SIFT, MSER, SURF, LBP, BRIEF, BRISK, DSIFT, GIST

Gary Overett (Slides adapted from CMU 16-720 2014)

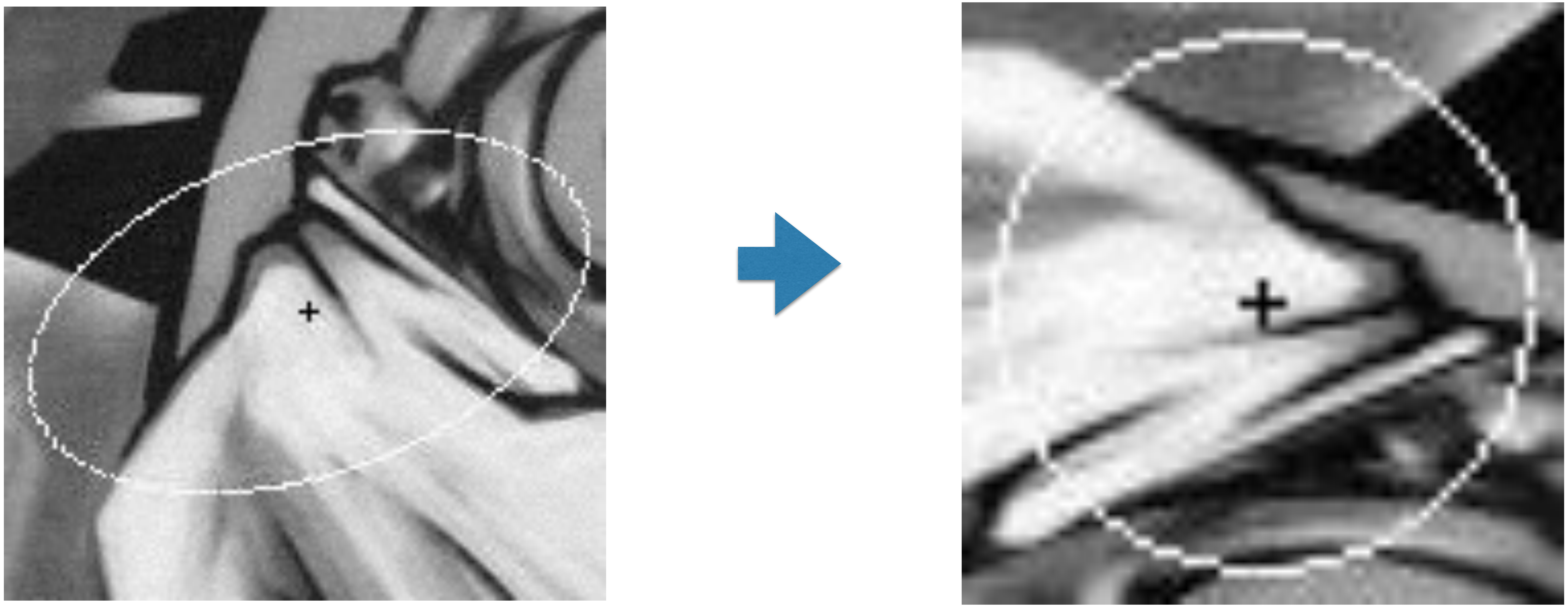
Descriptors

- We have scale or affine invariant interest points with region around each of them
- Descriptor = vector describing the image content in the neighborhood around the interest point

Descriptors

- Example: Affine neighborhood

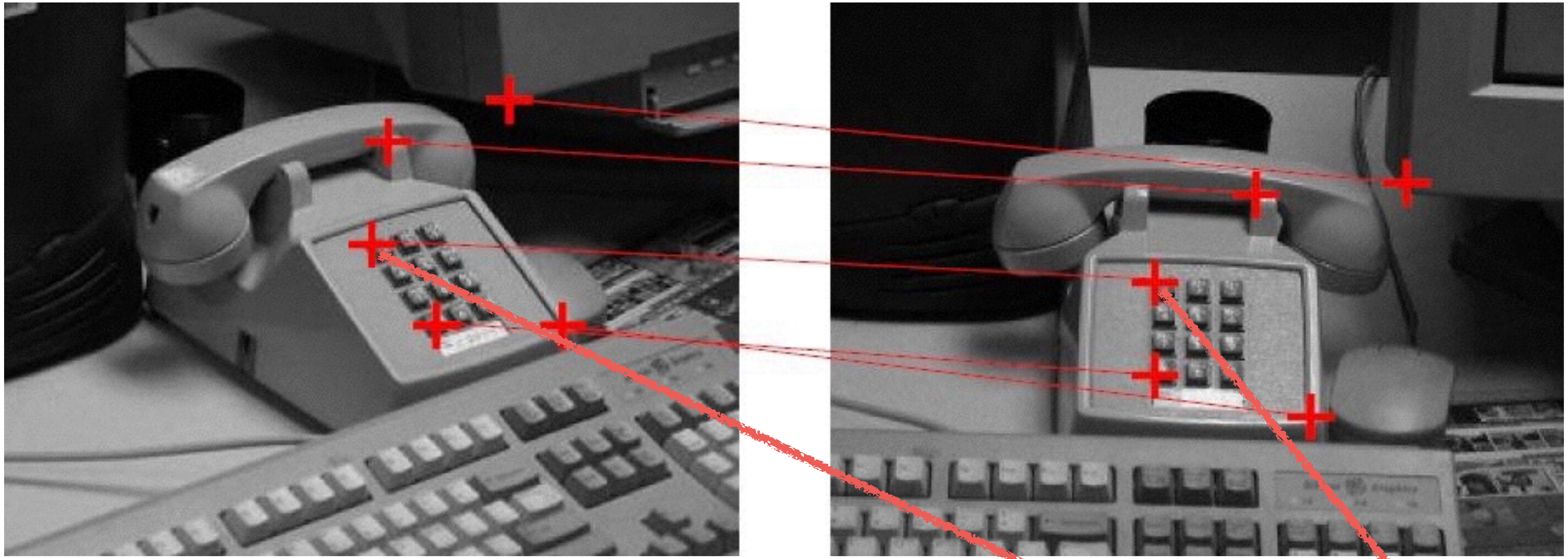
- First transform neighborhood to canonical square window, for example: 16x16 window



Outline

- Descriptors as continuos feature vectors
    - Local Gradient Histograms : SIFT
    - Faster Version : Surf
- Descriptor as Binary Code : LBP, BRIEF, BRISK, …
- Descriptors as Densely Sampled Points
- Global Descriptors : GIST

Descriptors as continuous vectors



Descriptor = representation of the distribution of pixels in a neighborhood suitable for computing a “distance” between . . feature points

$$d(p_1, p_2) = \left\| \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix} - \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix} \right\|$$

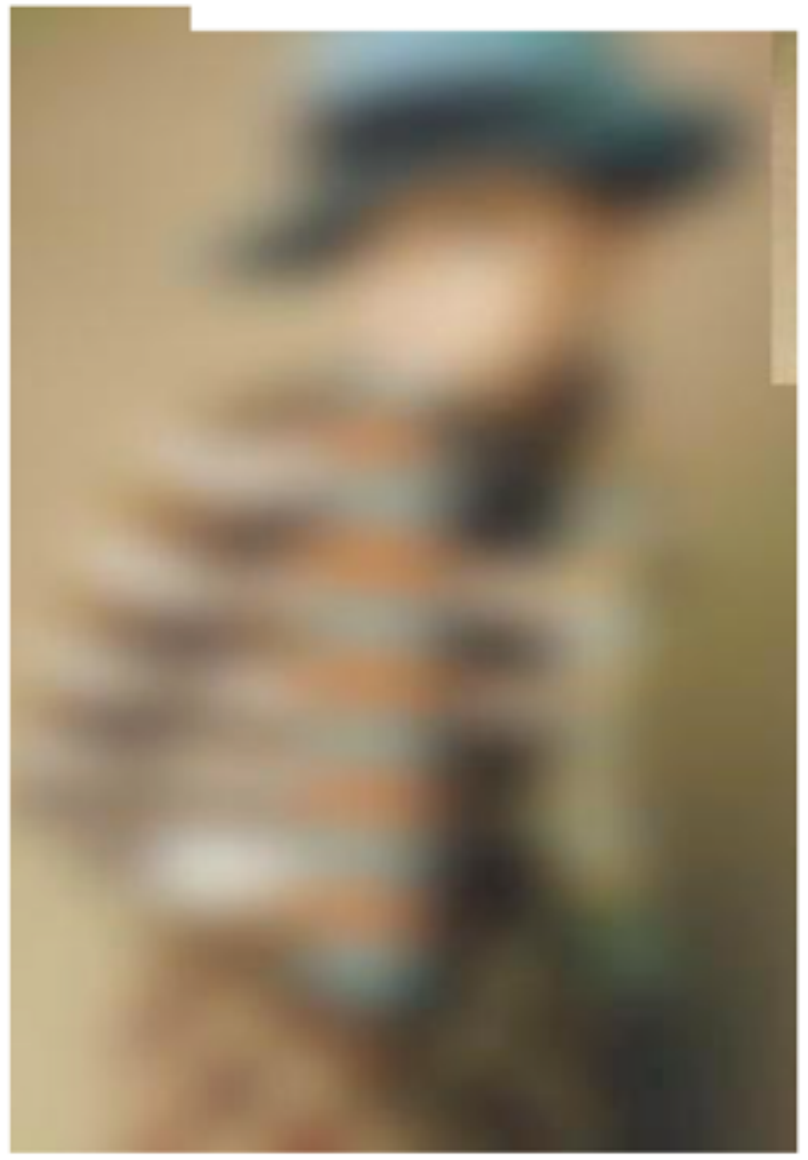
Conflicting criteria

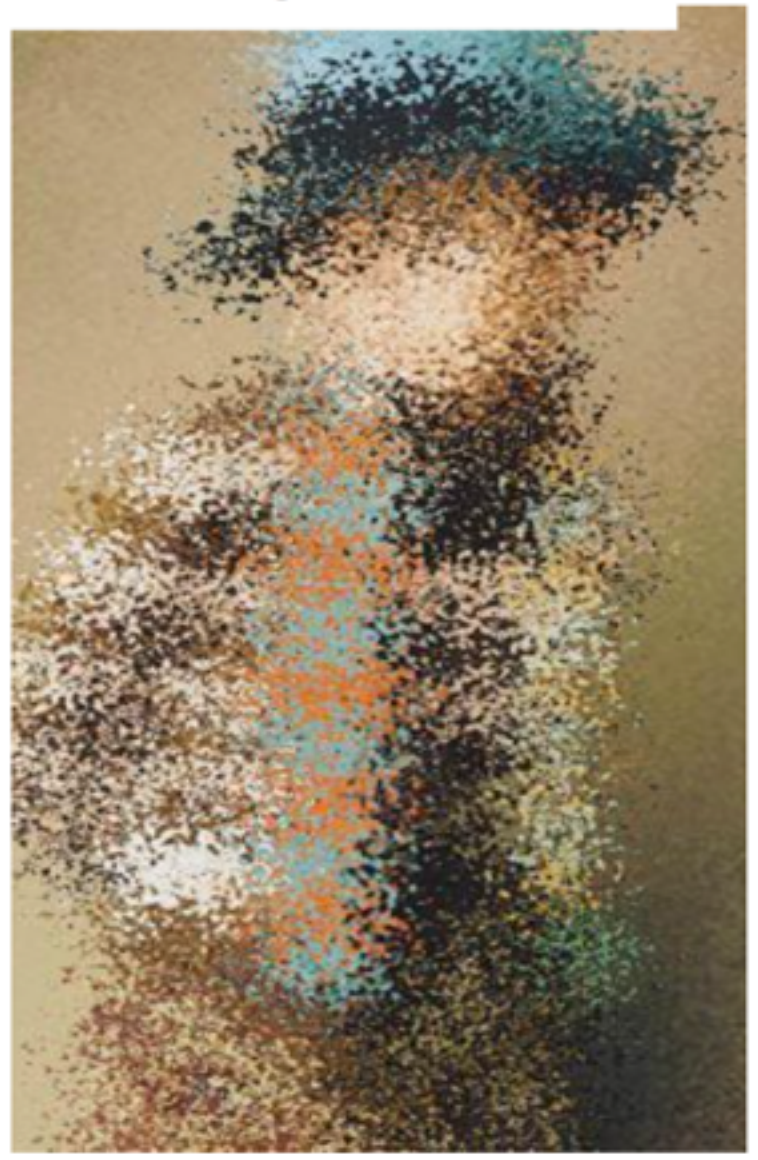
- Discrimination:
  – $d(p_1,p_2) \gg d(p'_1,p'_2)$ if $d(p'_1,p'_2)$ is a true correspondence and $d(p_1, p_2)$ is not.
- Generalization:
  – Robust to changes in illumination, viewpoint,...
- Speed:
  - Fast to compute (obvious!) AND
  - Low-dimension (not as obvious)
- Note: Maybe we can learn for discriminability and generalizability

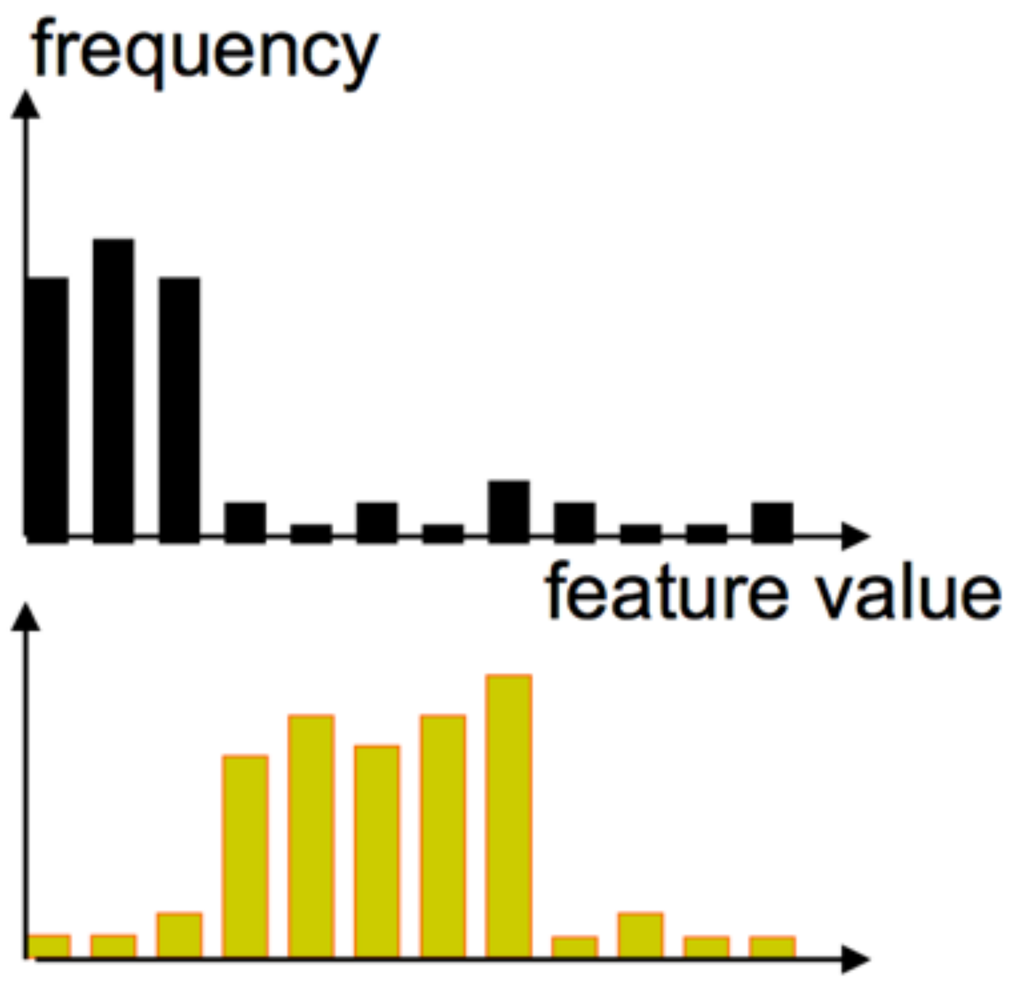
Aside : Image Representations

Discrimination power

Image patch:







1. Template classification: Techniques inspired from template matching: Compare directly with pixels from training image. Perhaps use blurring to be robust to small shifts, etc.

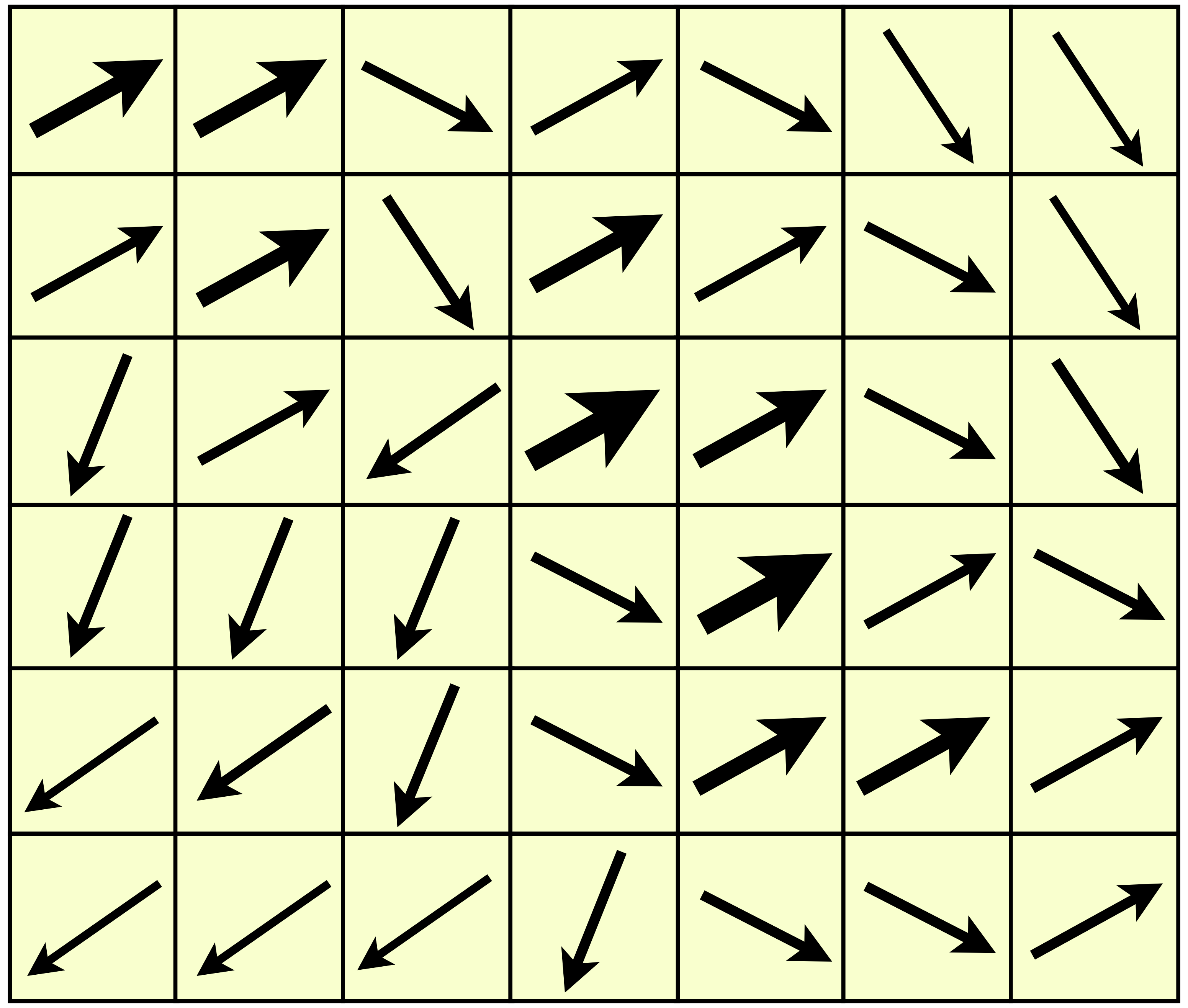
2. Locally orderless structures: Intermediate solution: Use local histograms of features instead of a single global histogram. Potentially combines the advantages of templates (strong spatial information) and histograms information) (robustness to distortions of the image)

3. Histograms: Use histograms of features to compare the windows. Example: Bags Of Words techniques. Problem: All local spatial information is lost since the representation is global. Will be robust to changes but much less discriminative.

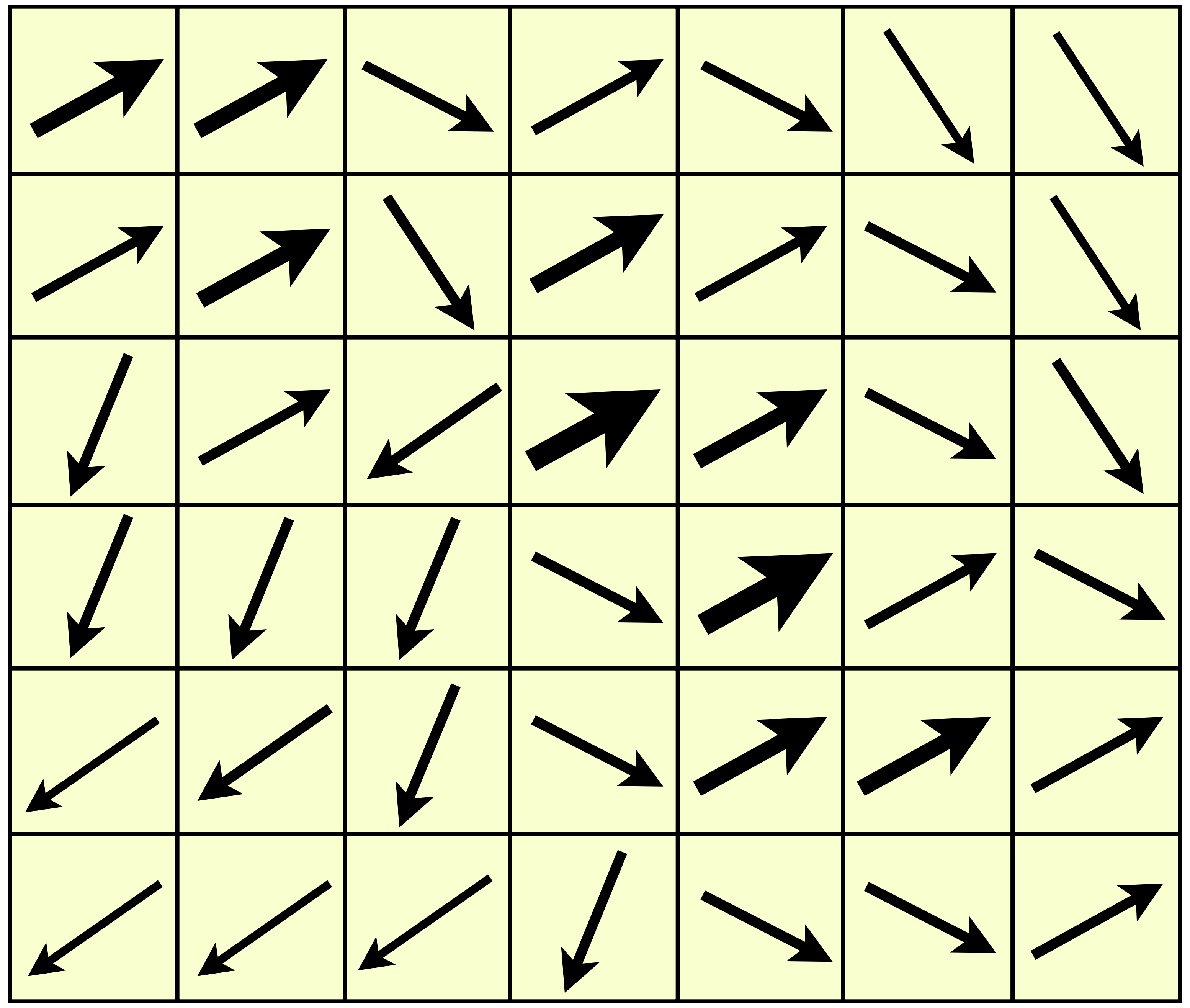
Generalization power

Example from: The Structure of Locally Orderless Images, Jan J. Koenderink AND A. J. Van Doorn, IJCV 1999.

Primer : Histograms of Oriented Gradients

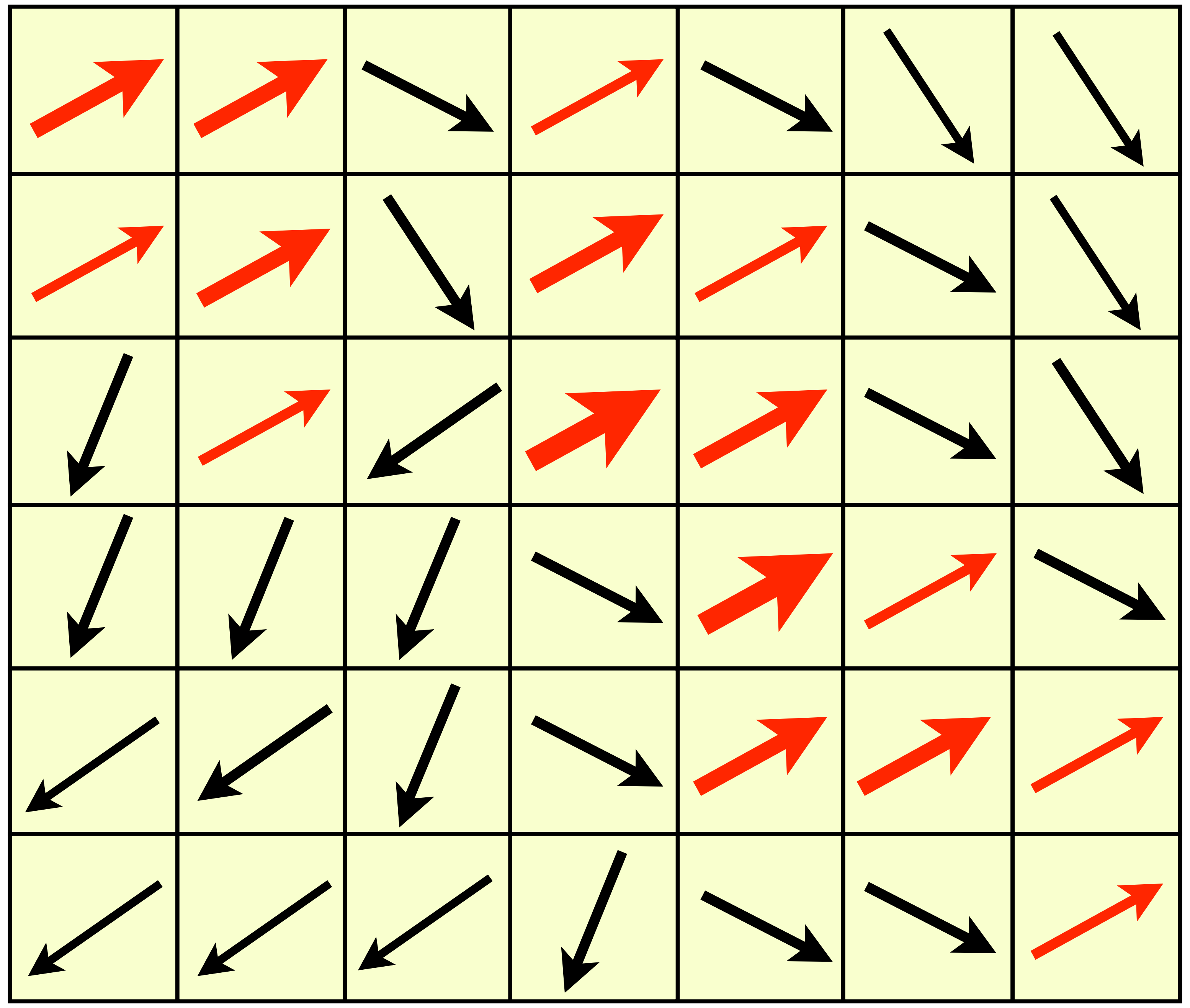


Primer : Histograms of Oriented Gradients

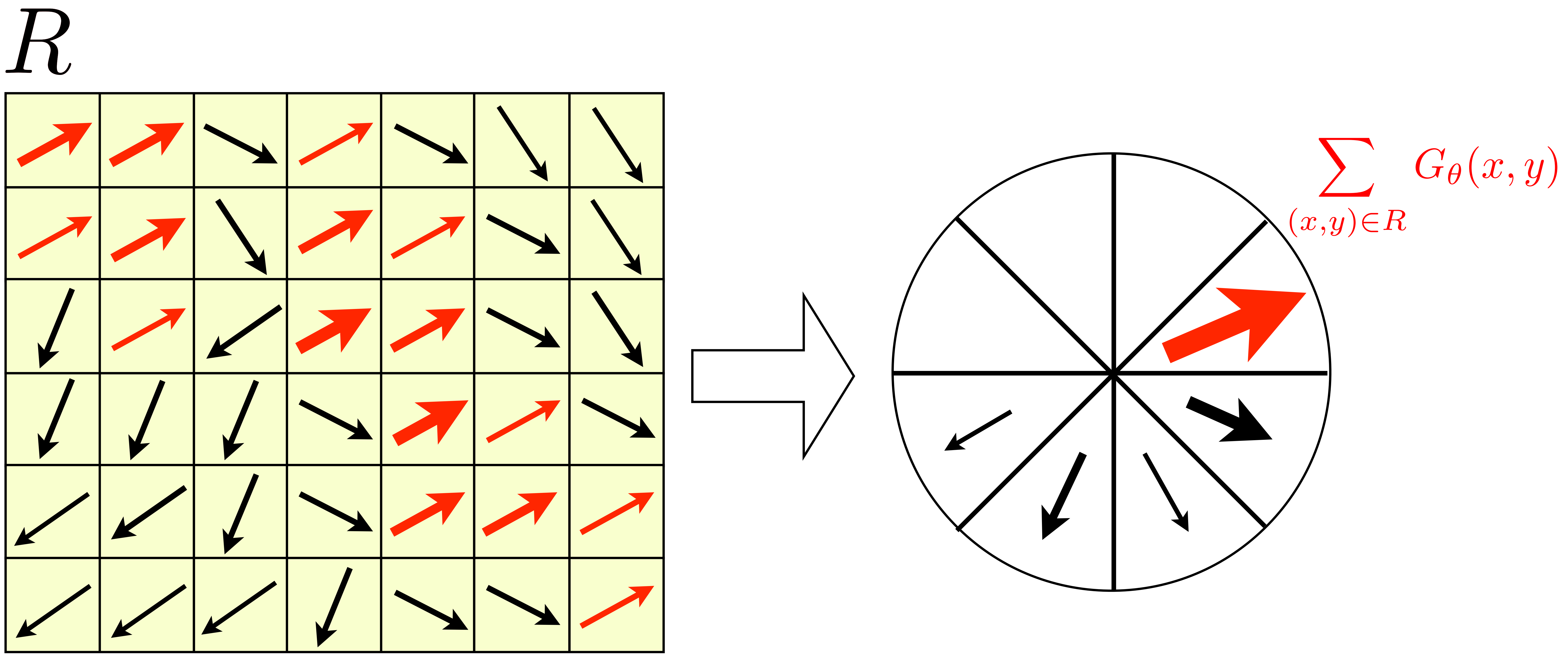


Primer : Histograms of Oriented Gradients

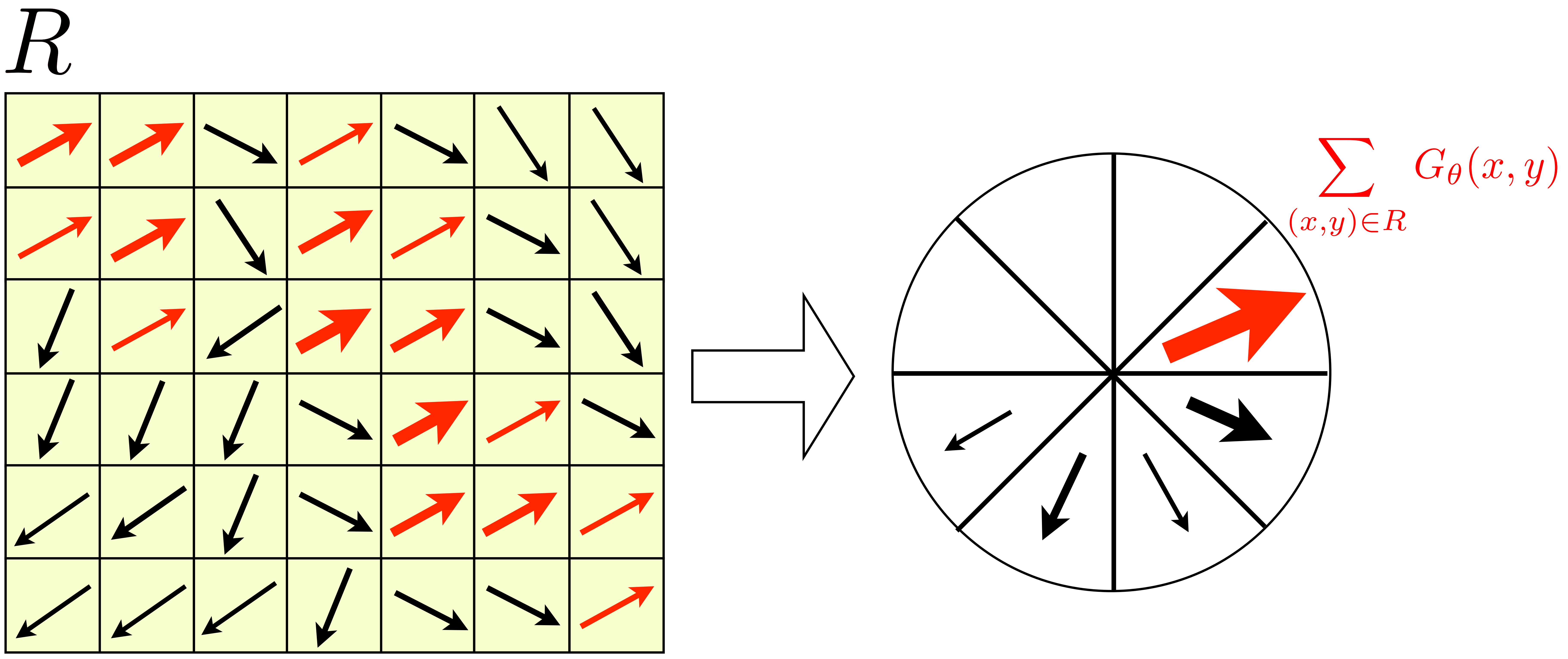
R



Primer : Histograms of Oriented Gradients



HOG : Histograms of Oriented Gradients

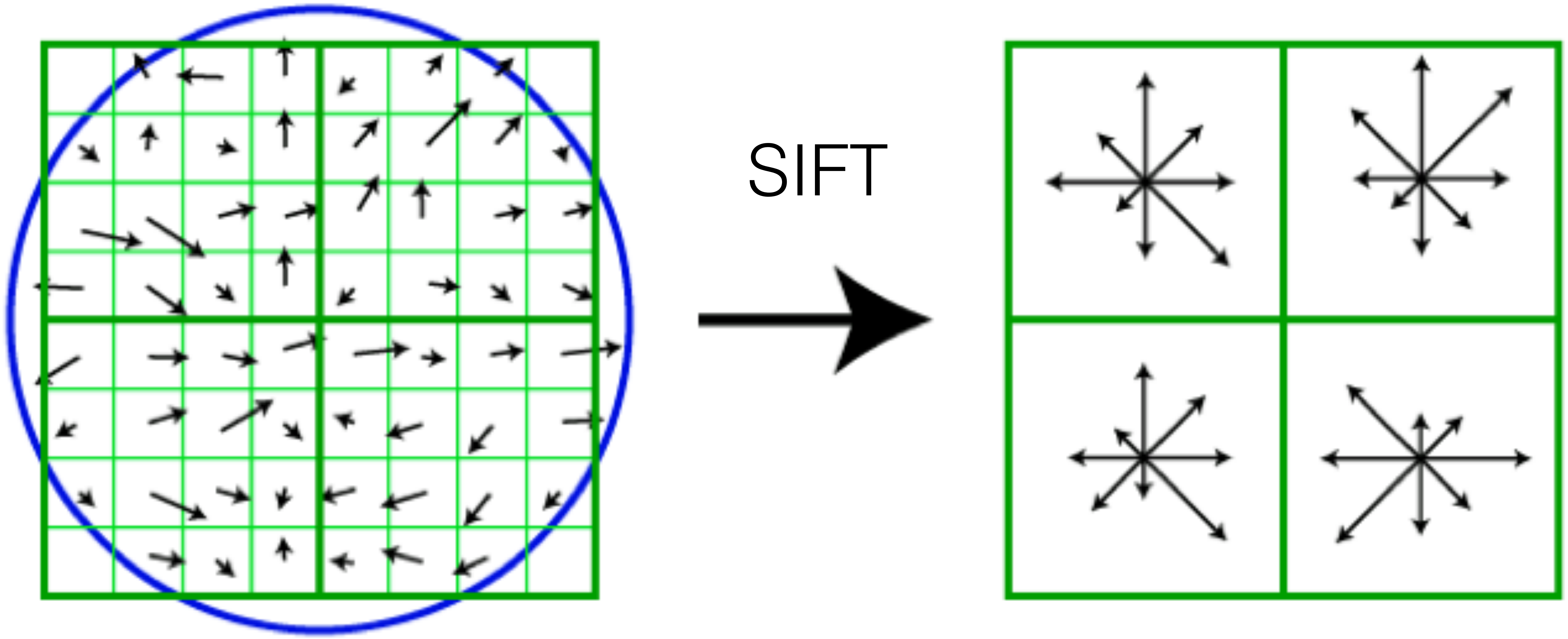


- In this case an 8-vector histogram of oriented gradients
- But there are many other configurations

SIFT : Scale Invariant Feature Transform

- Image gradients are sampled over 4kx4k array of locations around interest point
- Create array of orientation histograms over 8 orientations in each of k^2 4x4 blocks
- 8 orientations x kxk histogram array $N = 8\ k^2$ dimensions
- In practice $k = 4$ $N = 128$

k=2



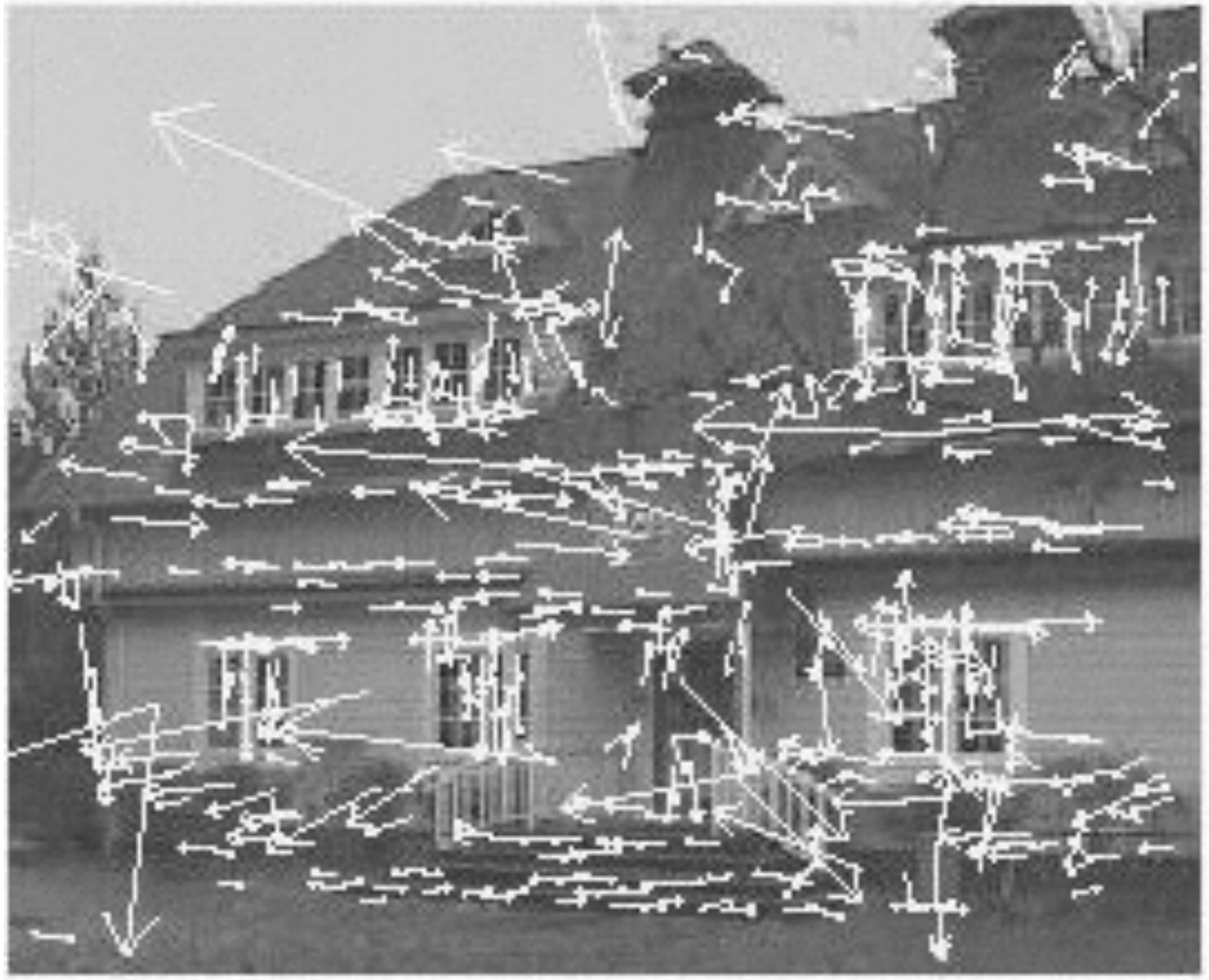
N=32

Image gradients

Keypoint descriptor

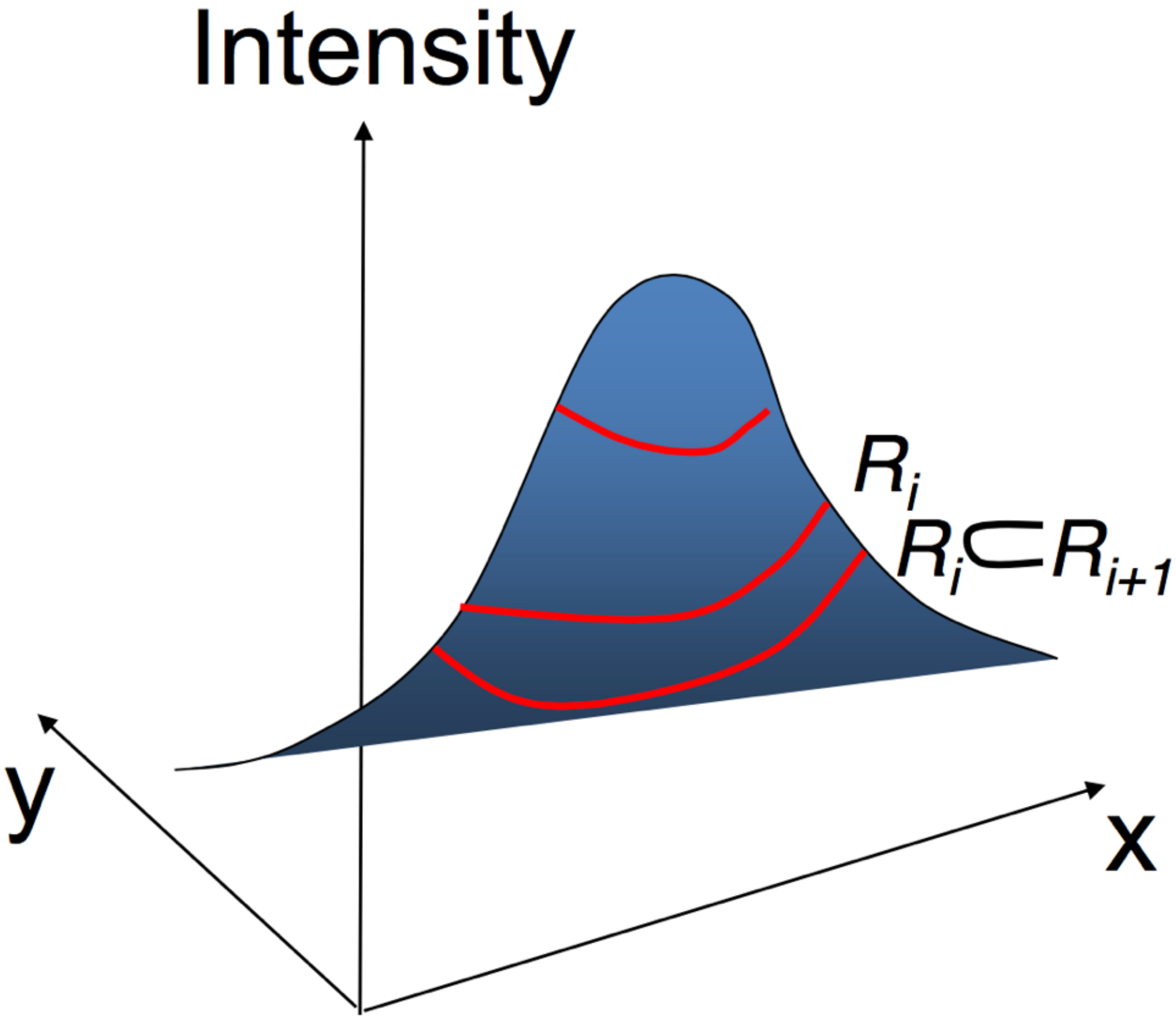
SIFT Tweak : Adding Rotational Invariance

- Descriptor is not rotationally invariant -> Select a dominant direction and express all the gradient orientations with respect to the dominant direction

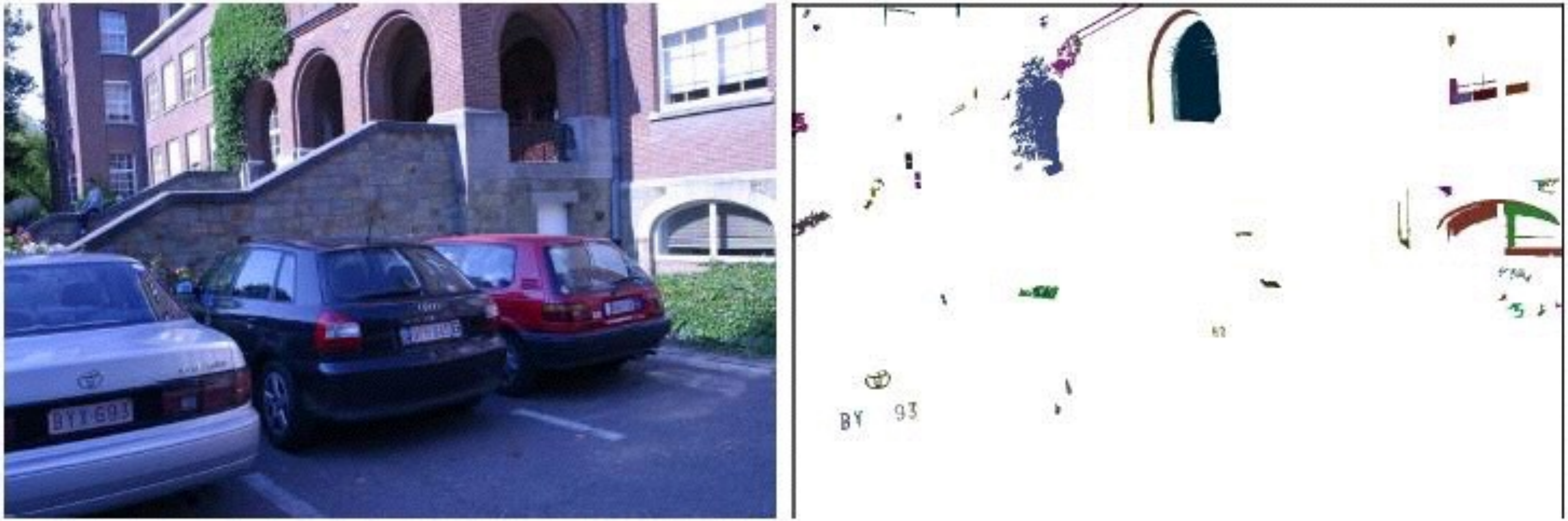


Regions Instead of Points : MSER

- MSER = Maximally stable extremal regions
- Objective find stable (=interest) regions
- Considers Images as regions made up of “contours” of equal intensity.
- Compares the ratio of nested contours to find a strong (maximally stable) contrasting region.



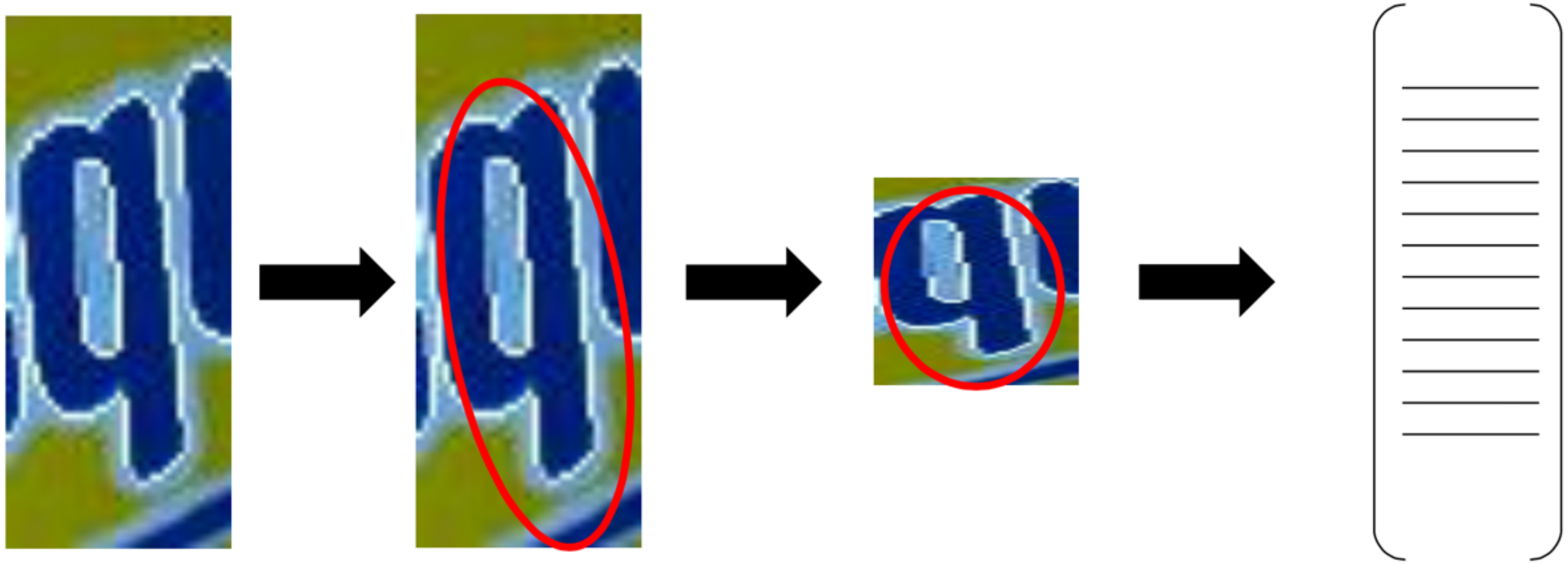
MSER : Maximally Stable Extremal Regions



Example for Forssen&Lowe, ICCV 2007.

Intensity
R_i
$R_i \subset R_{i+1}$
y
x
$|R_i|$
i
$(|R_{i+1}| - |R_i|) / |R_i|$
i
Maximally stable region

Combining MSER & SIFT



Find extremal region

Fit ellipse Eigenthings of:

Rotate + scale to canonical patch

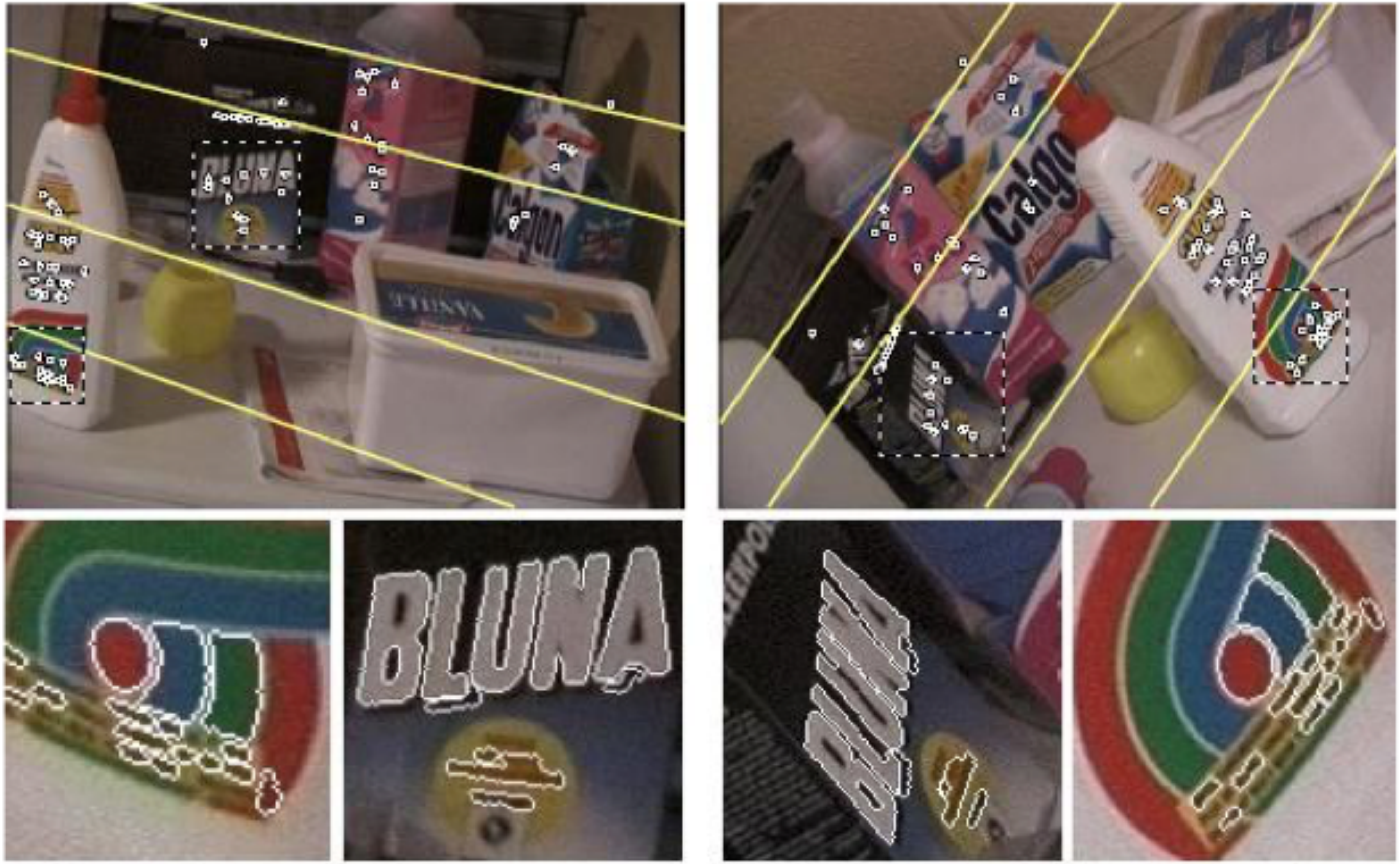
Compute 128-dim SIFT vector

$$\sum_{region}(P-\bar{P})(P-\bar{P})^T =$$

$$M = \begin{bmatrix} \sum_{region}(x-\bar{x})^2 & \sum_{region}(x-\bar{x})(y-\bar{y}) \\ \sum_{region}(x-\bar{x})(y-\bar{y}) & \sum_{region}(y-\bar{y})^2 \end{bmatrix}$$

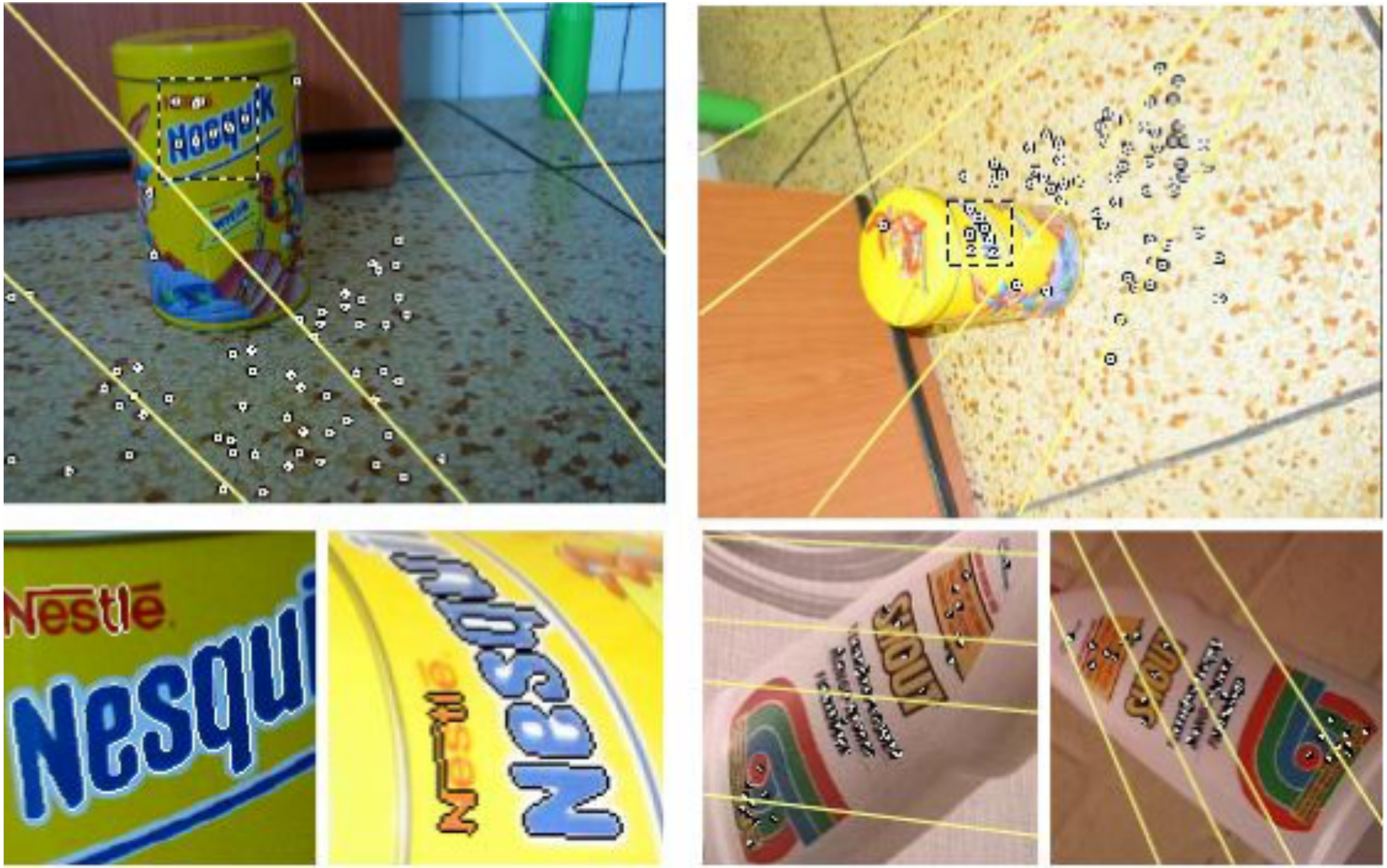
Examples from Image Matching

Example from Matas et al. BMCV 2002



Examples from Image Matching

Example from Matas et al. BMCV 2002



Examples from Image Matching

SIFT Features are Bit Slow

- Gradients and scales issues make SIFT a little heavy to compute
- Let's fix that…

But First : A Cool Idea - Integral Images

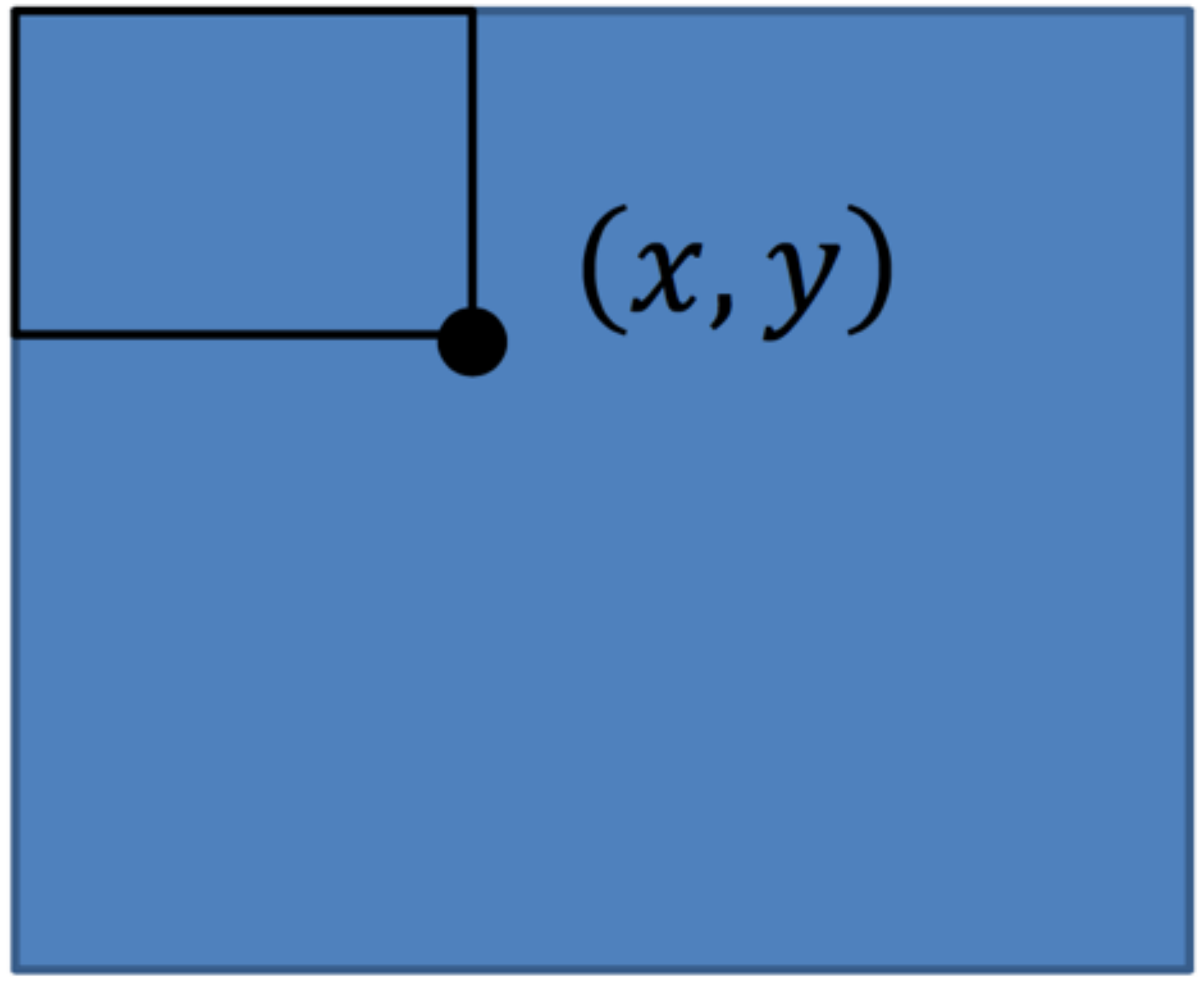
- A really really fast way to find the **average** intensity of a **rectangular** image region.

- My creating a new data structure - the **Integral Image**

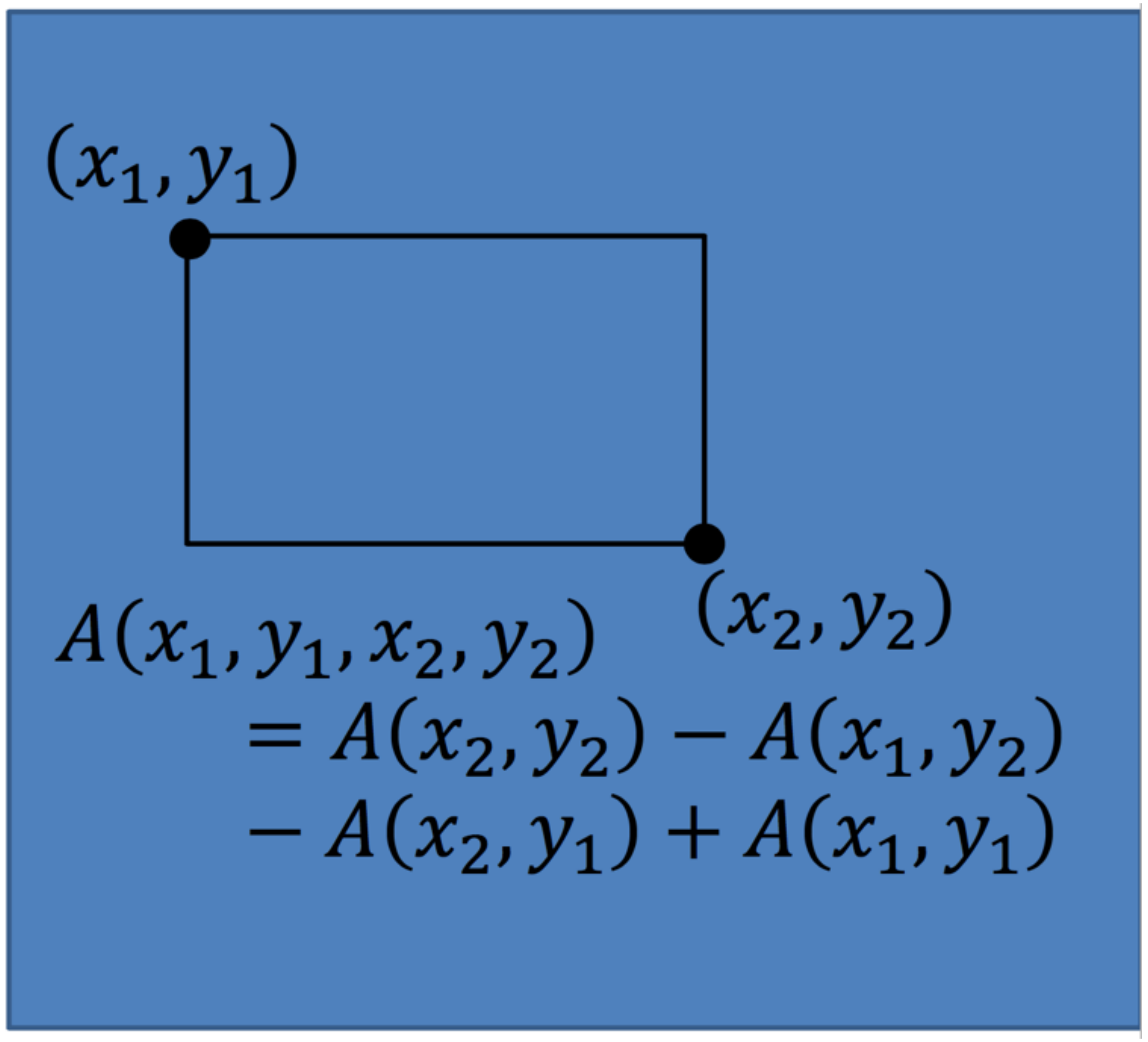
- Let's see…

Integral Image

- Haar wavelets = Bunch of sums over rectangular regions
- Integral image: Sum over a rectangular region of any size can be computed in 3 operations



$$A(x, y) = \sum_{x' \le x, y' \le y} A(x', y')$$



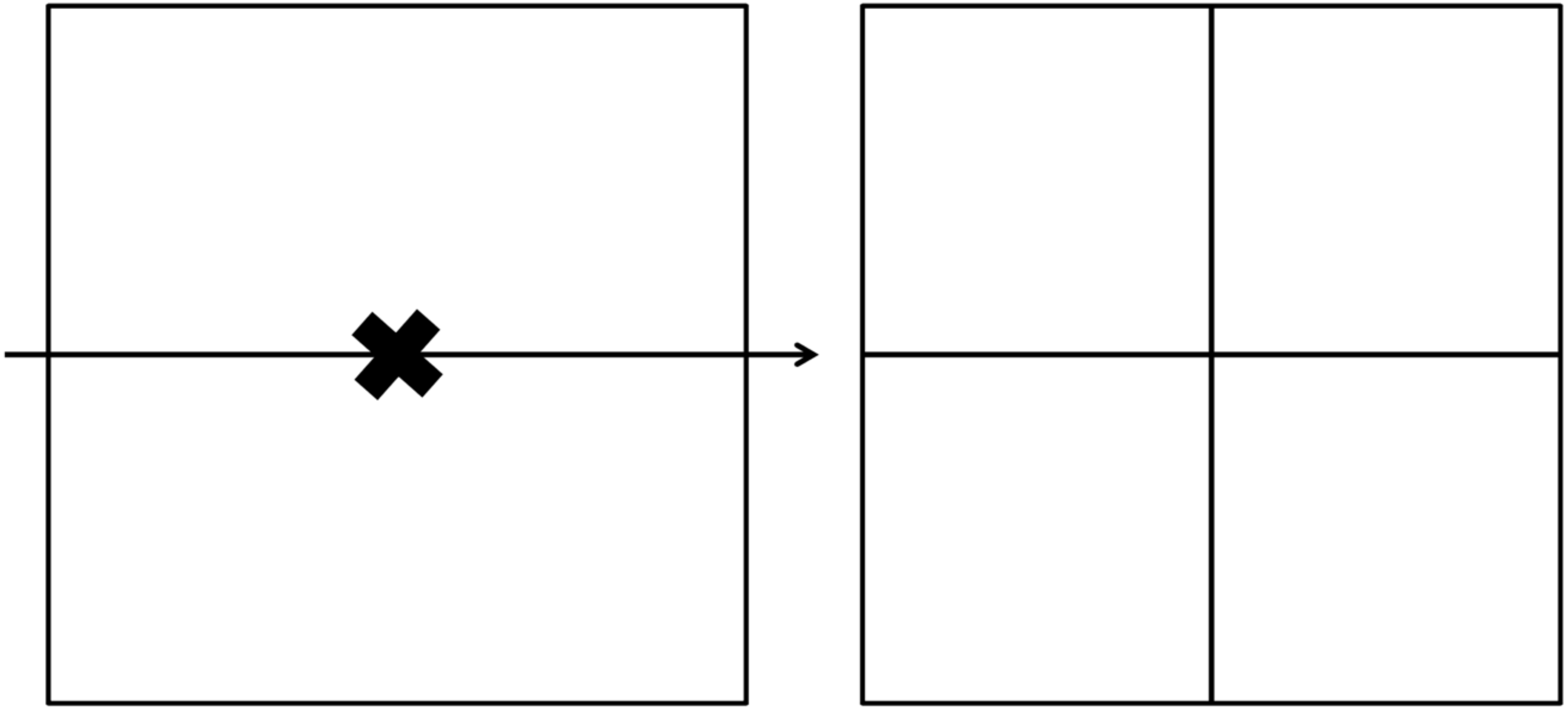
Integral Image

Integral Image

Haar Wavelets

SURF : Speeded Up Robust Features

- Fast detector by replacing Gaussian derivatives by Haar wavelets

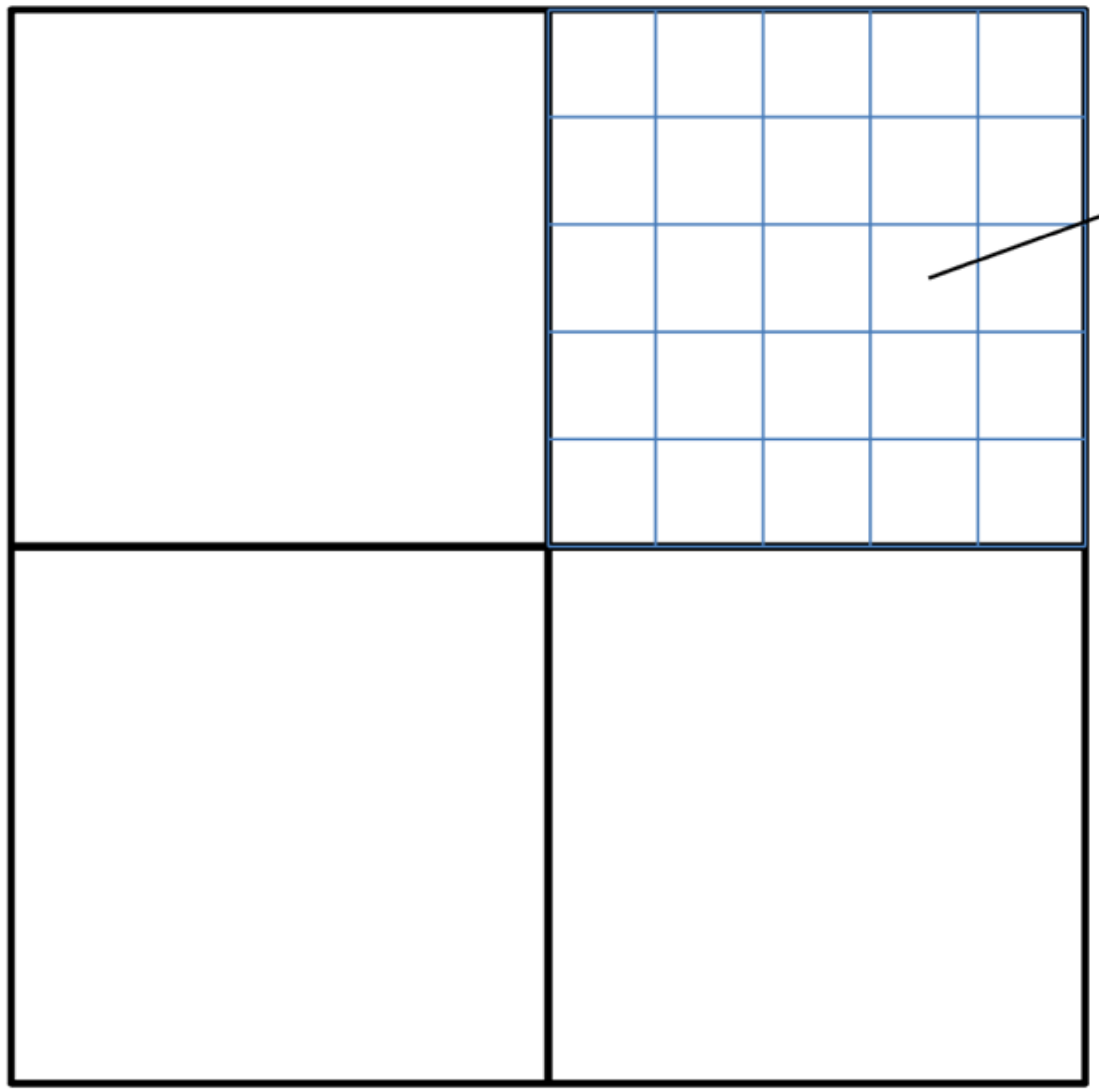


20σ neighborhood
aligned with gradient
σ = characteristic scale

4 x 4 grid

Cunningly shown here
with a 2x2 diagram!

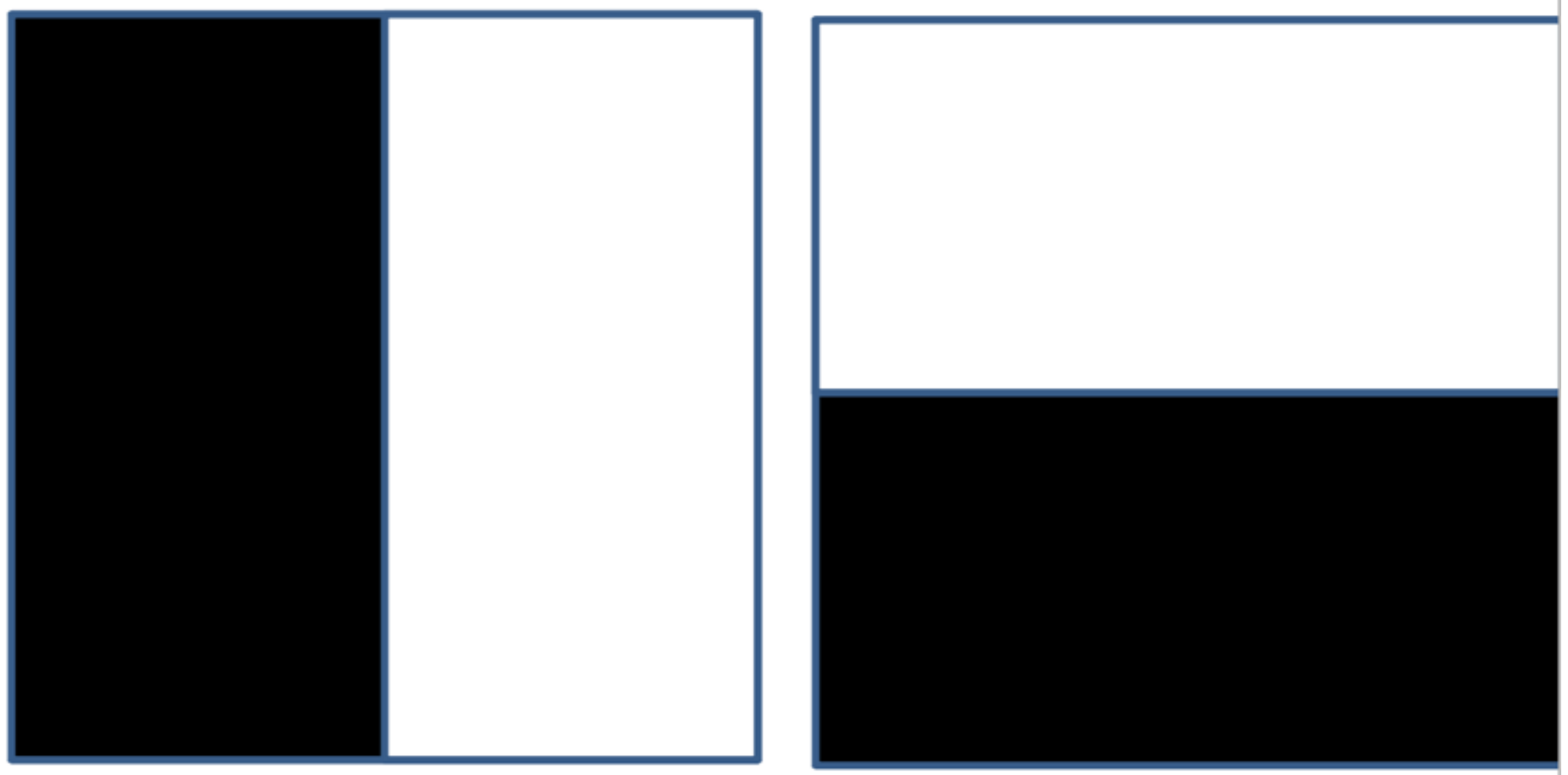
SURF : Speeded Up Robust Features



4 x 4 grid

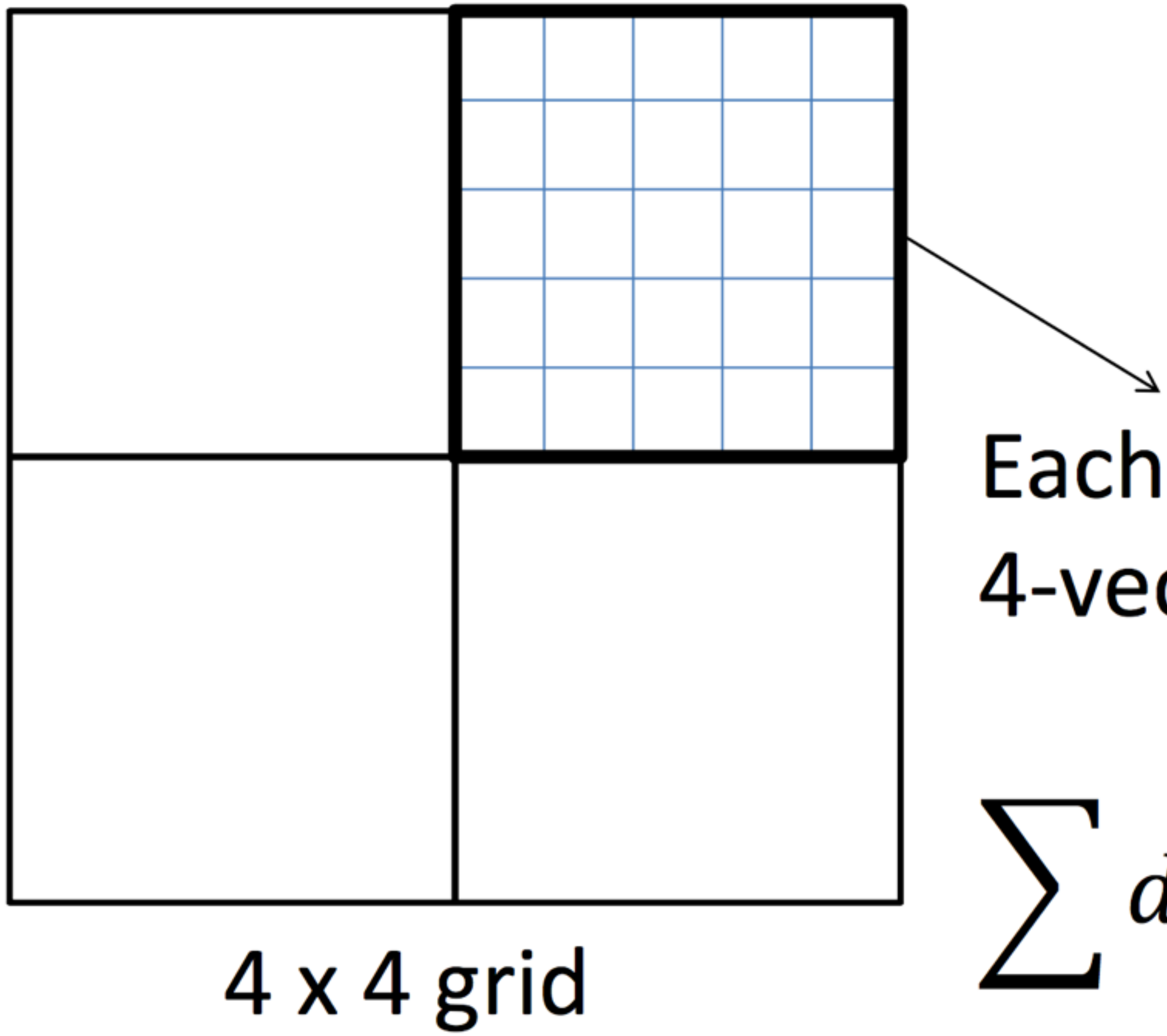
Output of 2 Haar wavelets at 2σ:

$d_x = w$ x $\qquad d_y = w$ x



w = value of Gaussian of scale 3.3σ centered at the interest point

SURF : Speeded Up Robust Features

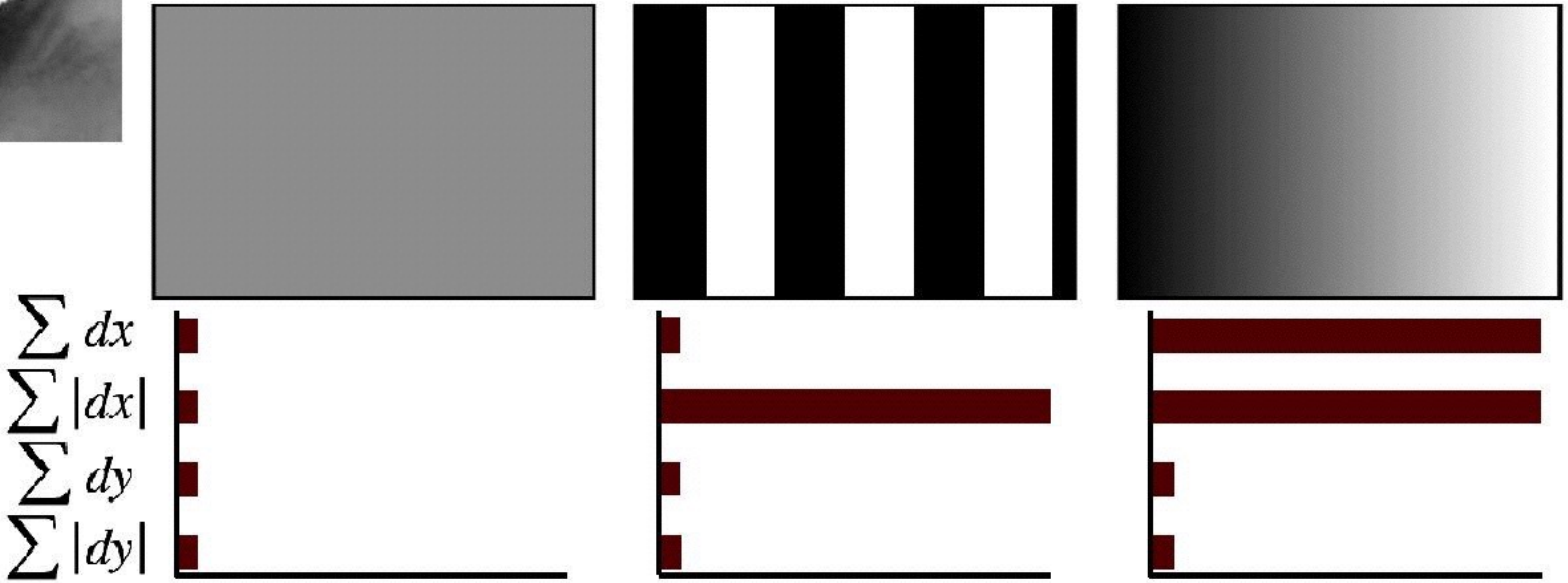


Each cell represented by 4-vector:

$$\sum d_x \sum d_y \sum |d_x| \sum |d_y|$$

Entire descriptor 64 dimensions = 4x4 cells x 4

SURF Features



Outline

- Descriptors as continuos feature vectors
  - Local Gradient Histograms : SIFT
  - Faster Version : Surf
- Descriptor as Binary Code : LBP, BRIEF, BRISK, …
- Descriptors as Densely Sampled Points
- Global Descriptors : GIST

Insight 1: Ordinal representations

28	50	70
5	10	80
3	1	30

Order:

6,3,2,9,1,5,4,7,8

125	154	176
87	98	189
92	85	140

Order:

6,3,2,9,1,5,7,4,8

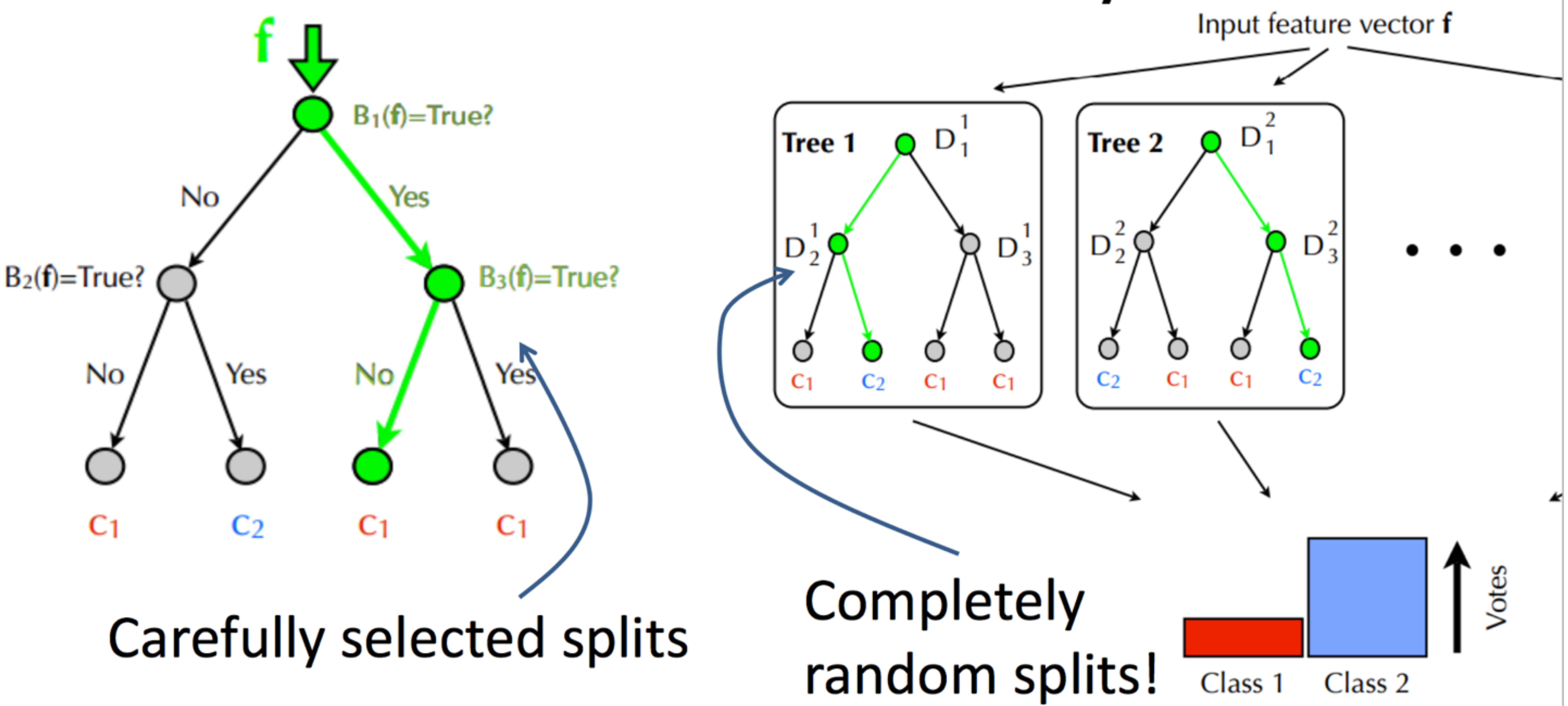
For robustness/generalization, the order of the pixel values is important but not their absolute value (or that of their derivatives)

Perhaps we could just represent the ordering of pixels

1. It's going to be expensive because we need to deal with permutations → combinatorial
2. It's robust but is it discriminative?

Dinkar N. Bhat and Shree K. Nayar. Ordinal Measures for Image Correspondence. PAMI Vol. 20, No. 4, April 1998

Insight 2: Random trees/forests

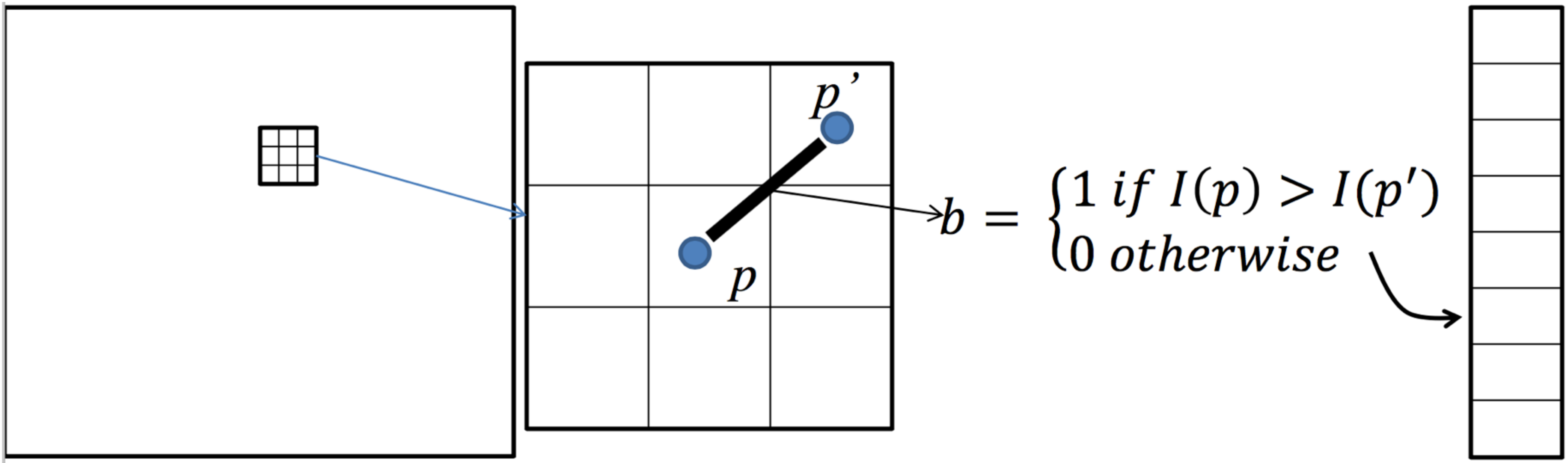


- Maybe we don't need to keep the most complete representation of the ordering
- Maybe we can get away with sampling *a small number of (randomly chosen)* comparisons between pixels (the equivalent of random splits) and keep discrimination power

Ho, Tin (1995) "Random Decision Forests" Int'l Conf. Document Analysis and Recognition
Breiman, Leo (2001) "Random Forests" Machine Learning
Amit, Y. & Geman, D. (1997). Shape quantization and recognition with randomized trees. Neural Computation, 1997

LBP : Local Binary Patterns

- Originally used for texture classification
- Now standard descriptor for vision tasks
- Simplest form:
  - Represent one point by 8-bit binary vector computed from 3x3 neighborhood
  - Represent patch by histogram of 8-bit vectors



LBP : Local Binary Patterns

- Justification (in the context of texture classification):
- Distribution of 3x3patterns

$$p(g_o, g_1, .., g_8)$$

- Relative to center pixel (invariant to illumination bias)

$$p(g_o)p(g_1 - g_o, .., g_8 - g_o) \rightarrow p(g_1 - g_o, .., g_8 - g_o)$$

- Sign only (invariant to changes in greyscale)

$$p(sign(g_1 - g_o), .., sign(g_8 - g_o))$$

g_1	g_2	g_3
g_8	g_o	g_4
g_7	g_6	g_5

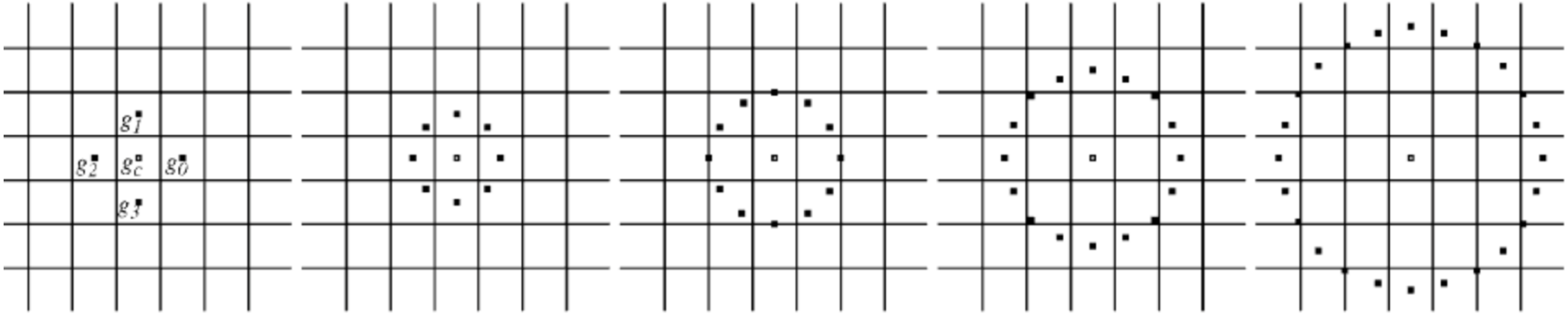
LBP : Local Binary Patterns

- If we can afford more complex descriptors: Multi-resolution LBPs

$$LBP_{P,R}$$

P = Number of sample points

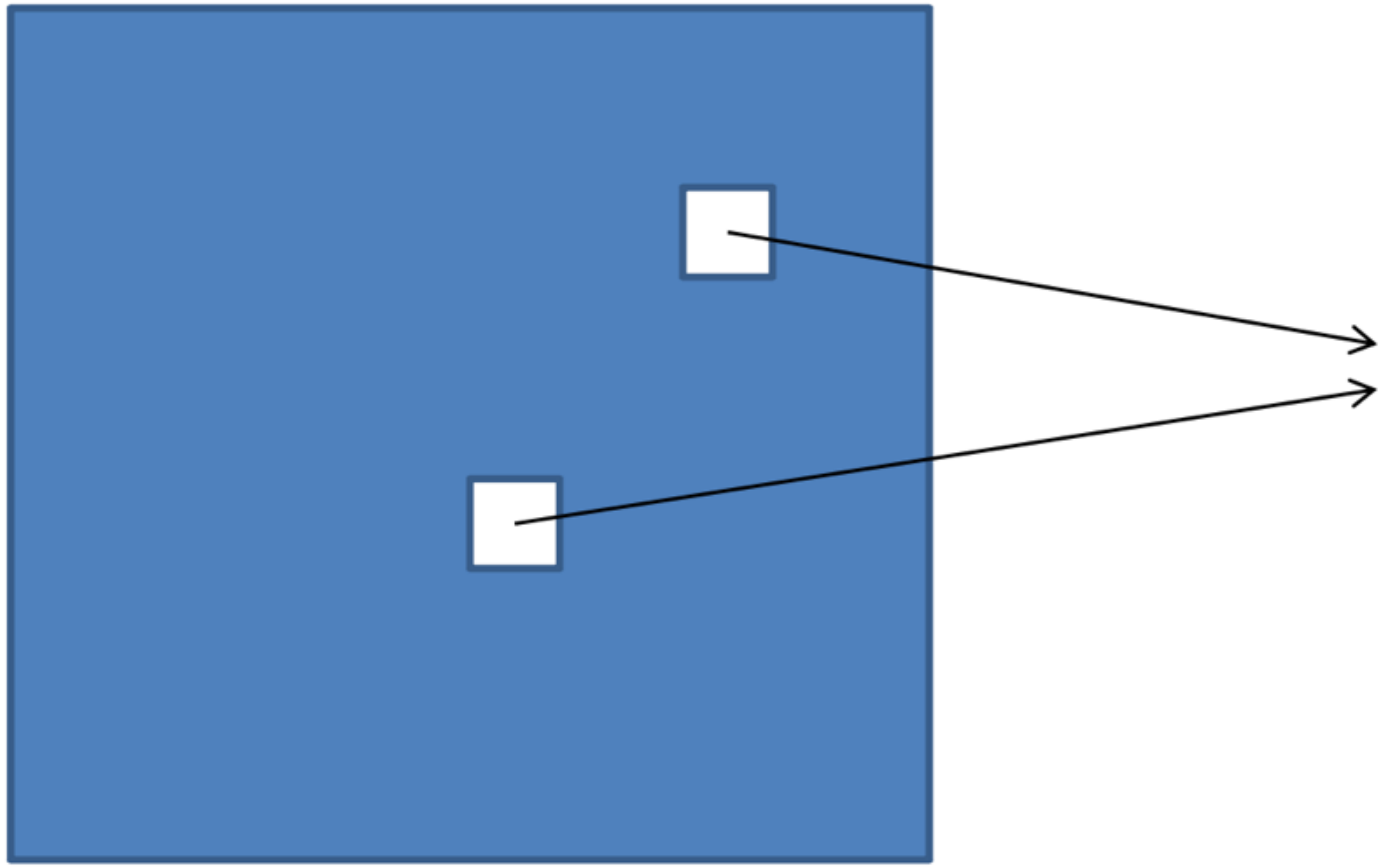
R = Radius



(P=4,R=1.0) (P=8,R=1.0) (P=12,R=1.5) (P=16,R=2.0) (P=24,R=3.0)

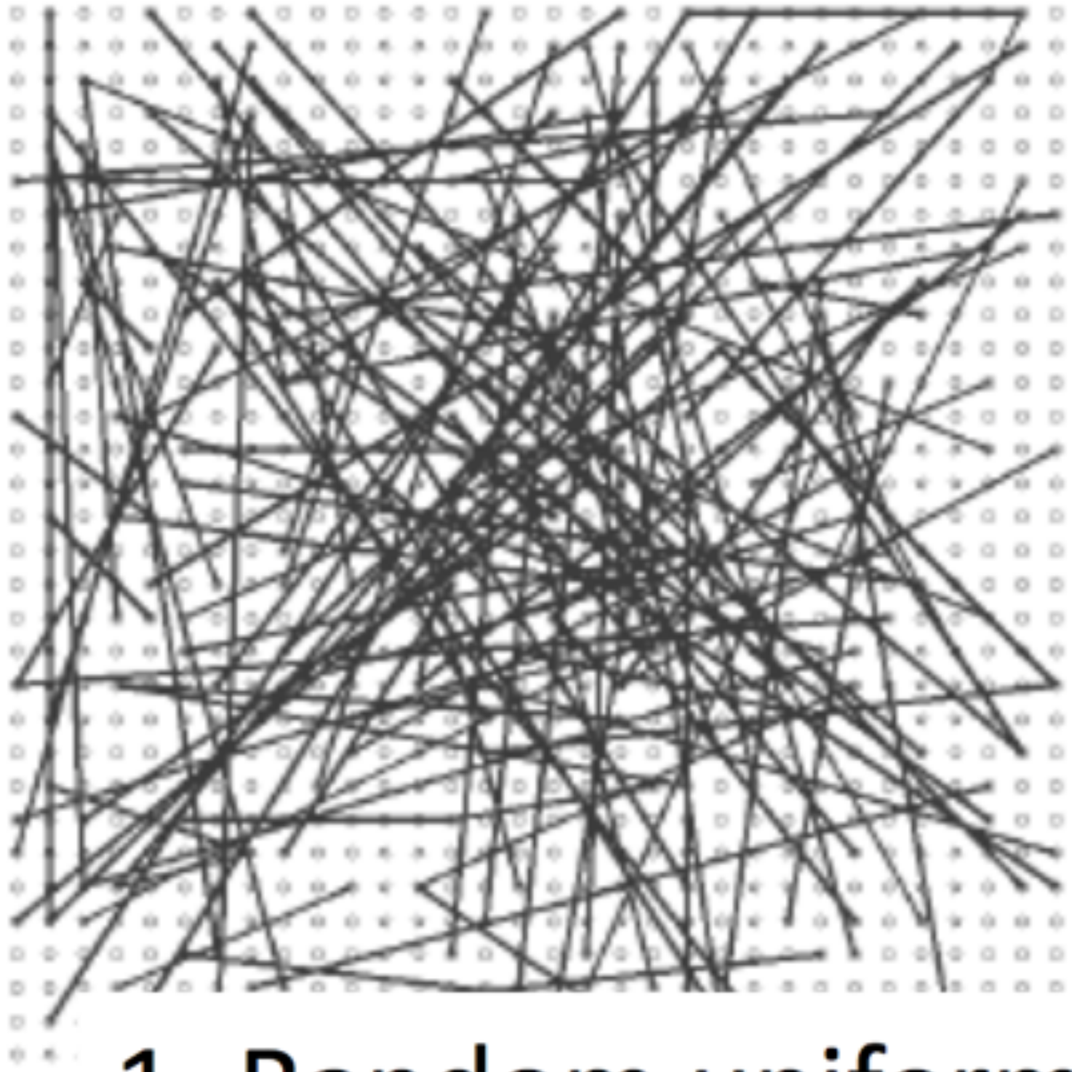
BRIEF : Binary Robust Independent Elementary Features

- Represent S x S patch by a string of bits
- Each bit represent a comparison between two pixels
- Used smoothed (Gaussian) value I(p)
- Typically 128, 256, 512 bits
- Design choice: How to choose the sampling pattern?

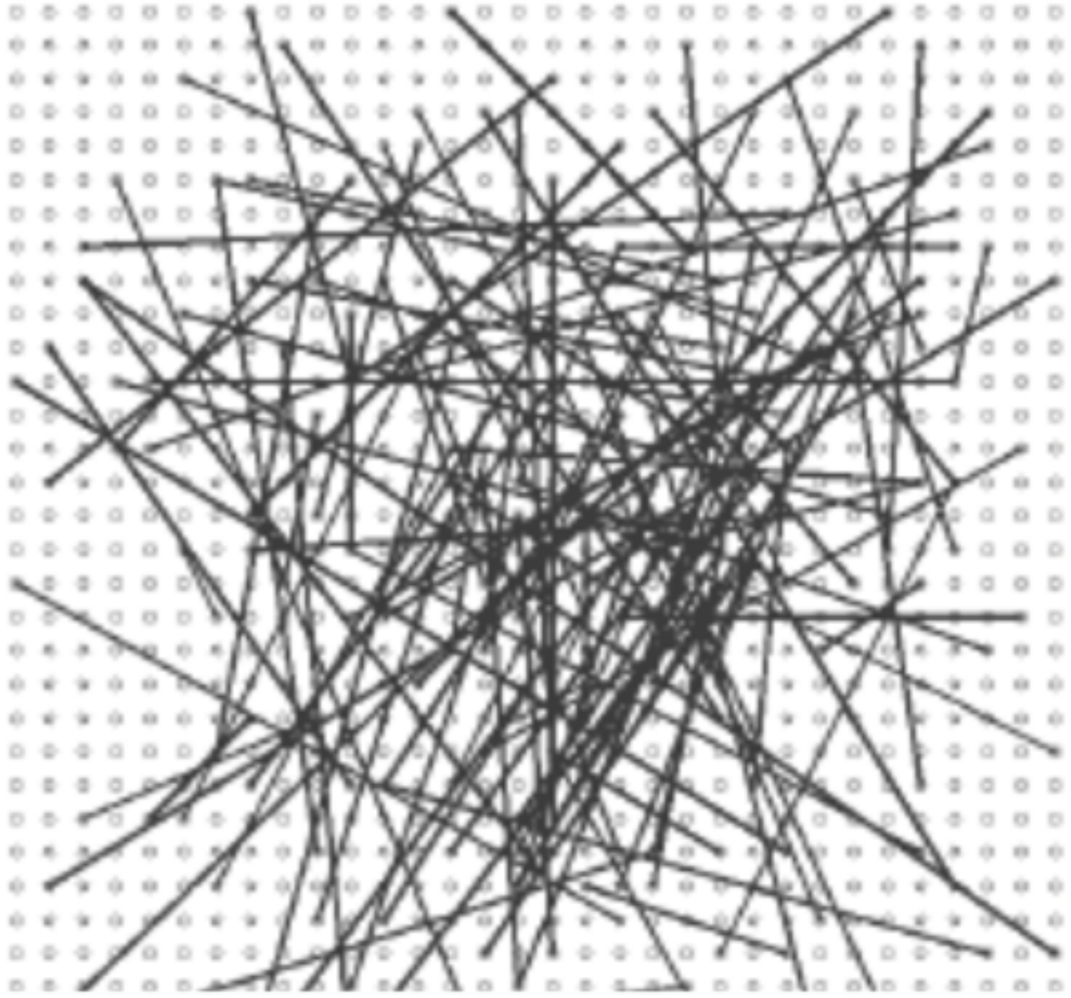


$$b = \begin{cases} 1 \; if \; I(p) > I(p') \\ 0 \; otherwise \end{cases}$$

BRIEF : Binary Robust Independent Elementary Features

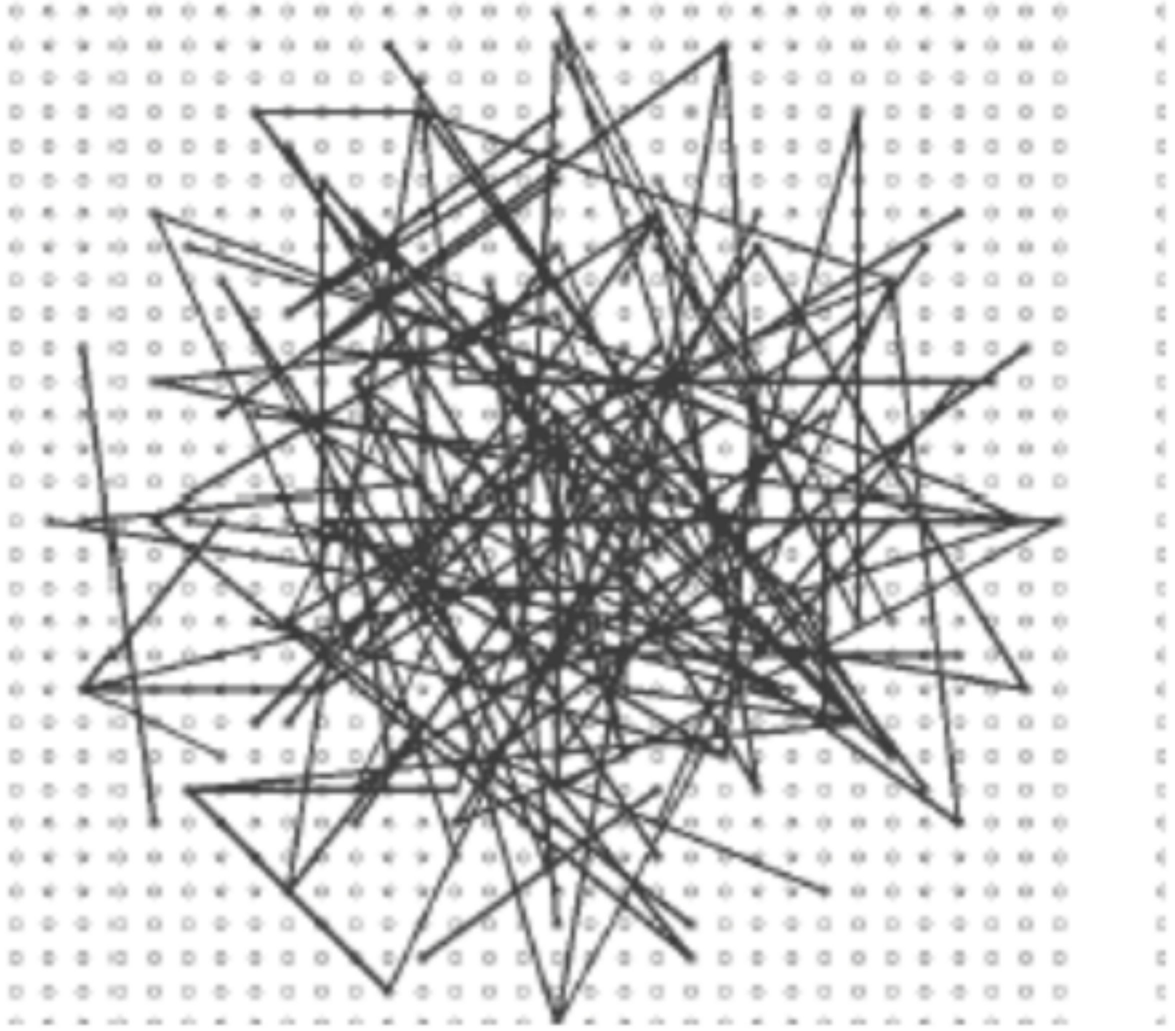


1. Random uniform on x,y grid



2. Normal distribution on p and p'

3. Normal distribution on p' relative to p

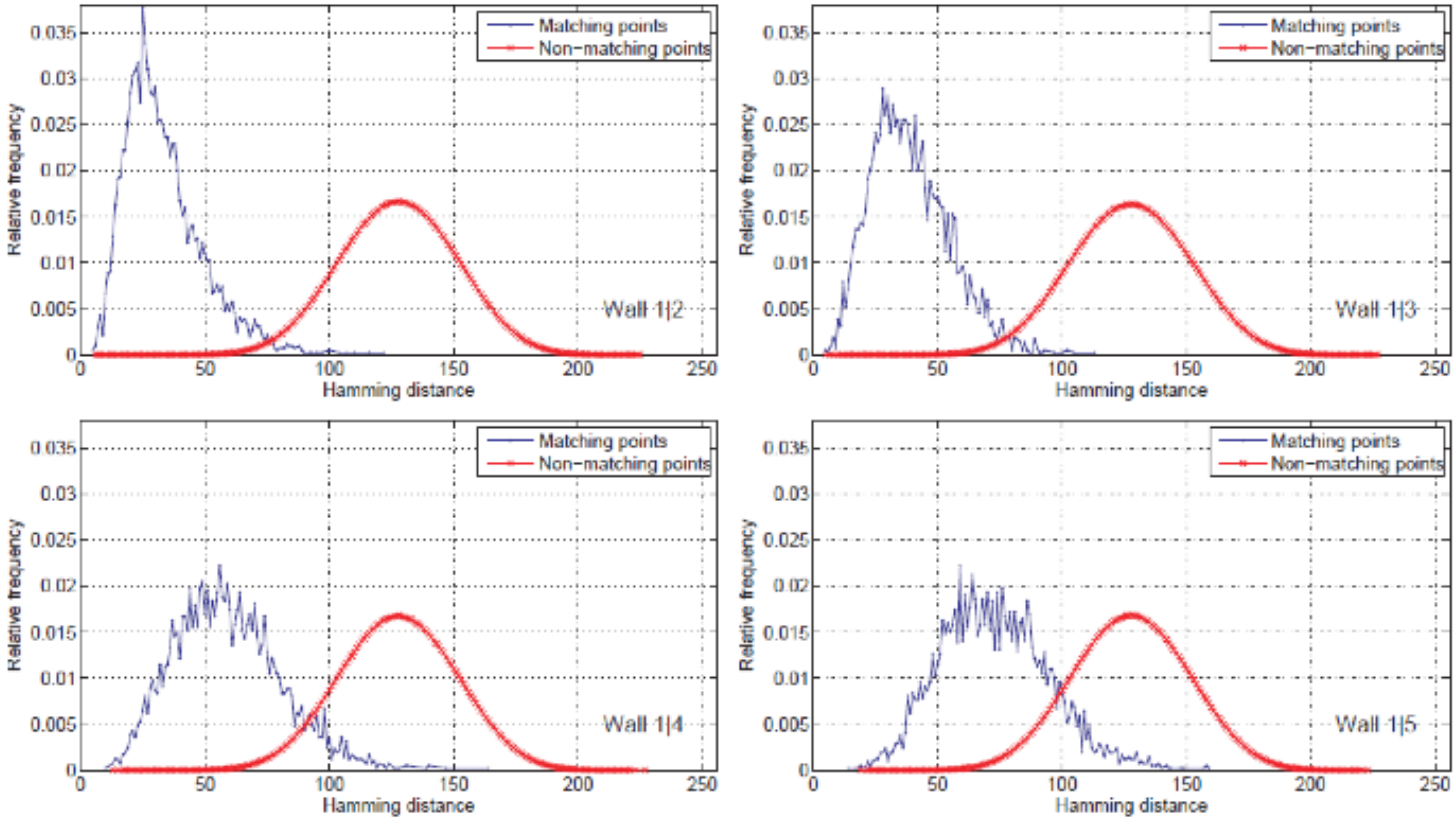


4. Random uniform on polar grid

2. Better in practice

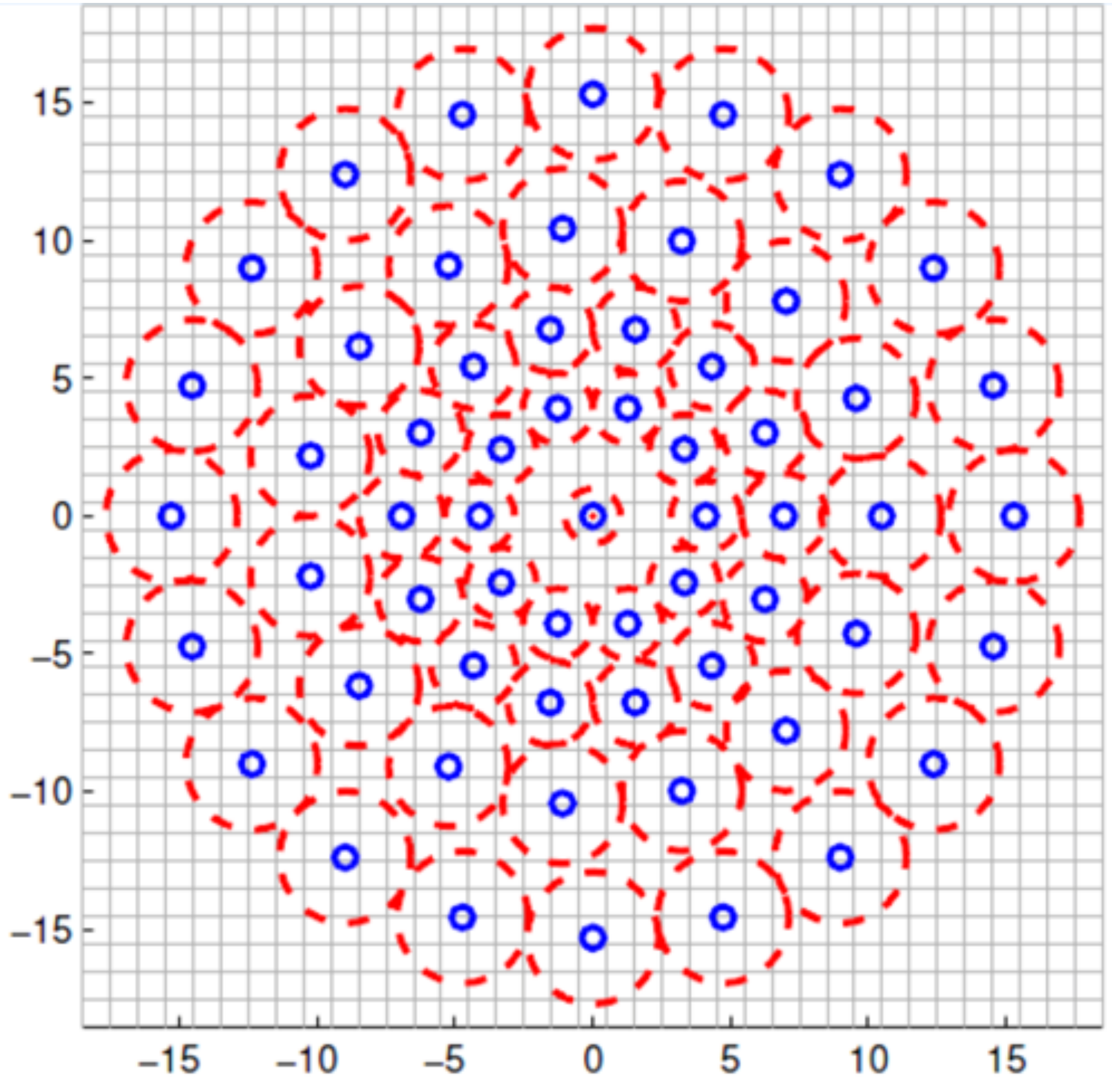
Random selection

Distance Between Descriptors : Hamming Distance



BRISK : Binary Robust Invariant Scalable Keypoints

- Many variations on the BRIEF-like descriptors
- Example BRISK:
  - 60 sample points regularly spaced over window relative to characteristic scale
  - Intensity smoothed by Gaussian proportional to distance to interest point
  - Binary code computed over the 512 pairs such that:



$$\|p_i - p_j\| < 9.75\sigma$$

Many Extensions

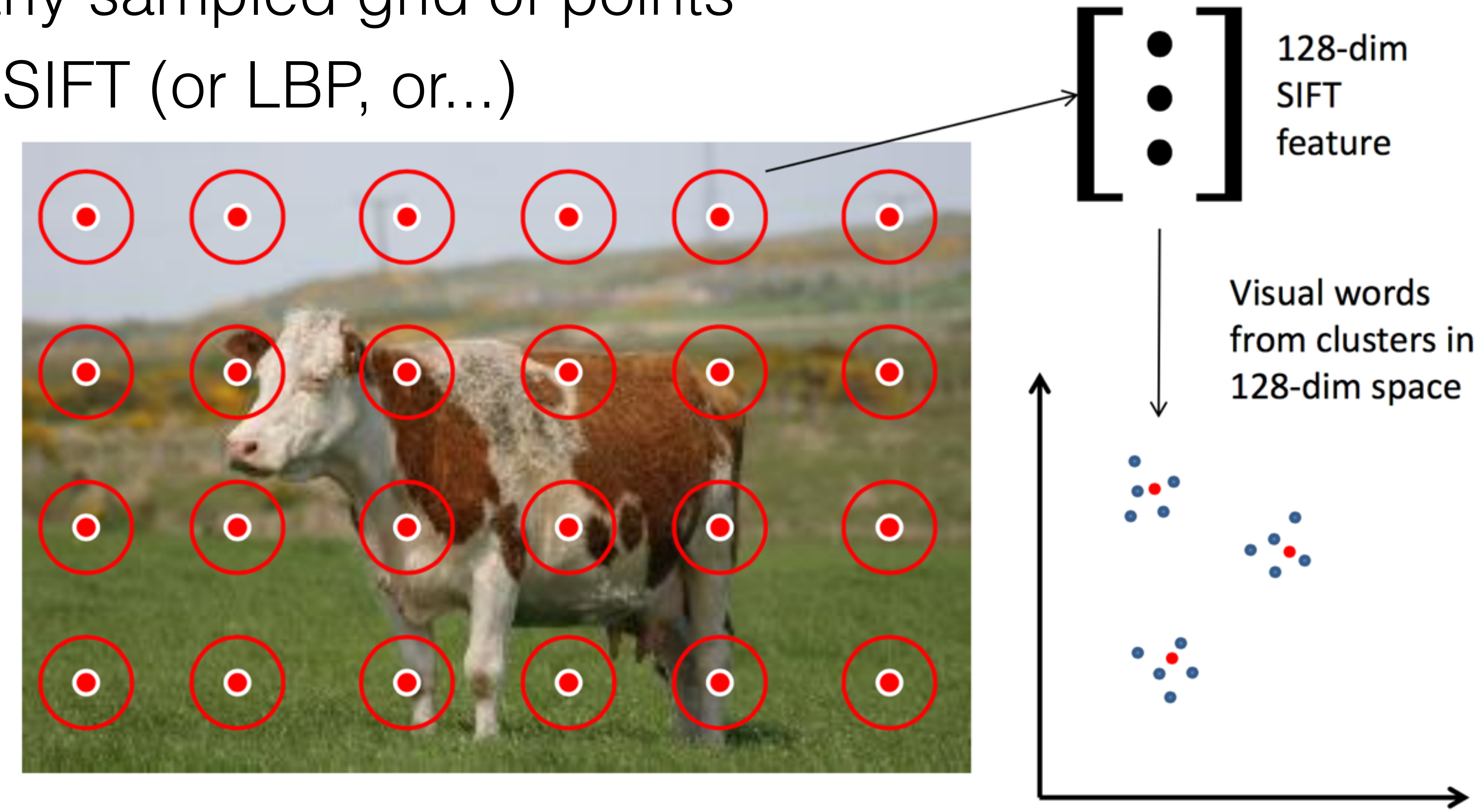
- Learning the patterns of comparisons
- D-BRIEF, ORB, Boost-BRIEF, LUCID....(EPFL, ETHZ, UCSD)
- Better pattern of Gaussian + multi-resolution
- Fast corner detection
- Lot’s of silly names and acronyms

Outline

- Descriptors as continuos feature vectors
  - Local Gradient Histograms : SIFT
  - Faster Version : Surf
- Descriptor as Binary Code : LBP, BRIEF, BRISK, …
- Descriptors as Densely Sampled Points
- Global Descriptors : GIST

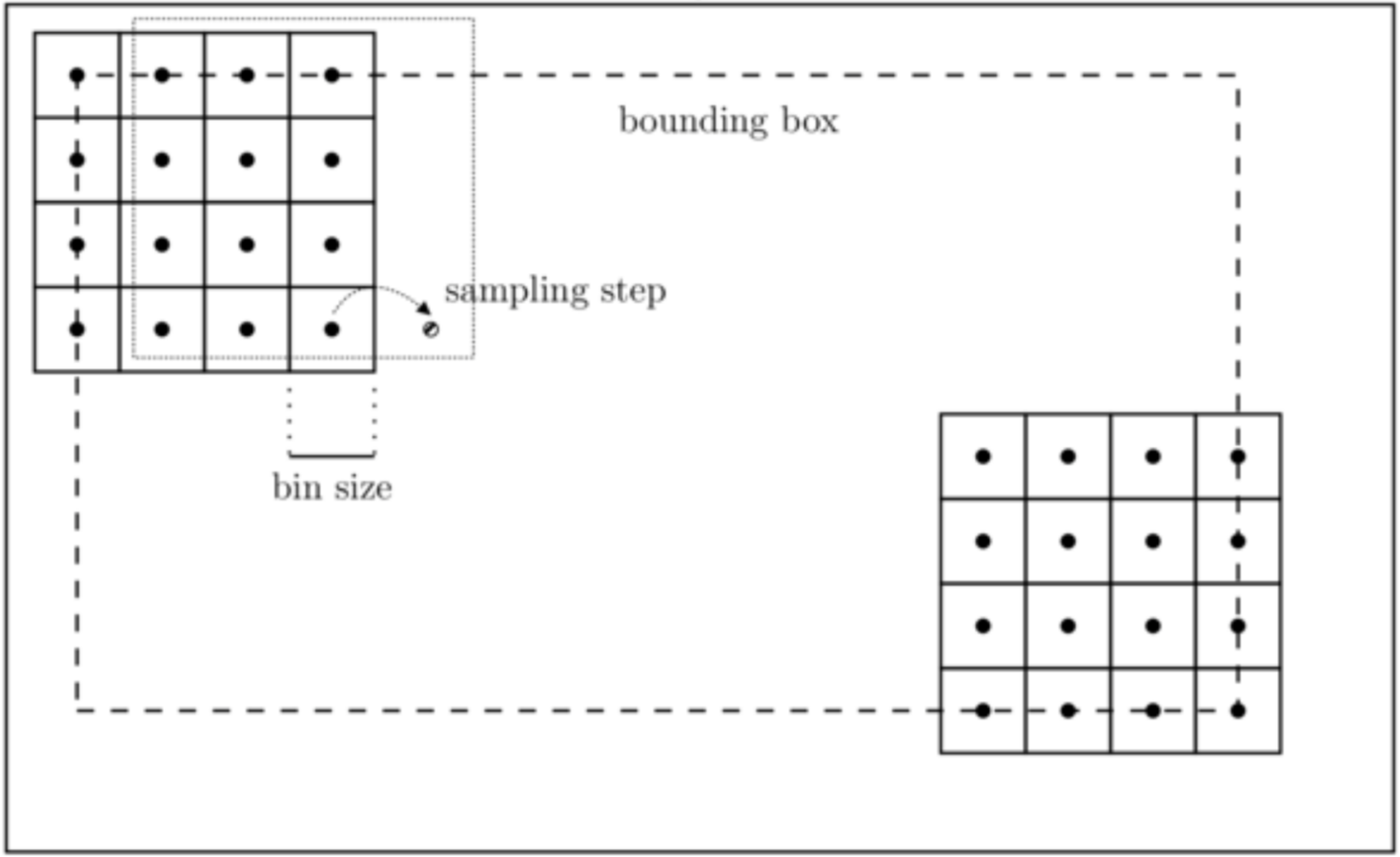
Dense sampling

- So far: Descriptors of patches centered at sparse interest points
- But we can use the descriptors at any point
- Common case:
  - Regularly sampled grid of points
  - Dense SIFT (or LBP, or...)



DSIFT : Dense SIFT

http://www.vlfeat.org/api/dsift.html

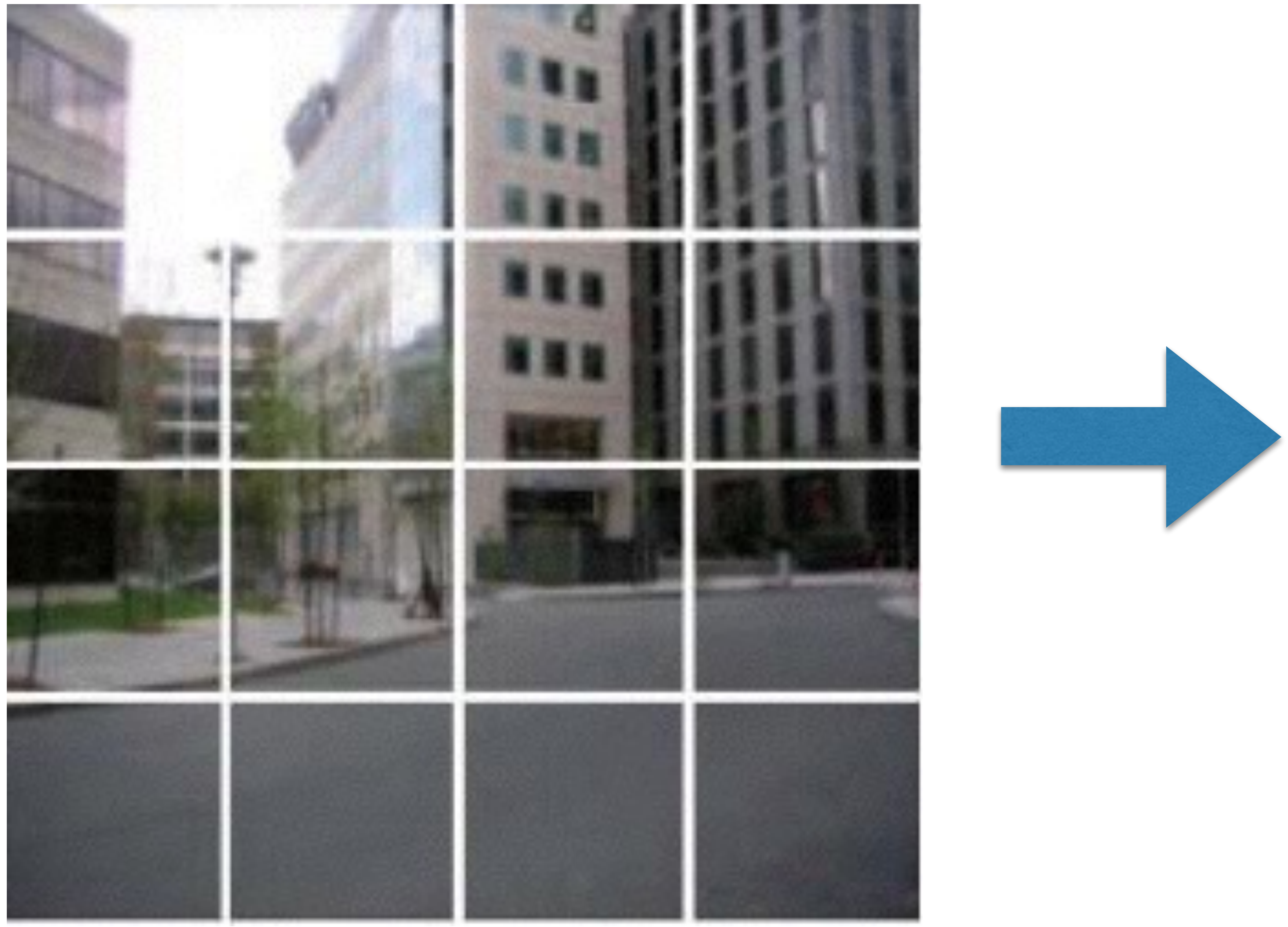


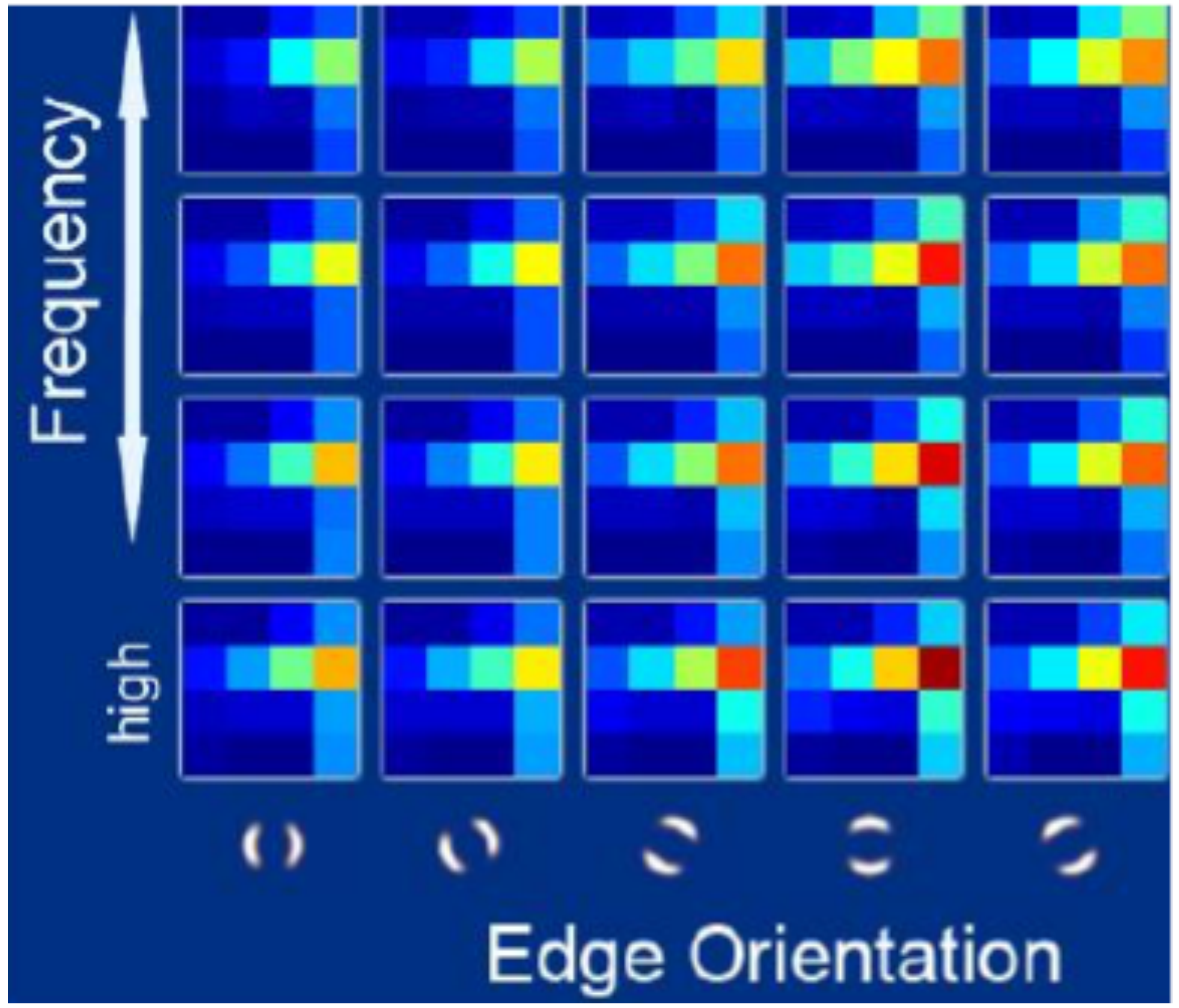
Outline

- Descriptors as continuos feature vectors
  - Local Gradient Histograms : SIFT
  - Faster Version : Surf
- Descriptor as Binary Code : LBP, BRIEF, BRISK, …
- Descriptors as Densely Sampled Points
- Global Descriptors : GIST

GIST

- Extreme case: Descriptor represent the entire image
- Divide the image in 4 bins and compute average power at different frequencies (scale) and orientations





4 cells x 8 orientations x 16 bins = 512 dimensions

GIST Example : Image Matching

Query Image

Matches

Resources

- BRIEF, (D-BRIEF, binboost, LBGM, ...):
  - Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua . BRIEF: Binary Robust Independent Elementary Features, Proc. ECCV 2010.
  - http://cvlabwww.epfl.ch/~lepetit/
- BRISK:

– Stefan Leutenegger, Margarita Chli and Roland Y. Siegwart. BRISK: Binary Robust Invariant Scalable Keypoints. Proc. ICCV 2011.

- GIST:
  - Aude Oliva, Antonio Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. International Journal of Computer Vision, Vol. 42(3): 145-175, 2001.
  - http://people.csail.mit.edu/torralba/code/spatialenvelope/
- K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, T. Kadir and L. Van Gool, A Comparison of Affine Region Detectors; International Journal of Computer Vision (IJCV), Volume 65, Number 1, 2005.
- Major resource for all the features (Vision Lab Features Library (VLFeat)): http://www.vlfeat.org/api/index.html

Resources

- SIFT:
  - David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. IJCV (International Journal of Computer Vision), 2004.
  - http://www.cs.ubc.ca/~lowe/keypoints/ • MSER:

– P-E. Forssen, P-E. and D. Lowe, Shape Descriptors for Maximally Stable Extremal Regions International Conference on Computer Vision (ICCV), 2007.

- SURF:
  - Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, SURF: Speeded Up Robust Features, Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, 2008.
  - http://www.vision.ee.ethz.ch/~surf/
  - DAISY: Engin Tola, Vincent Lepetit, Pascal Fua, DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2010.
- LBP:
  - Timo Ojala, Matti Pietikainen, and Topi Maenpa. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 24, No. 7, July 2002
  - http://www.cse.oulu.fi/CMV/Research/LBP