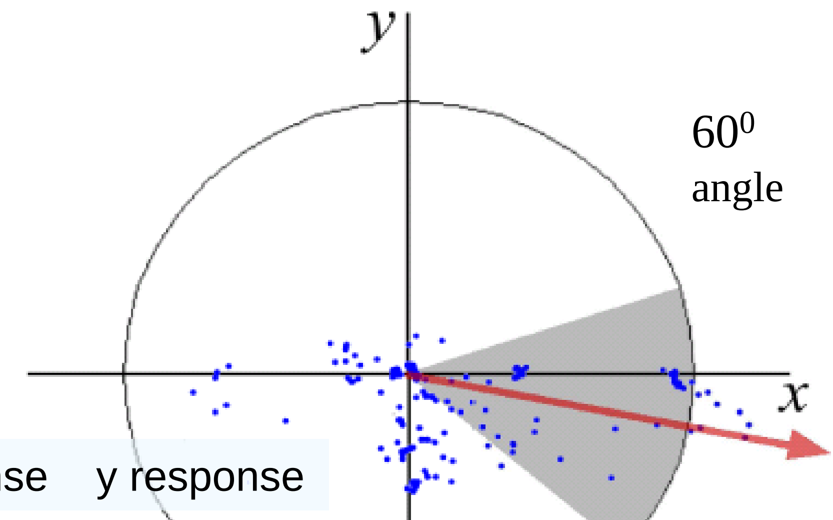
- Plot responses in a 2-dimensional space
- Sum horizontal and vertical response and determine the orientation
- Quantize in bins of 60°



Circular neighborhood of radius 6σ around the interest point (σ = the scale at which the point was detected)

x response y response

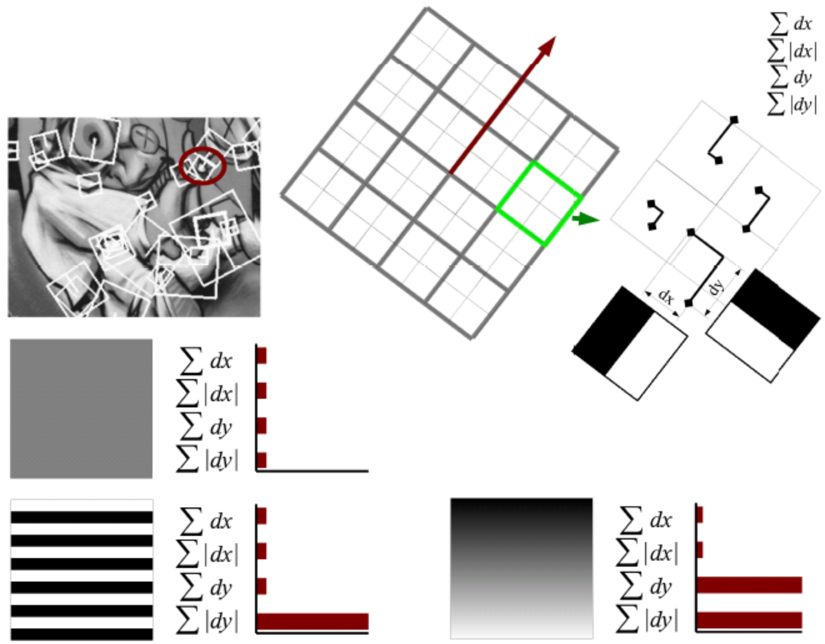




Haar wavelets (responses weighted with Gaussian)

Descriptor

- Consider 16 regions (same as SIFT)
- For each region:
  - Sum the response for d_x and d_y
  - Sum the modules for d_x and d_y
- Vector normalization
- 4 elements per region
  - Feature size: 4×16=64

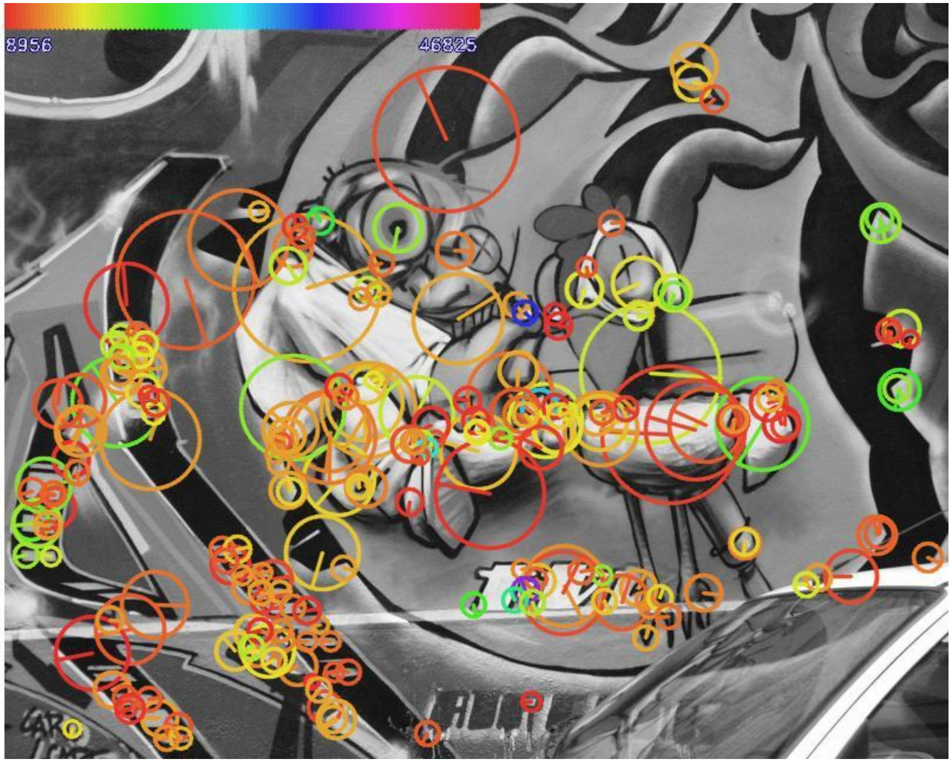


SURF – recap

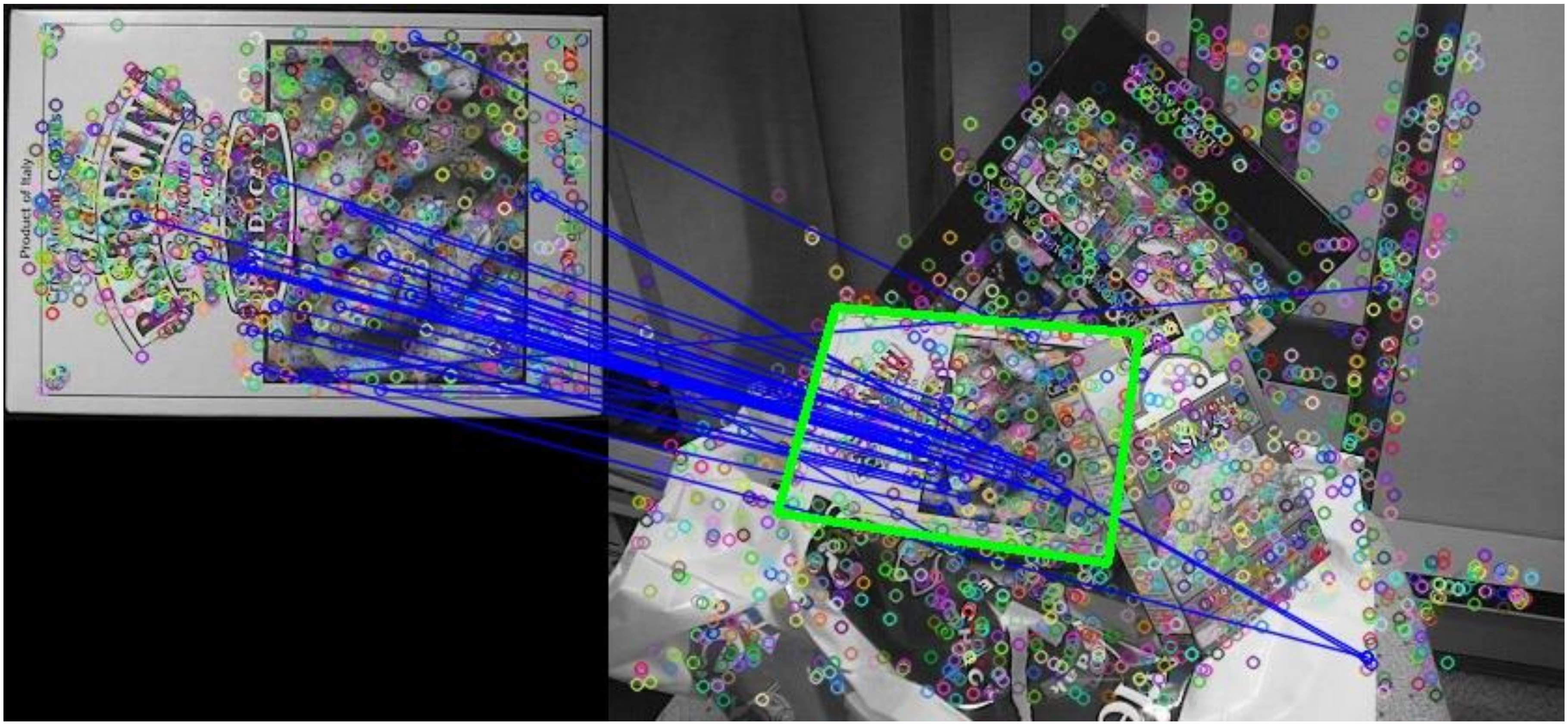
- Faster than SIFT
- Less robust to illumination and viewpoint changes WRT SIFT
- Both are patented

K. Mikolajczyk and C. Schmid,"A Performance Evaluation of Local Descriptors", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 10, pp. 1615-1630, 2005

SURF – example

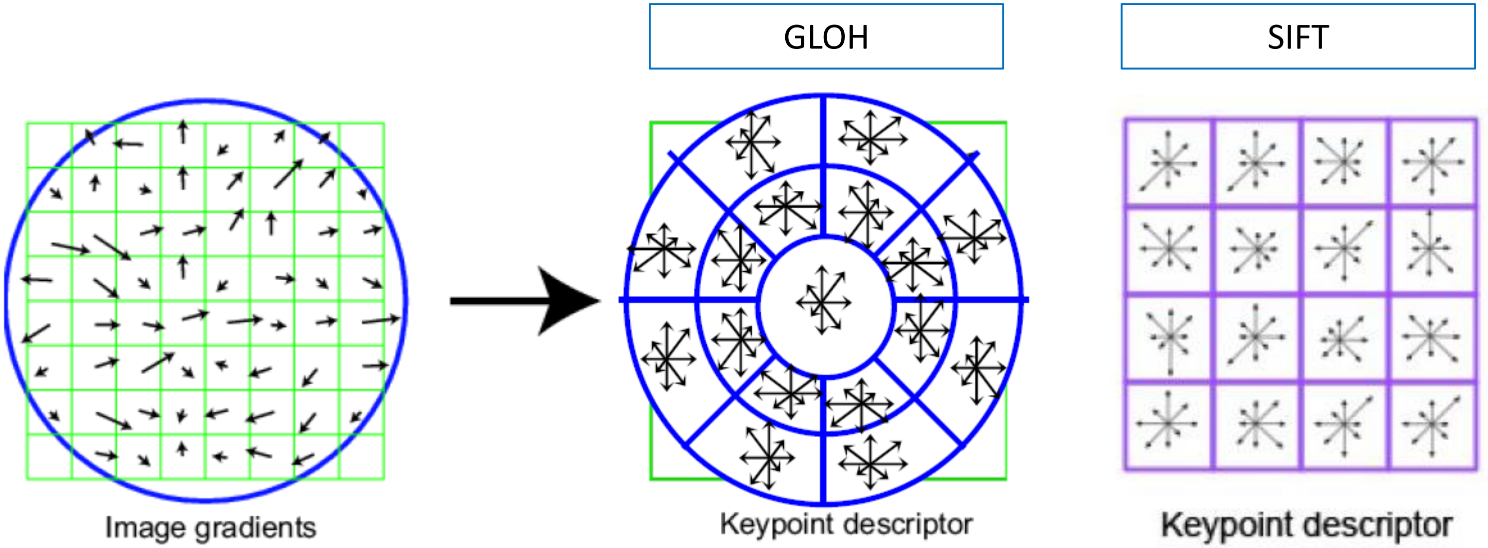


Surf – matching



GLOH

- Gradient Location-Orientation Histogram
- Obtained (again) modifying the 4th step of the SIFT algorithm
- Computes SIFT descriptor using a log-polar location grid



GLOH

- 3 bins in radial direction (radii: 6, 11, 15)
- 8 bins in angular direction (central bin not subdivided)
- 16 orientations
- Overall: (1+8+8)×16 = 272 orientation bins
- PCA used for reducing dimensionality to 128

GLOH

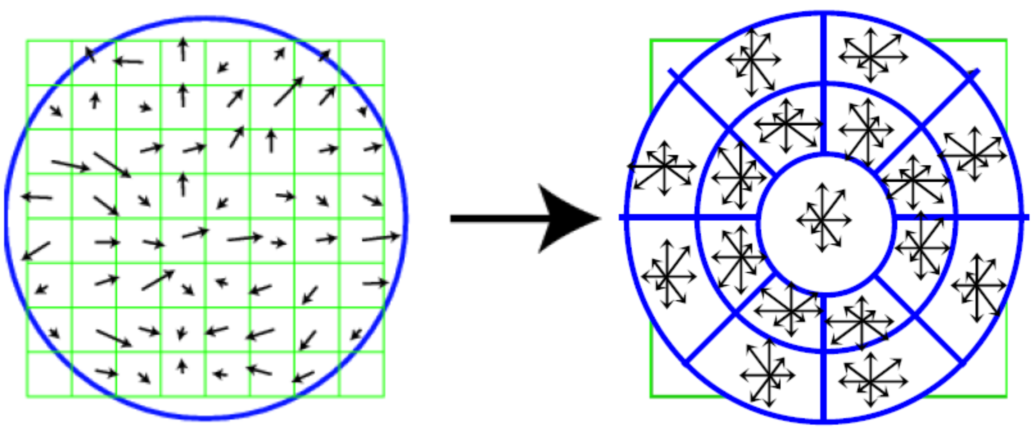
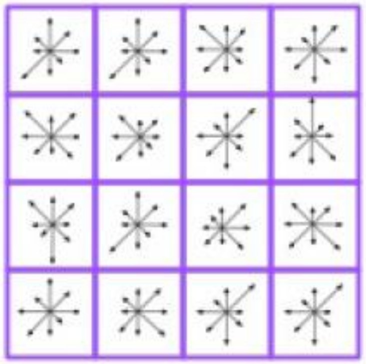


Image gradients

Keypoint descriptor

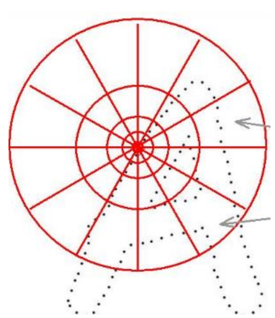
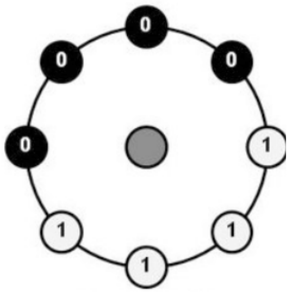
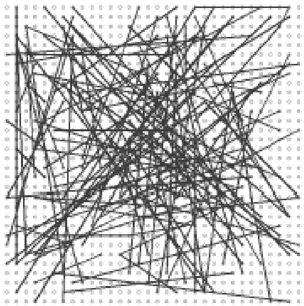
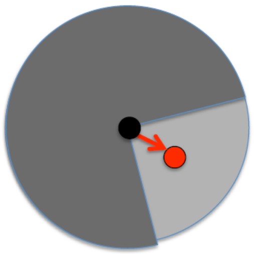
SIFT



Keypoint descriptor

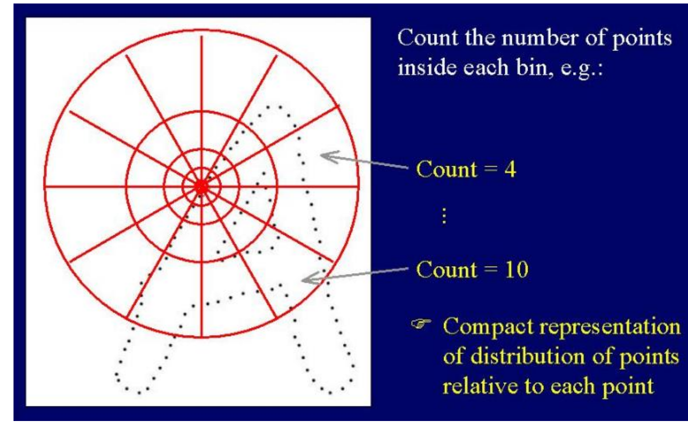
Other approaches

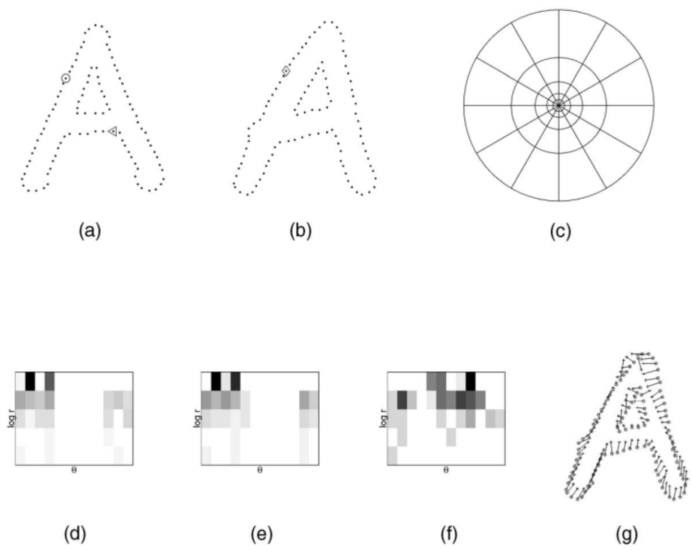
Other approaches

			
Shape Context	*LBP*	*BRIEF*	*ORB*

Shape context

- **Shape context descriptor**
- 3D histogram of edge point locations and orientations
  - Edges often extracted using Canny
- Location quantized using log-polar coordinate system
  - 5 bins for distance
  - 12 bins for orientation
  - 60 combinations

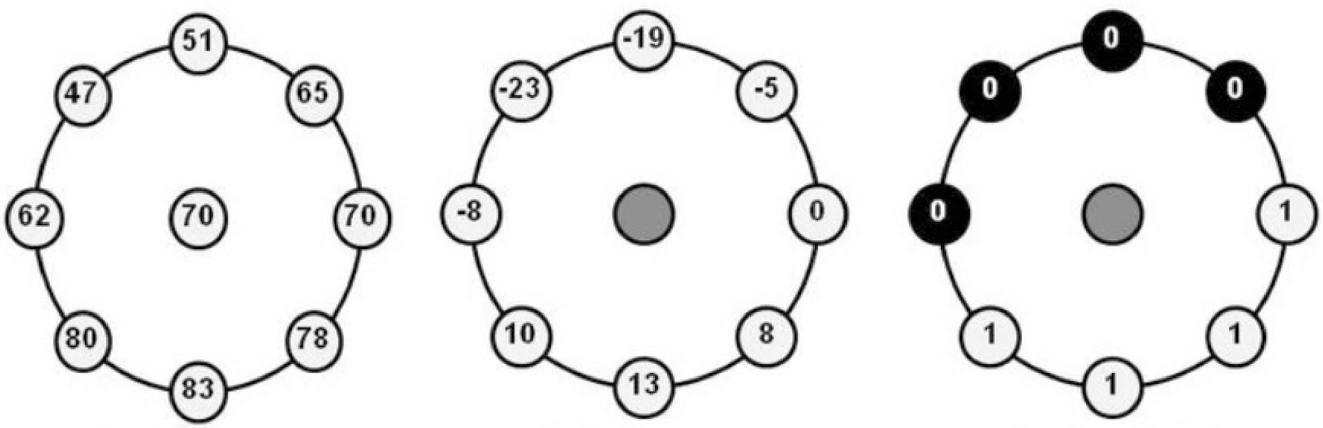




Serge Belongie, Jitendra Malik, and Jan Puzicha, "Shape matching and object recognition using shape contexts", IEEE Transactions on Pattern Analysis and Machine Intelligence, 24.4 (2002): 509-522

Local Binary Pattern (LBP)

- **LBP descriptor**
- First proposed for texture recognition in 1994
- Compare the central pixel with surrounding samples and build a binary sequence of signs of differences



1. Sample   2. Difference   3. Threshold

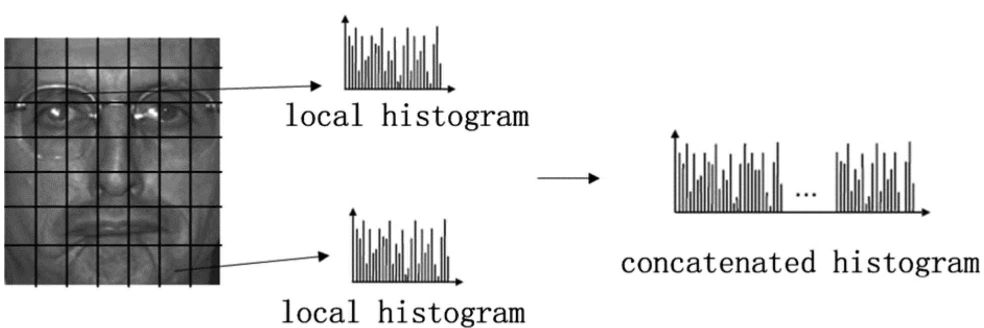
1*1 + 1*2 + 1*4 + 1*8 + 0*16 + 0*32 + 0*64 + 0*128 = 15

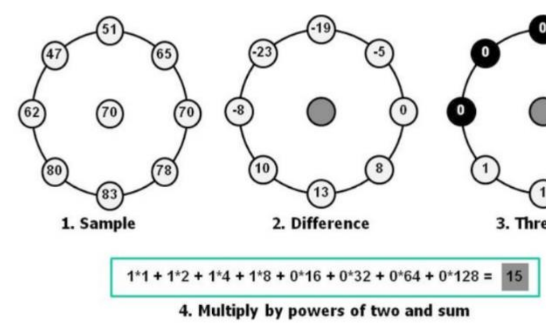
Ojala T., Matti P., Topi M., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns." IEEE Trans. on pattern analysis and machine int. 24.7 (2002): 971-987

LBP – approach

- Select a pixel and compare it with its 8 neighbors
- Follow the pixels along a circle: if it is > center, assign 1, 0 otherwise
- Compute the histogram of such values
- Histogram normalization (optional)
- A whole image can be analyzed divided into subwindows (histograms are concatenated)

$$\tau(x, x_i) = \begin{cases} 1 & if\ i(x) > i(x_i) \\ 0 & if\ i(x) \leq i(x_i) \end{cases} \qquad f(x) = \sum_{i=1}^{8} 2^{i-1} \tau(x, x_i)$$





Cita 250

BRIEF descriptor

- Binary Robust Independent Elementary Features
- Based on:
  - Gaussian smoothing
  - Pairs of pixels compared inside a window

$$\tau(\boldsymbol{p}; x, y) = \begin{cases} 1 & if\ p(x) < p(y) \\ 0 & otherwise \end{cases}$$

  - Build a vector with comparison output

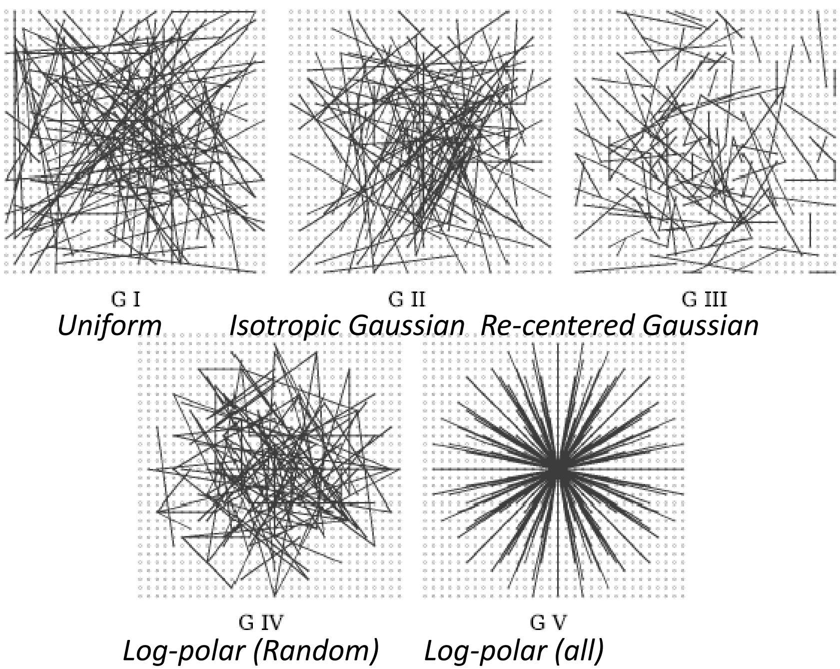
$$f(\boldsymbol{p}) = \sum_i 2^{i-1} \tau(\boldsymbol{p}; x, y)$$

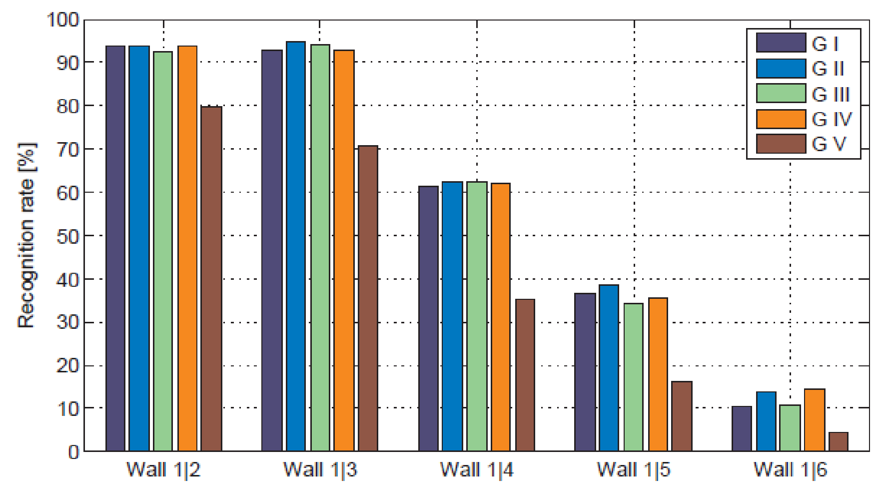
- Vectors compared using the Hamming distance (XOR)
- **Homework: check the documentation***

M Calonder, V Lepetit, C Strecha, P Fua, "Brief: Binary robust independent elementary features", ECCV 2010
*Documentation: https://medium.com/data-breach/introduction-to-brief-binary-robust-independent-elementary-features-436f4a31a0e6

BRIEF descriptor

- Fixed sampling pattern of 128, 256 or 512 pairs
- Random pattern provides the best performance
  - Best solution: isotropic gaussian distribution

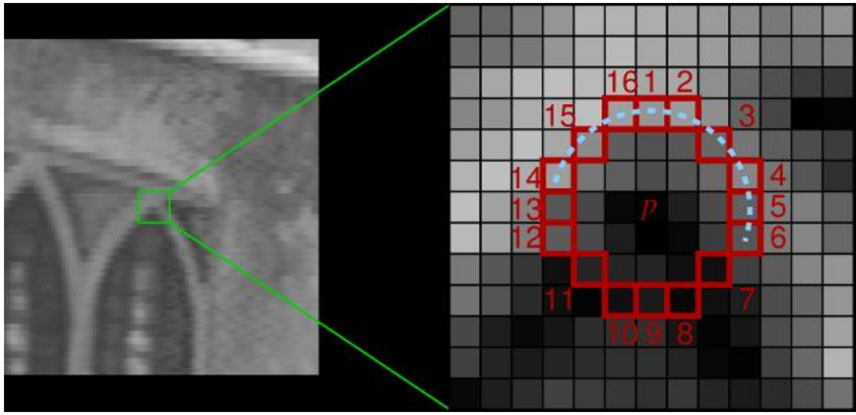


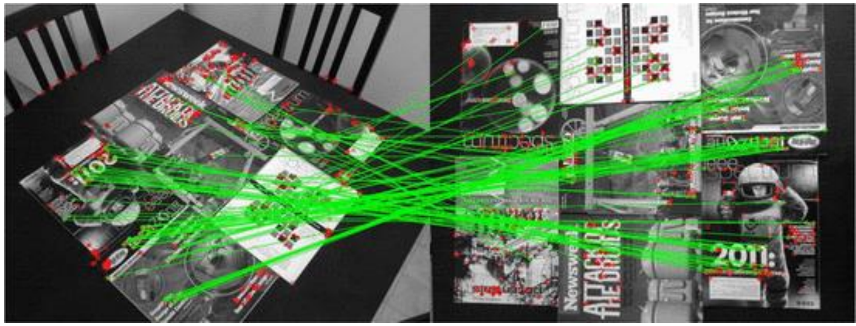


BRIEF – results

ORB

- Oriented FAST and Rotated BRIEF
- FAST corner detector
- Add rotation invariance to BRIEF
- Orientation assignment based on the intensity centroid WRT the central pixel
- **Homework: check the documentation***

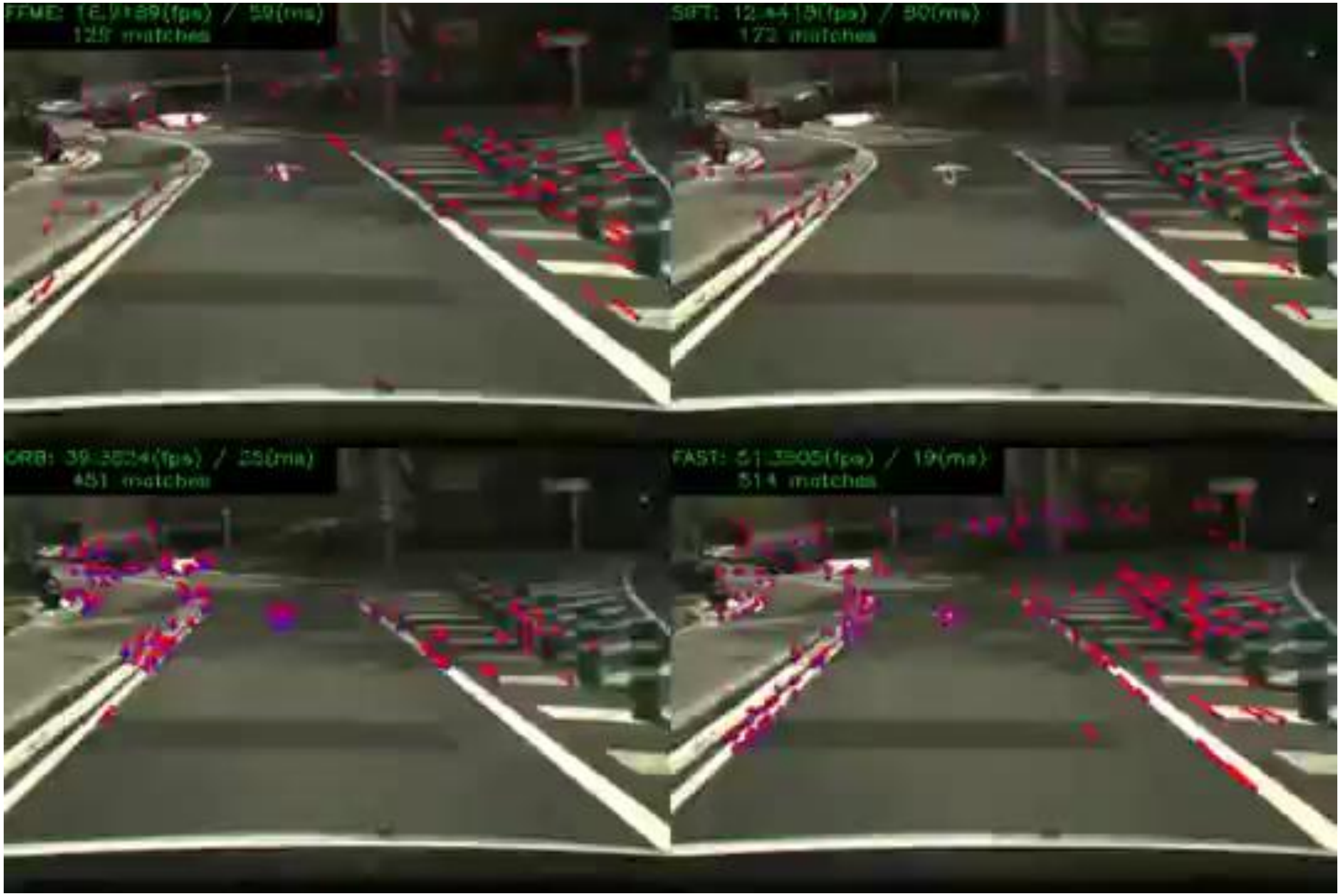




Rublee et al., "ORB: An efficient alternative to SIFT or SURF", ICCV 2011
*Documentation: https://medium.com/data-breach/introduction-to-orb-oriented-fast-and-rotated-brief-4220e8ec40cf

Feature extraction example

- Comparison of SIFT, ORB and FAST



UNIVERSITÀ DEGLI STUDI DI PADOVA

An overview of existing features

Stefano Ghidoni