SURF: Speeded-Up Robust Features

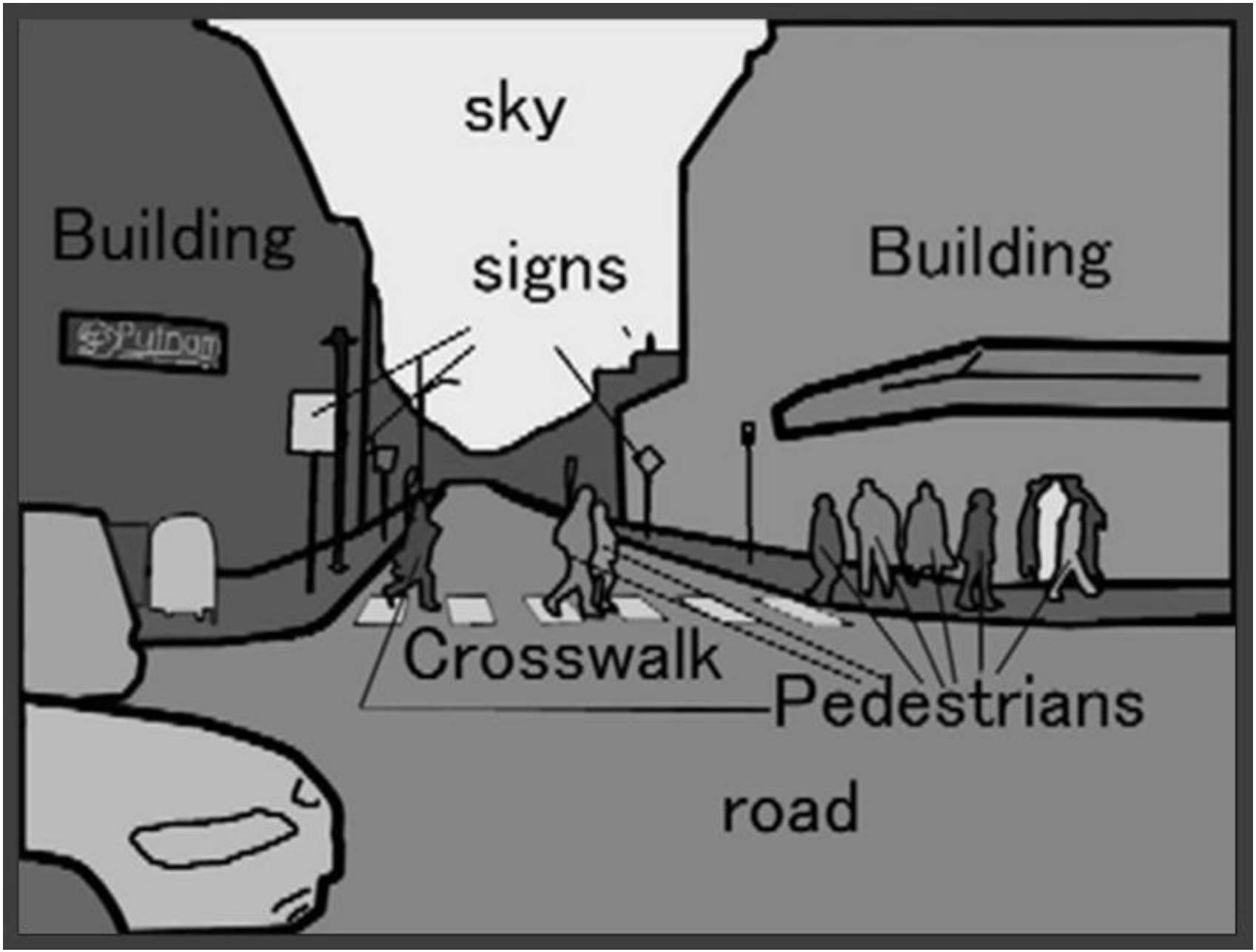
Castleberry, Cherry, and Firth

September 27, 2012

Motivation

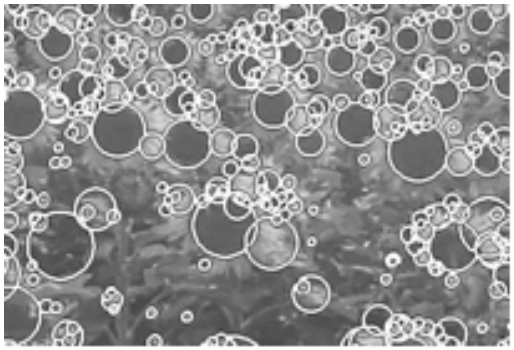
- The motivation is to facilitate advances in:
  - Image registration
  - Camera calibration
  - Object recognition
  - Image retrieval


Camera Calibration


sky
Building
signs
Building
Crosswalk
Pedestrians
road

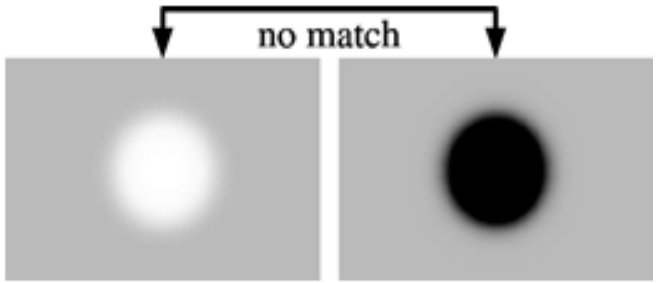
Problem Definition

- The problem is to efficiently and accurately find point correspondences between two images depicting the same scene, thereby enabling camera calibration and object recognition.
- The problem solution is subdivided into three stages:
  - **Detection**: identify points of interest. The most important aspect of a detector is its repeatability.
  - **Description**: create a vector which holds data about the feature(s). It should be simple (low-dimensional) to facilitate efficient matching but complex enough to adequately describe the feature.
  - **Matching**: match the feature vectors across images. The matching is based on a distance measure between the two feature vectors (such as the ▸ Mahalanobis or Euclidean distance).



Detect

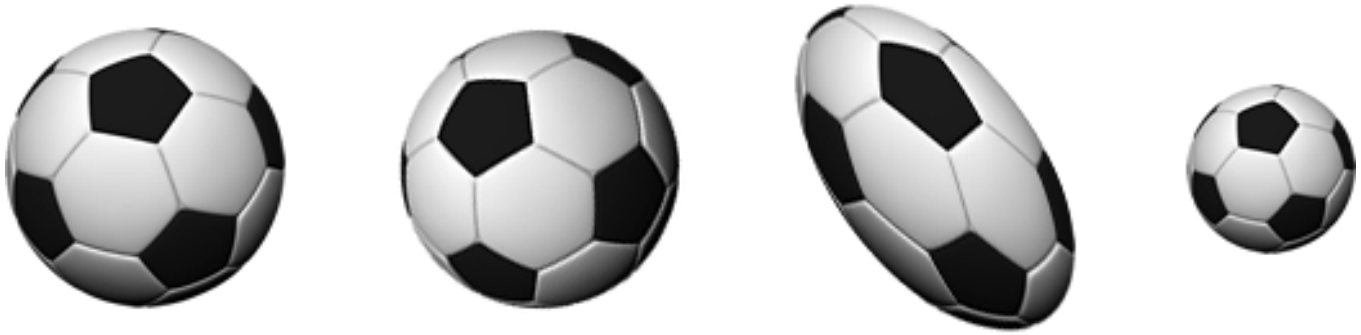
Describe



Match

Problem Definition

- The goal is to develop a detector and descriptor which, in comparison to the state-of-the-art detectors and descriptors of the day, are computationally inexpensive but do not sacrifice performance (accuracy of matches).
- The focus is on scale and in-plane rotation invariant detectors and descriptors. The descriptor is robust enough to handle skew, anisotropic scaling (stretching), and perspective effects.
- The handling of photometric deformations is limited to bias (offset, or brightness changes) and contrast changes (by a scale factor).



Geometric Image Deformations

e.g. scaling, translating, rotating, skewing, etc.

Photometric Image Deformations

e.g. lighting changes

Interest Point Detection

- Harris corner detector
  - Uses eigenvalues of second moment matrix
  - Not scale invariant
- Lindeburg
  - Introduced concept of automatic scale selection
  - Experimented with the determinant of the Hessian matrix and the Laplacian
- Mikolajczyk and Schmid
  - Harris-Laplace or Hessian-Laplace
  - Scale invariant feature detection with high repeatability
  - Used determinant of the Hessian matrix to select location, and Laplacian to select scale

Interest Point Detection

- Lowe
  - Used a Difference of Gaussians filter to approximate a Laplacian of Gaussians
- Conclusions from previous work on interest point detection: Hessian-based detectors are more stable and repeatable than their Harris-based counterparts. Also, approximations such as DoG provide good speed with minimal loss in accuracy.

Interest Point Description

- Many interest point description techniques exist, including: Gaussian derivatives, moment invariants, complex features, steerable filters, phase-based local features...
- Lowe (SIFT)
  - Computes a histogram of local oriented gradients around the interest point and stores the bins in a 128-dimensional vector
- Ke and Sukthankar (PCA-SIFT)
  - Apply PCA to the gradient image around the interest point
  - 36-dimensional descriptor vector is faster in matching but less distinctive than SIFT
  - Also proposes GLOH, but is similarly computationally expensive due to it's use of PCA
- Grabner
  - Used integral images to approximate SIFT

Background

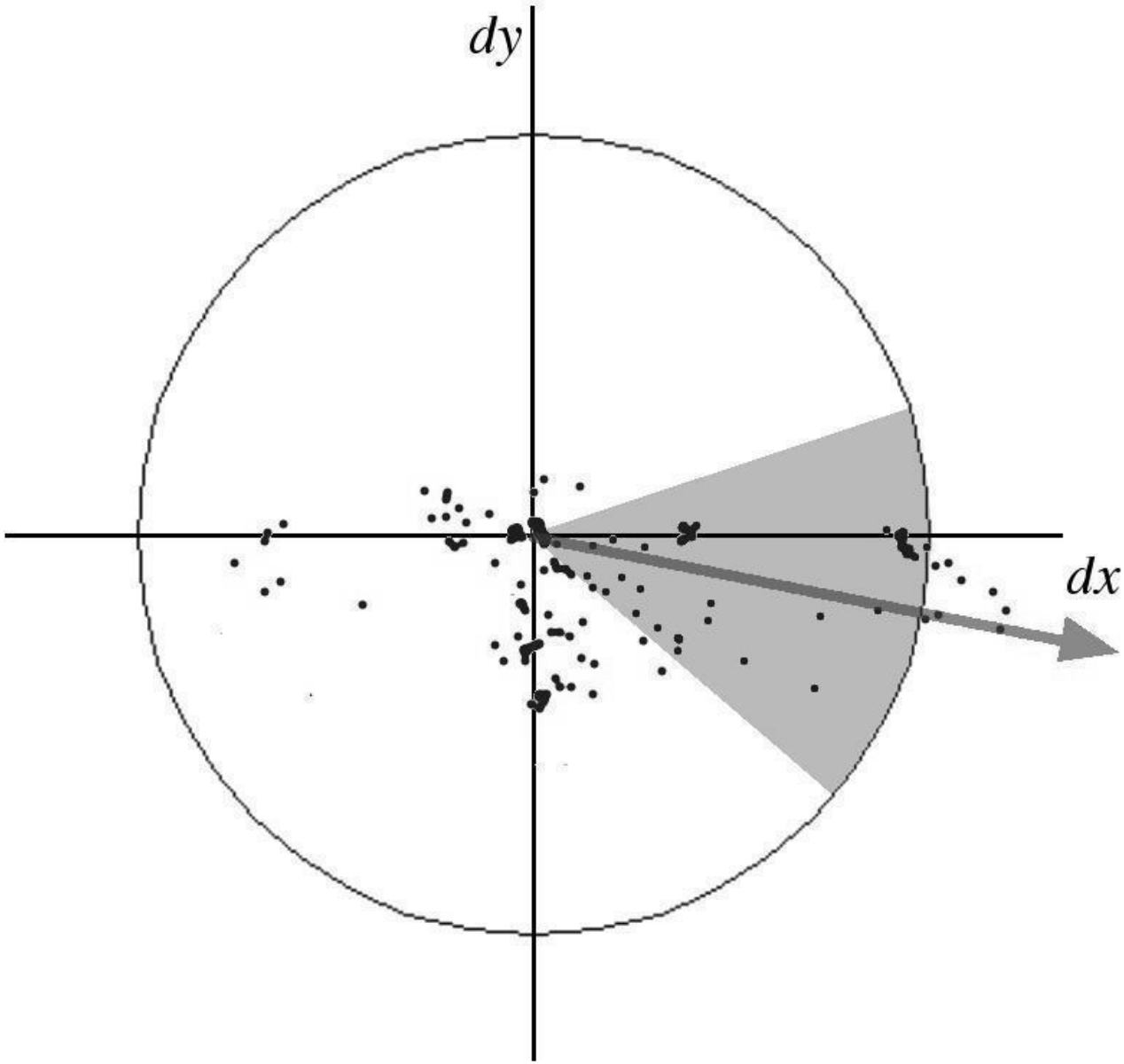
- **Detection** of interest points is done through approximations of the ▸ Laplacian of ▸ Gaussians, then finding extrema within the scale space of the image.
- **Description** is handled by assigning orientation vectors using ▸ Haar wavelets over a 4x4 grid. Four values $(d_x, d_y, |d_x|, |d_y|)$ are stored for each cell, yielding a 64-dimension description vector.
- **Matching** is facilitated by indexing the results with the sign of the Laplacian, which indicates if the blob is block-on-white or white-on-black. The nearest-neighbor ratio matching is used.

Blob Detection

- To summarize the method: an ▸ integral image is first calculated on the image I(x,y), which facilitates the subsequent approximation of the ▸ determinant of the ▸ Hessian matrix for the image (x, y) over its scale space. The scale space is constructed not by taking ▸ Gaussians of increasing scales and downsampling as in SIFT, but instead by ▸ convolving with the image box filters (of increasing size) which approximate the ▸ Laplacian of Gaussians. Extrema of the Hessian determinants found within the **octaves** constituting the scale space of the image indicate blob responses.

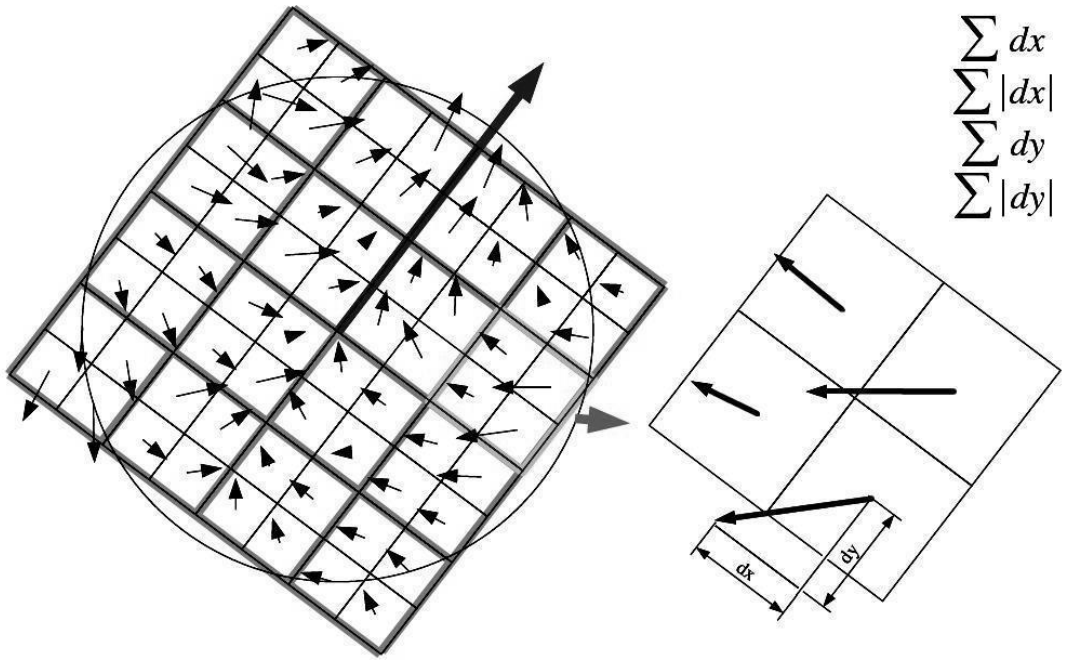
Orientation Assignment

- The ▸ Haar wavelet responses for each point within a neighboorhood of $6s$ (where s is the image scale) are calculated by convolving a Haar wavelet filter of $4s$ over the image. The wavelet responses are weighted with a Gaussian ($\sigma = 2s$) at the center of the interest point. Haar responses (x_h, y_h) within a circle of $6s$ around the interest point are graphed. Haar responses within a window of θ to $\theta + \frac{\pi}{3}$ are summed for θ from 0 to 2π to form an orientation vector for that value of θ. The maximum of these Haar response vectors is taken to give the dominant orientation for the interest point.


dy
dx

Description Vector

- Square regions of size $20s$ are laid over the interest points along the orientation of the square. These squares are divided into $4x4$ regions, each of which is broken up into a $5x5$ region. Haar wavelet responses (d_x, d_y) are calculated for each of these cells. Then, the sums $\sum d_x, \sum d_y, \sum |d_x|, and \sum |d_y|$ are calculated and assigned to each region in the $4x4$ grid. These form a 64-dimension descriptor vector for the interest point.

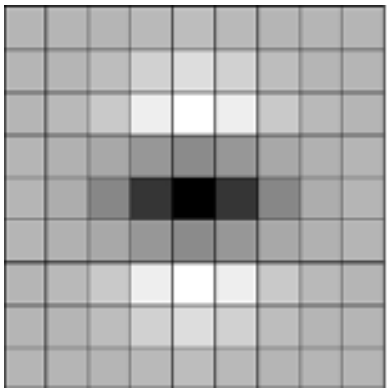
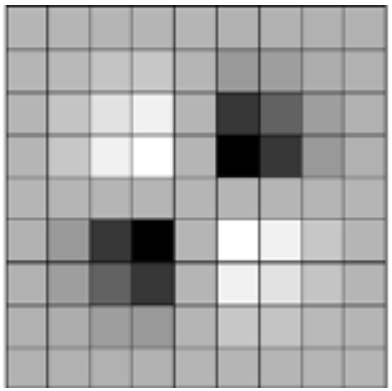
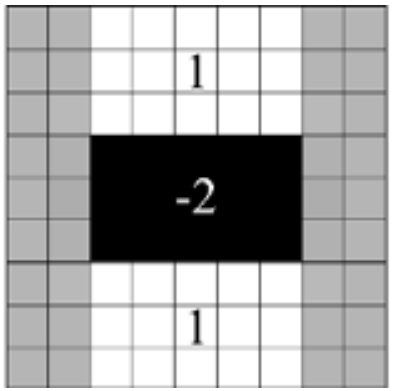
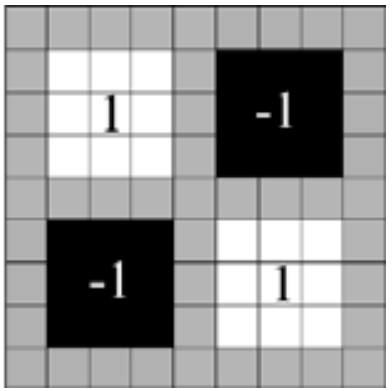

$\sum dx$
$\sum |dx|$
$\sum dy$
$\sum |dy|$
dx
dy

Hessian Approximation

- A ▸ Hessian Matrix is approximated using box filters. The box filters are approximations of second-order derivatives of Gaussians within a rectangular region. These approximations are efficiently computed using ▸ integral images.
- The determinant of the Hessian matrix is approximated using the box filters:

$$det_{approx}(\mathcal{H}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (1)$$

- where w is a weight needed to adjust for the difference between the approximated and actual Gaussian.




1
-2
1

1
-1
-1
1

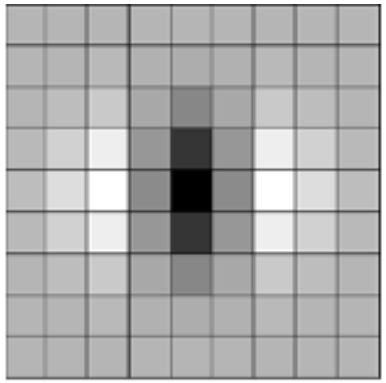
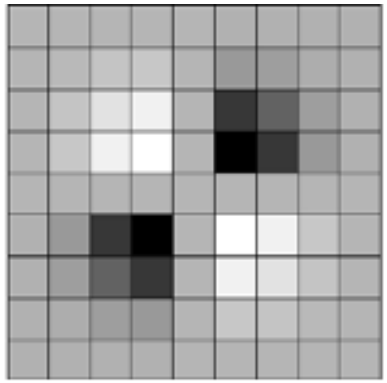
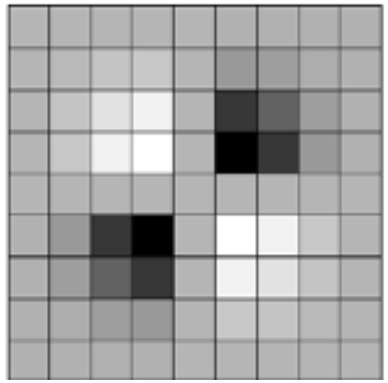
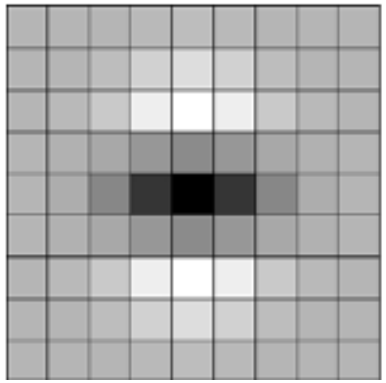
Hessian Determinant Approximation

- w for an nxn box filter approximating a Gaussian with σ is equal to:

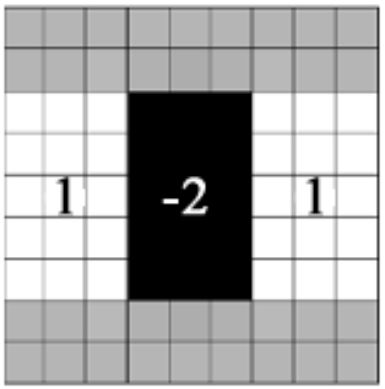
$$\frac{|L_{xy}(\sigma)|_F / |L_{yy}(\sigma)|_F}{|D_{xy}(n)|_F / |D_{yy}(n)|_F} \tag{2}$$

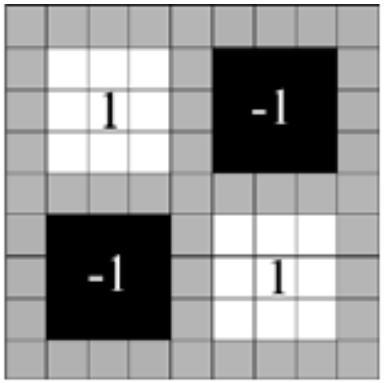
  where $|x|_F$ is the ▸ Frobenius norm. This factor changes with filter size; however, it is desirable to keep it constant. Therefore the filter responses (D) are normalized with respect to their size to guarantee a constant Frobenius norm.
- The det_{approx} represents a *blob response* in I at $\mathbf{x}$. det_{approx} for all locations $\mathbf{x}$ in the image gives a *blob response map*. Local maxima are detected to give the locations of blobs.

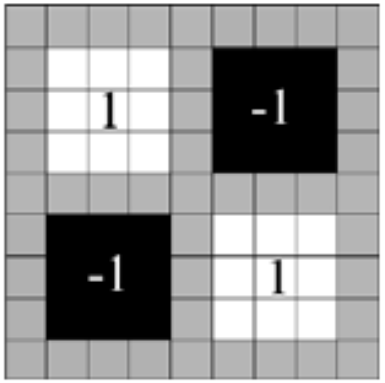
$$H = \left[\quad \right]$$

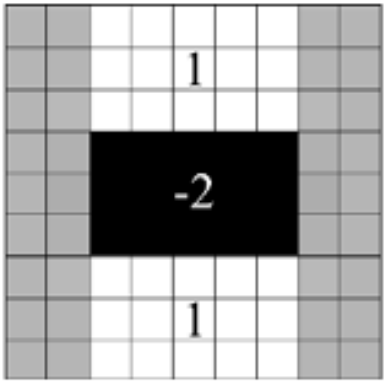





$$\mathcal{H} = \left[\quad \right]$$





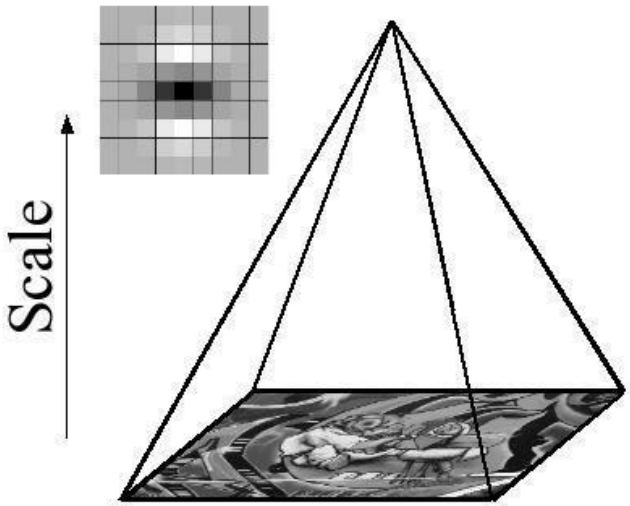


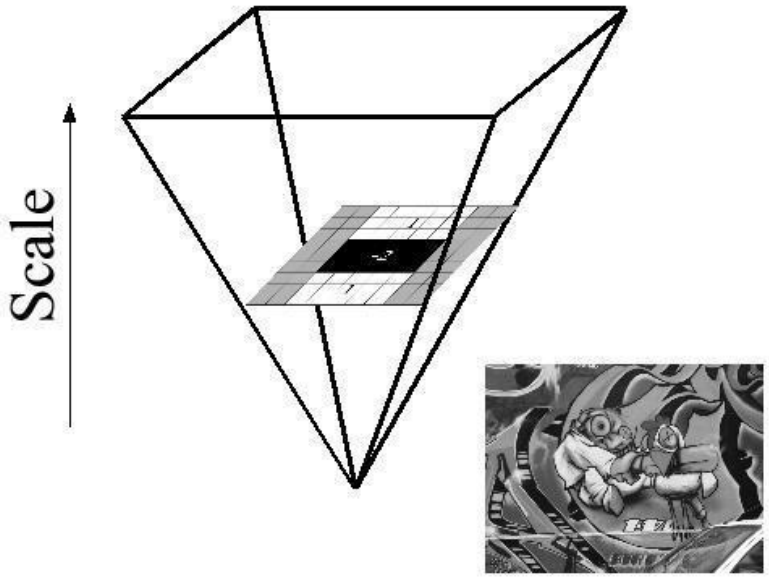


$$\det(\hat{H}) = \boxed{1\ \text{-2}\ 1} * \boxed{\begin{matrix}1\\ \text{-2}\\ 1\end{matrix}} - (w * \boxed{\begin{matrix}1 & \text{-1}\\ \text{-1} & 1\end{matrix}})^2$$

Scale Spaces

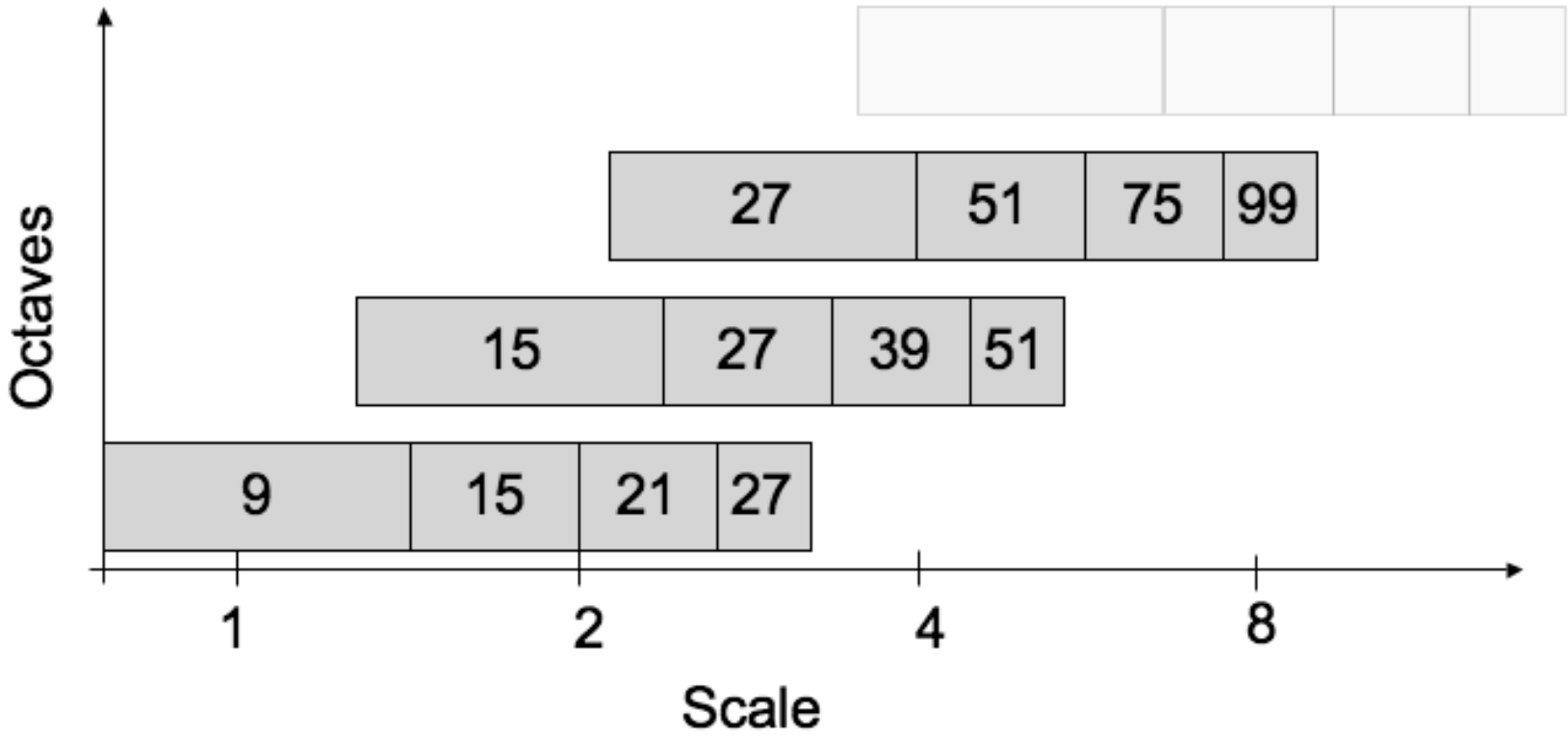
- Interest points should be found at different scales. To represent the image at different scales, a **scale pyramid** is used.
- Rather than iteratively reducing the size of the image, the box filters are upscaled and computed, for which there is little additional computational cost. As a side effect of not downsampling the image, there is no **aliasing**.
- A scale space is divided into **octaves**.


Scale


Scale

Octaves

- An octave represents a series of filter response maps obtained by ▸ convolving the same input image with a filter of increasing size.
- The octave encompasses a scaling factor of 2. The pixel difference between scales of the image is at least one-third of the filter size (which is the size of the lobes in D_{xx} or D_{yy}). For odd-n filter sizes, a minimum of 2 pixels is required to guarantee a central pixel. In the case of a filter of size 9, this amounts to a difference of 6.


Octaves
27
51
75
99
15
27
39
51
9
15
21
27
1
2
4
8
Scale

Scale Interpolation

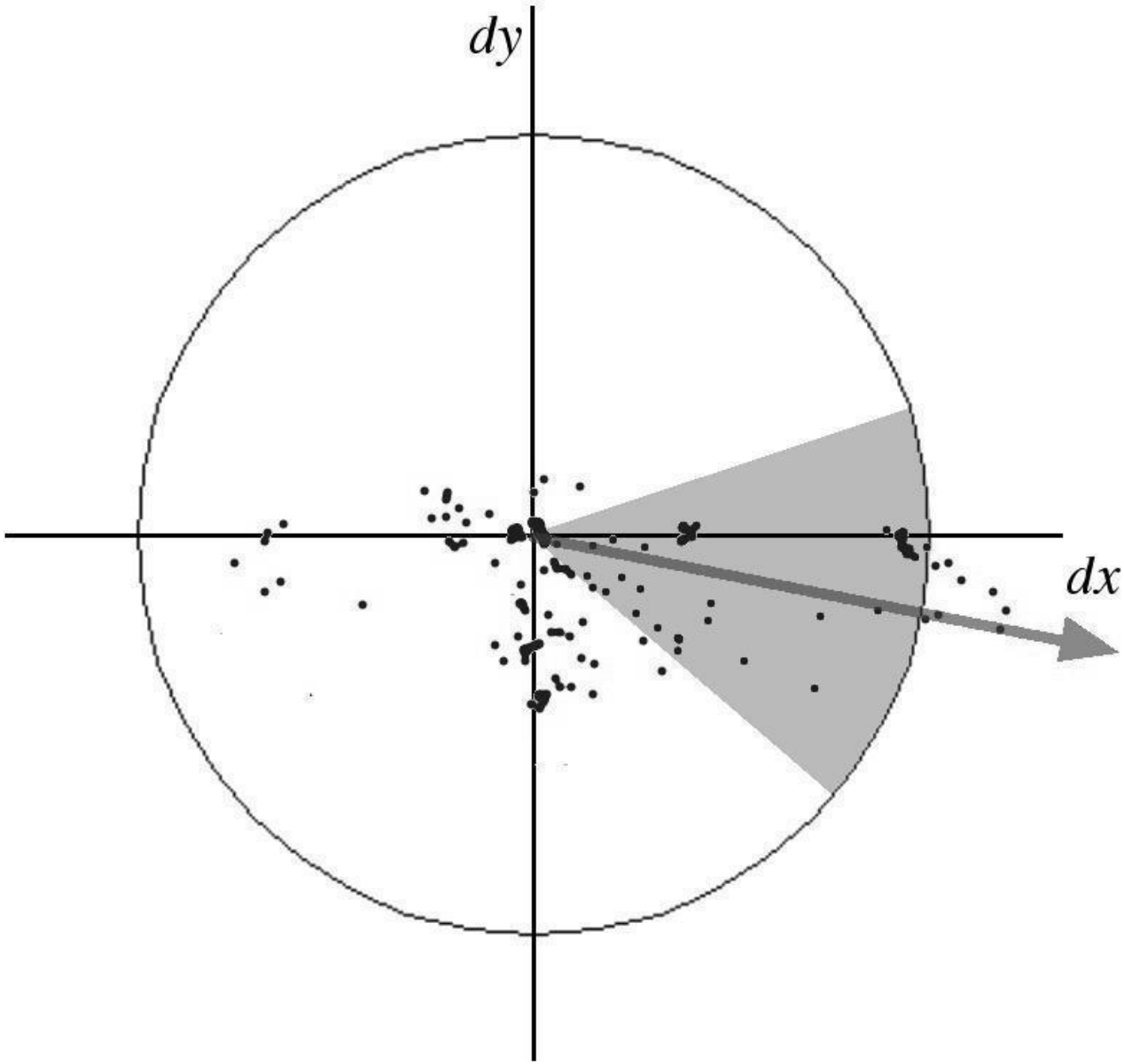
- To localize interest points in the image over scales, non-maximum suppression in a 3x3x3 neighboorhood is applied (Neubeck and Van Gool).
- The maxima of the determinant of the Hessian matrix are then interpolated in scale and image space (Brown et al).

Interest Point Description

- Similar to SIFT, the SURF describes the distribution of the intensity within the interest point neighoorhood, but with first-order ▸ Haar wavelet responses in the x and y dimensions rather than the gradient.
- Also, integral images are exploited for efficiency, and only 64 dimensions are used.

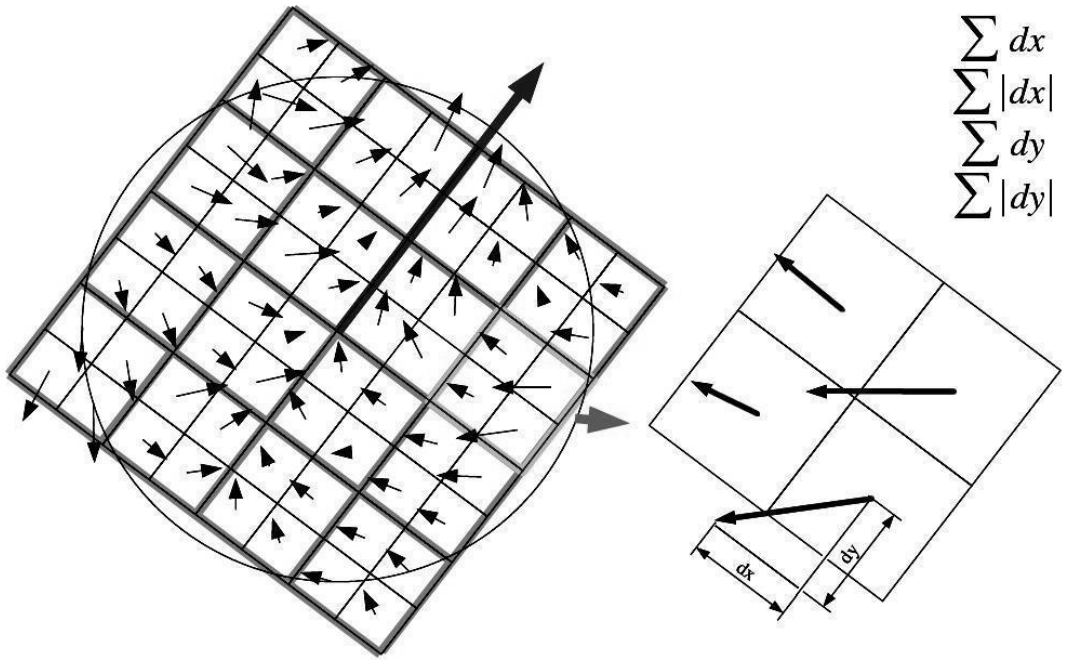
Orientation Assignment

- For the interest points to be rotation-invariant, the orientation must be reproducible. Haar wavelet responses are calculated in the x and y directions within a circular neighboorhood of radius $6s$, where s is the scale factor.
- Integral images are used for fast filtering. Only six operations are required to compute the Haar wavelet response in x or y for any s.
- The Haar wavelet responses are weighted with a Gaussian ($\sigma = 2s$) centered at the interest point. They are represented as points (x_{Haar}, y_{Haar}) where x_{Haar} represents the magnitude of the horizontal response and y_{Haar} represents magnitude of the vertical response.
- The circle is divided into slides of $\frac{pi}{3}$ and the Haar responses are summed for each slice to give a local orientation vector.


dy
dx

Description Vector

- Square regions of size $20s$ are laid over the interest points along the orientation of the square.
- These squares are divided into $4x4$ regions, each of which is broken up into a $5x5$ region. This results in a $20x20$ grid.
- Haar wavelet responses (d_x, d_y) are calculated for each of the cells in this grid. Then, the sums $\sum d_x, \sum d_y, \sum |d_x|, and \sum |d_y|$ are calculated and assigned to each region in the $4x4$ grid. These form a 64-dimension descriptor vector for the interest point.


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

Data

- Standard testing data provided by Mikolajczyk, used to test detector and descriptor.
- Includes images of real textures and structured scenes, and different types of geometric and photometric transformations (varying viewpoint, zoom, rotation, lighting, blur...).
- To test applying this to 3D reconstruction, two views of the same object are evaluated to measure the angle between two planes.
- For n-views 3D reconstruction, 13 input images of an old vase are used.
- For the application to object recognition, data is taken from the Caltech face, background, airplane, and motorcycle databases.

Experimental Setup

- All timings measured on a 3Ghz Pentium 4 machine.
- To test SURF, the authors simply use the testing software provided by Mikolajczyk.
- To test application to 3D reconstruction, SURF was added to the Epoch 3D Webservice of the VISICS research group at the K.U. Leuven
- To test application to image recognition, the experiment is based on the two bag-of-words classifer. The task is to determine if an object occurs in an image or not.

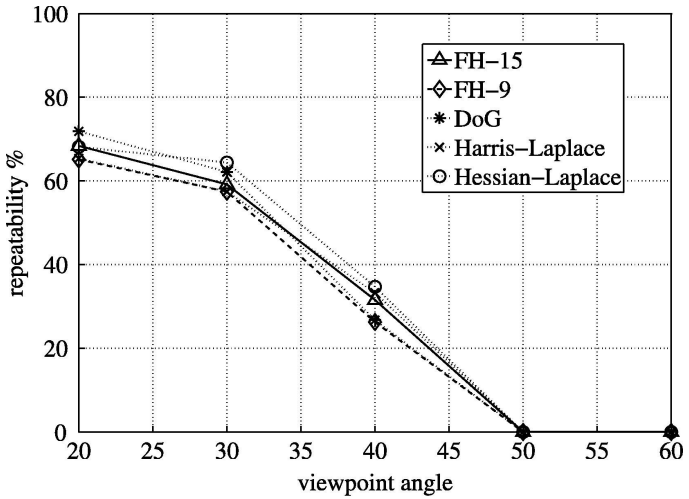
Results

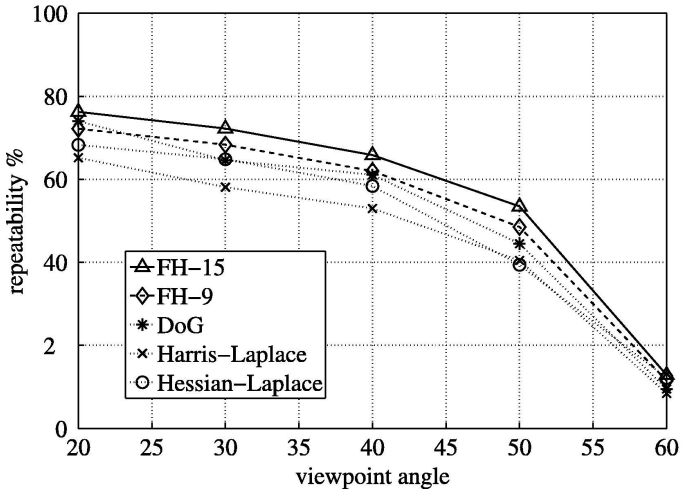
Table 1. Thresholds, number of detected points and calculation time for the detectors in our comparison

Detector	Threshold	Nb of points	Comp. time (ms)
FH-15	60,000	1813	160
FH-9	50,000	1411	70
Hessian-Laplace	1000	1979	700
Harris-Laplace	2500	1664	2100
DoG	Default	1520	400

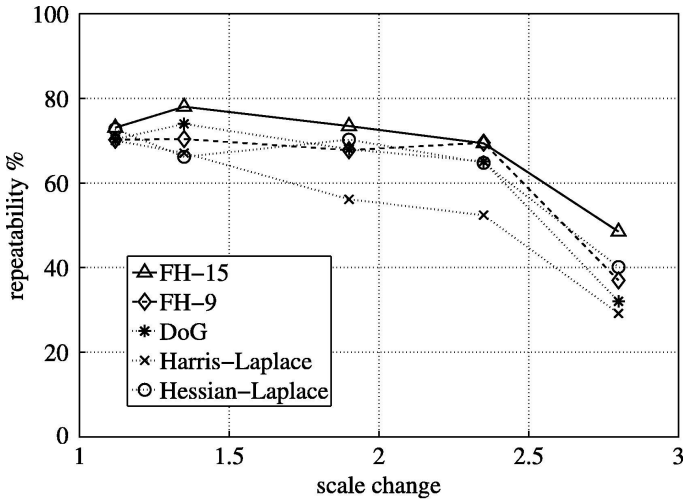
First image of Graffiti scene, 800×640.

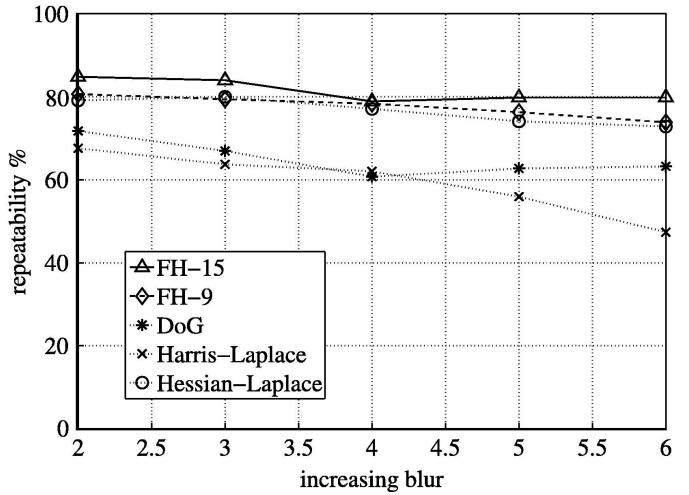
Results


100
80
60
40
20
0
repeatability %
20
25
30
35
40
45
50
55
60
viewpoint angle
FH–15
FH–9
DoG
Harris–Laplace
Hessian–Laplace

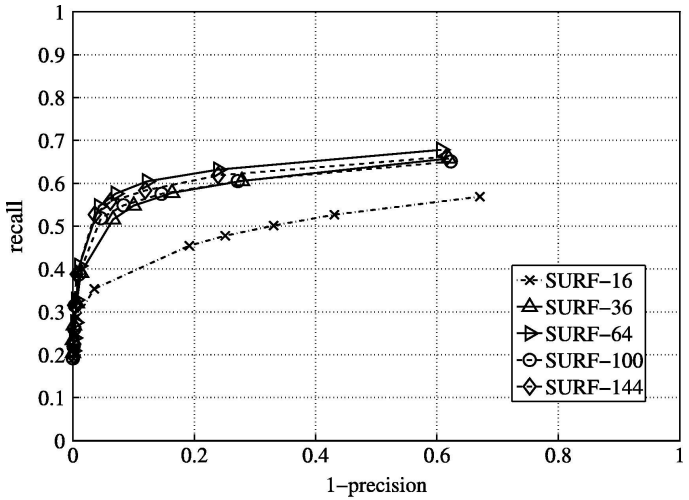

100
80
60
40
2
0
repeatability %
20
25
30
35
40
45
50
55
60
viewpoint angle
FH–15
FH–9
DoG
Harris–Laplace
Hessian–Laplace

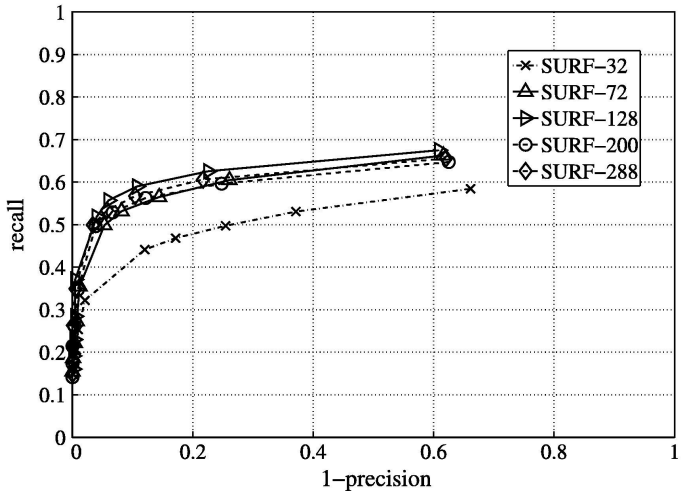
Results


100
80
60
40
20
0
repeatability %
1
1.5
2
2.5
3
scale change
FH–15
FH–9
DoG
Harris–Laplace
Hessian–Laplace

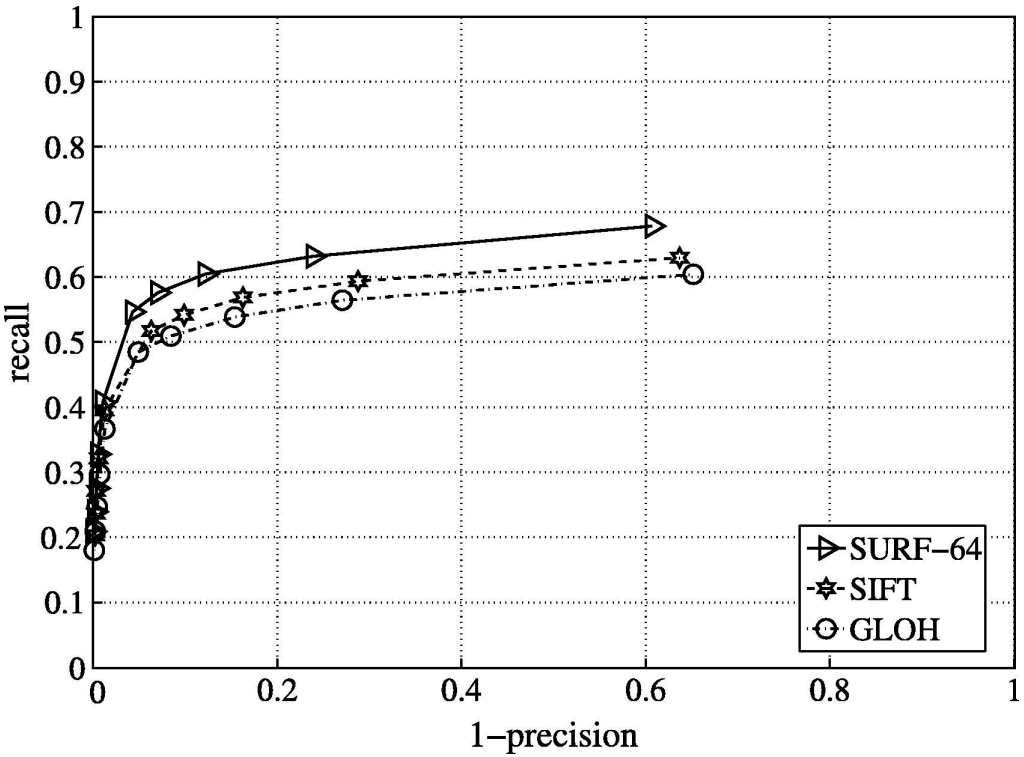

100
80
60
40
20
0
repeatability %
2
2.5
3
3.5
4
4.5
5
5.5
6
increasing blur
FH–15
FH–9
DoG
Harris–Laplace
Hessian–Laplace

Results

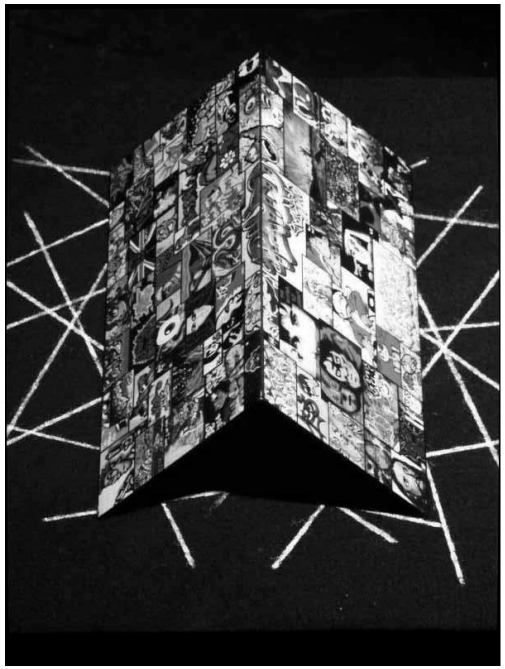

recall
1−precision
SURF−16
SURF−36
SURF−64
SURF−100
SURF−144

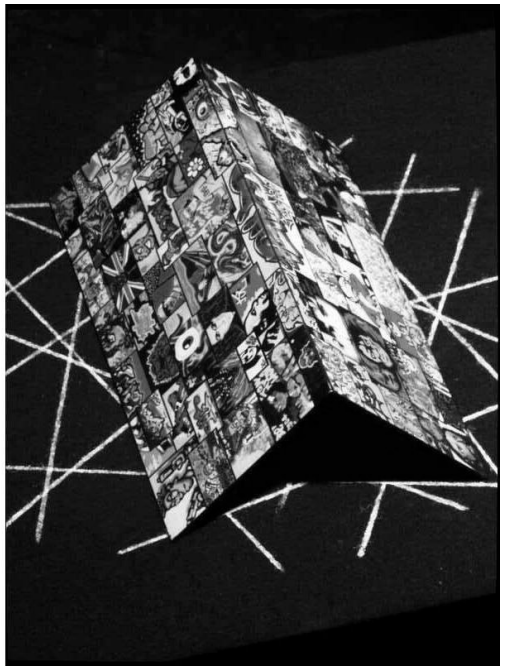

recall
1−precision
SURF−32
SURF−72
SURF−128
SURF−200
SURF−288

Results


recall
1–precision
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1
SURF–64
SIFT
GLOH

Results





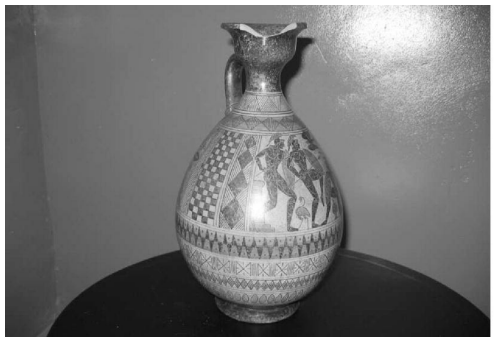
Results

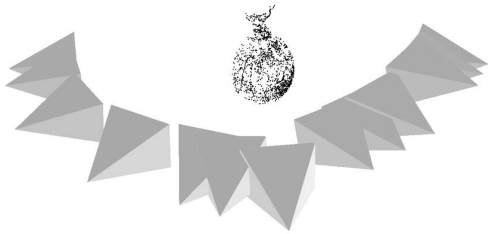
Table 2. Comparison of different interest point detectors for the application of camera calibration and 3D reconstruction

Detector	Angle (°)	Mean dist (pixels)	SD (pixels)
FH-15	88.5	1.14	1.23
FH-9	88.4	1.64	1.78
DoG	88.9	1.95	2.14
Harris-Laplace	88.3	2.13	2.33
Hessian-Laplace	91.1	2.85	3.13

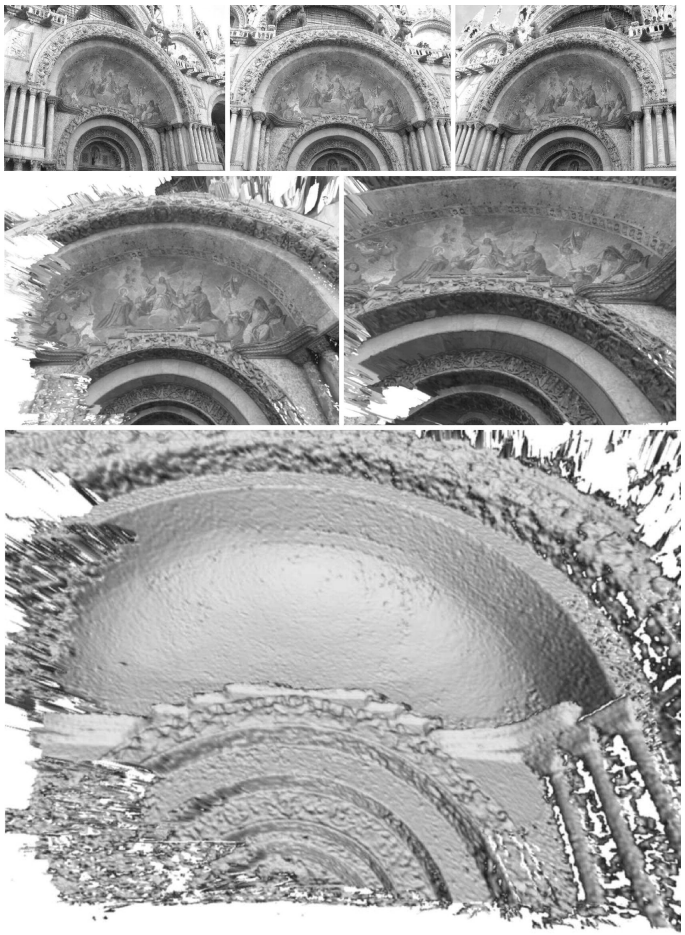
The true angle is 88.6°.

Results

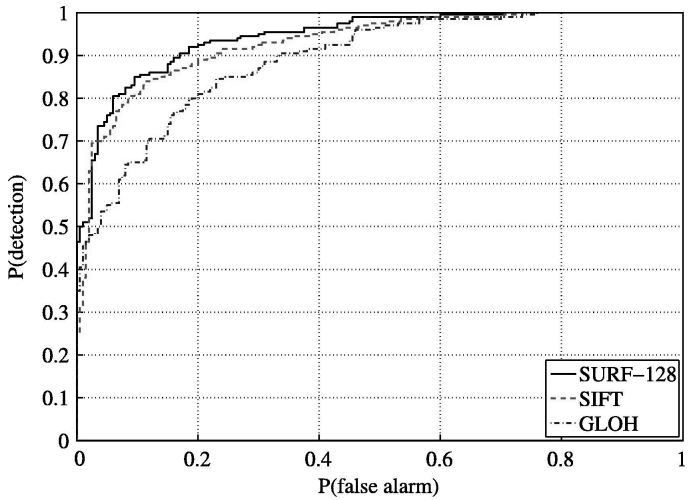


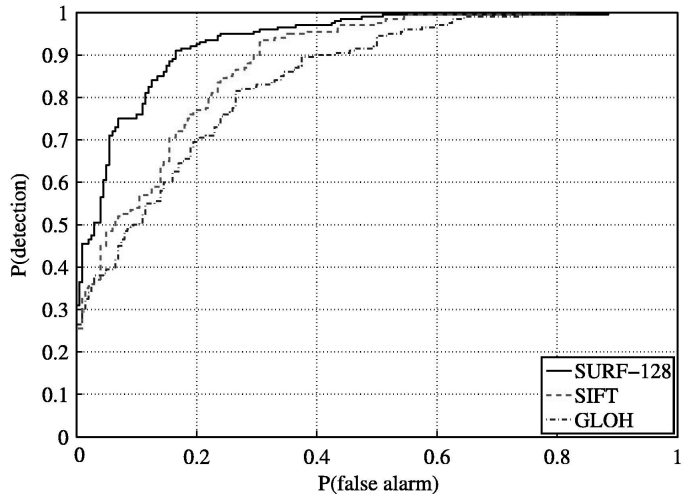


Results



Results


P(detection)
P(false alarm)
SURF–128
SIFT
GLOH


P(detection)
P(false alarm)
SURF–128
SIFT
GLOH

Discussion

SURF vs SIFT

- SURF uses 64-dimensional descriptor vector; SIFT is 128-dimensional.
- SIFT directly computes Gaussian convolution, requiring 25 multiplications and 24 additions to apply a 5x5 filter.
- SURF uses integral images to speed up the application of filters and only requires 10 addition and 2 subtractions to apply the derivative-of-gaussian filter.
- SIFT downscales image to move through scale space; SURF increases filter size.
- SURF requires a contant amount of computations to apply an arbitrary size filter.

Conclusion

- Speeded-Up Robust Features is an approach for detecting, describing, and matching interest points found in an image in a computationally fast but robust way.
- The main metric of evaluation for detection is repeatability, which is the ability of an interest point to be detected under several varying circumstances.
- For description it is uniqueness and completeness and for matching it is mainly speed and accuracy.

Conclusion

- Several approximations are used including box filters and haar wavelets along with several concepts such as integral images, gradient sums, and fast indexing in an attempt to achieve greater efficiency overall, but not at the cost of performance.
- Careful tradeoffs were made and evaluated in an effort to pick the best combinations of parameters and to push approximations even further.
- Through all these techniques, SURF became faster and more accurate than previous methods, including SIFT, Difference of Gaussians, and Hessian-Laplace.

References

- H. Bay et al. SURF: speeded up robust features. Computer Vision and Image Understanding 346-359. Elsevier. 2008.
- http://www.wolframalpha.com/
- http://stuffyoudontwant.com/sports-equipment/soccer-balls/
- http://www.cs.ucf.edu/ mali/haar/
- http://en.wikipedia.org/wiki/Scale_space
- http://www.comp.nus.edu.sg/ cs4243/lecture/imageproc.pdf
- http://rt.com/files/news/curiosity-rover-ready-drive-029/penny-camera-calibration-target.jpg
- http://www.cs.tau.ac.il/ wolf/ORWS/

Index of Mathematical Concepts

- Determinant
- Convolution
- Laplacian
- Gaussian
- Distance measures
- Integral images
- Frobenius norm
- Clairaut's theorem
- Hessian matrix
- Haar wavelets

Determinant

The **determinant** of a matrix A is defined as:

$$det(A) = \sum_{\sigma \subset S_n} sgn(\sigma) \prod_{i=1}^{n} A_i, \sigma_i \tag{3}$$

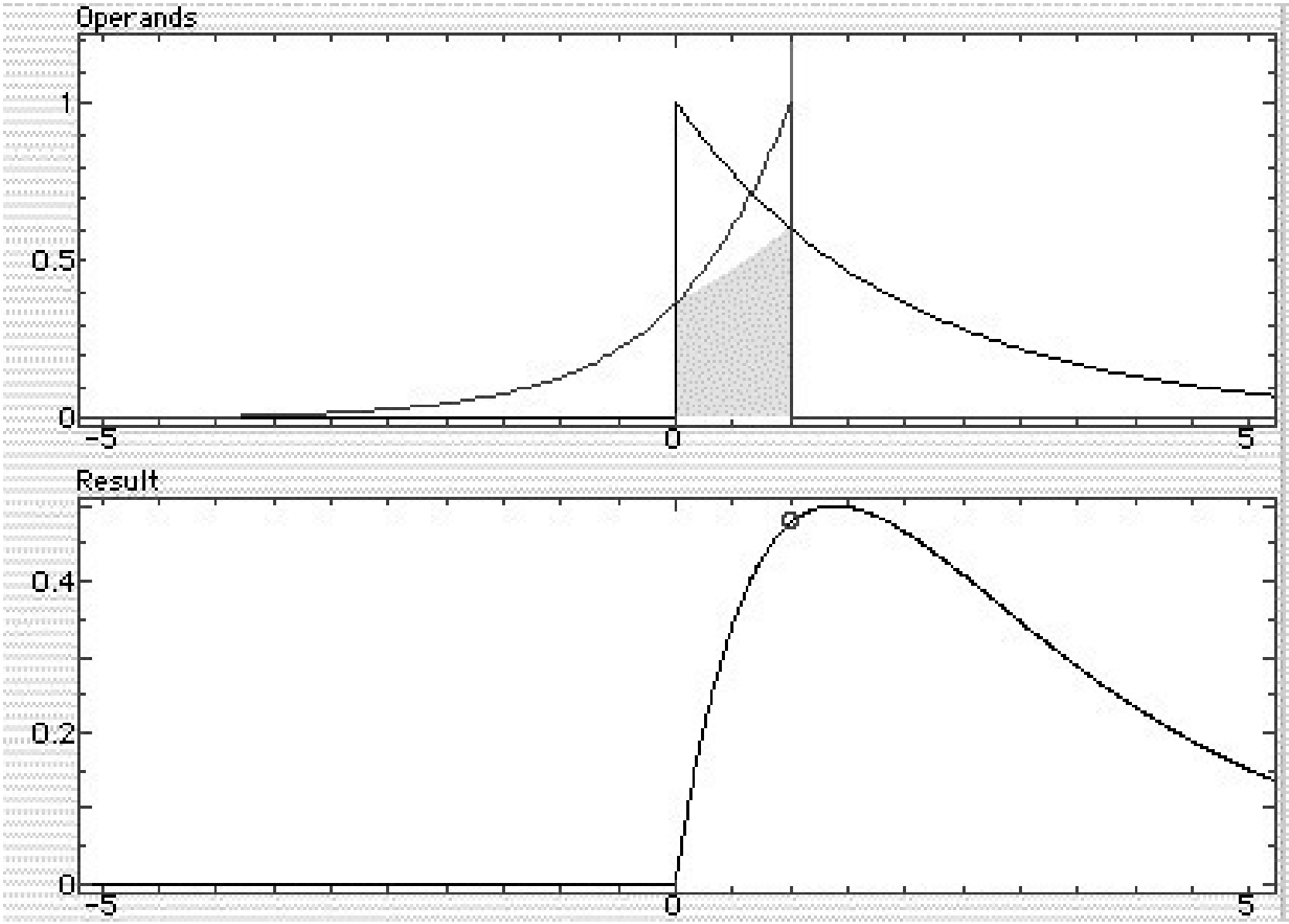
If a parallelogram is represented by a matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ with points (0,0), (a,b), and (c,d), (a+b,c+d), then the determinant $ad - bc$ gives the area of the parallelogram. Likewise the determinant of a matrix representing a parallelepiped yields the volume.

Convolution

The convolution is an integral transform on a function f using a function g and is defined as:

$$(f * g)(t) = \int_{\infty}^{-\infty} f(\tau)g(t-\tau)d\tau. \tag{4}$$

The convolution gives the area of overlap between f and g for all values of the offset t.


Operands
1
0.5
0
-5
0
5
Result
0.4
0.2
0
-5
0
5

Laplacian

The Laplacian operator, or ∇^2, is defined as the n-dimensional vector:

$$\langle \frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \dots \frac{\partial}{\partial x_n} \rangle. \tag{5}$$

The Laplacian of f, or $\nabla^2 f$, is thus defined as:

$$\sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2}; \tag{6}$$

that is, the sum of the second-order partial derivatives of f.

- Suppose $f(x, y) = x^2 + y^2$. Then the Laplacian is:
- $\frac{\partial^2}{\partial_x^2}(x^2 + y^2) + \frac{\partial^2}{\partial_x^2}(x^2 + y^2) = 2 + 2 = 4$.

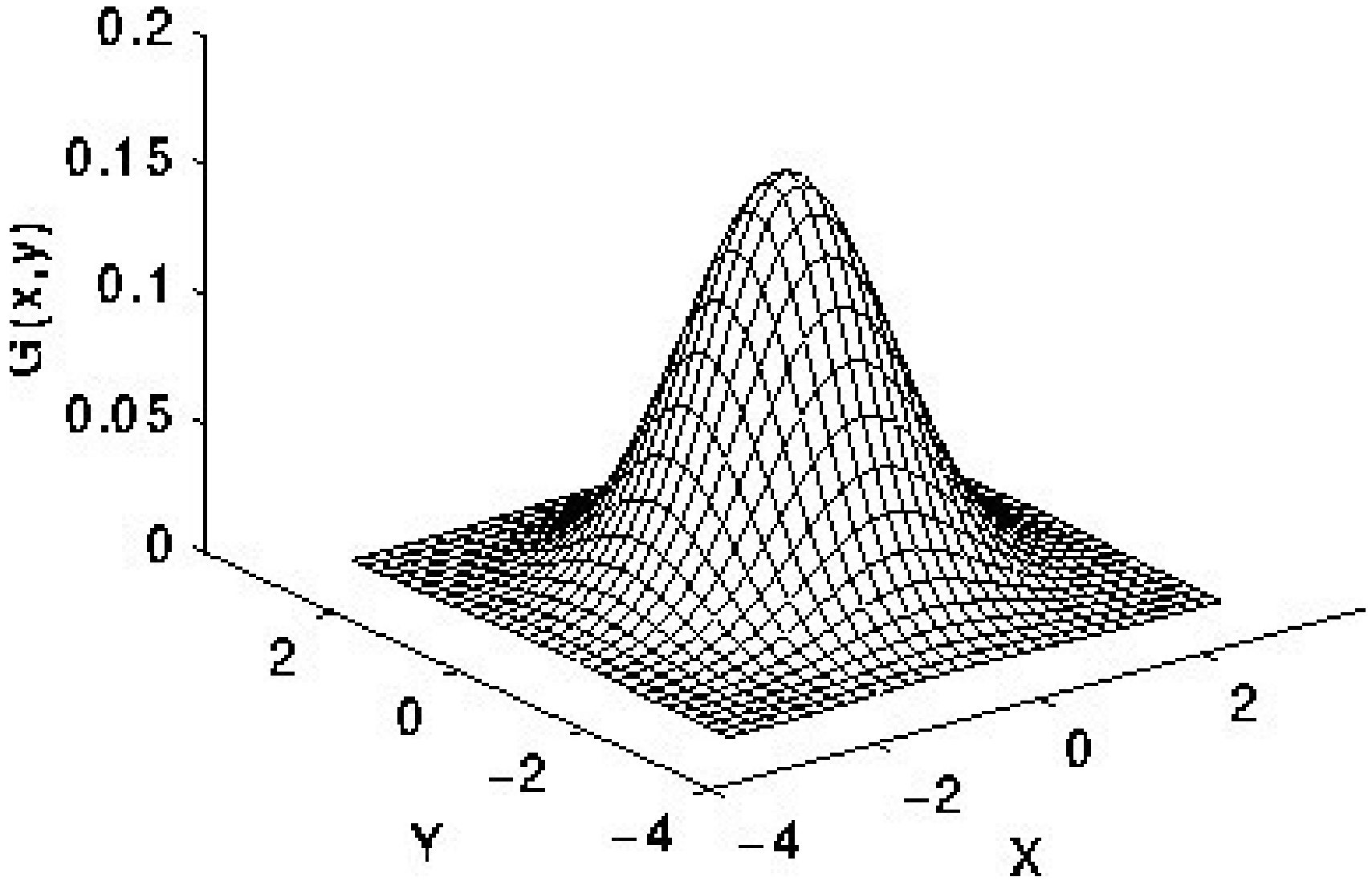
- Suppose $f(x, y) = x^2 y^2$. Then the Laplacian is:
- $\frac{\partial^2}{\partial_x^2}(x^2 y^2) + \frac{\partial^2}{\partial_x^2}(x^2 y^2) = 2x^2 + 2y^2$.

Weierstrass Transform, or Gaussian Blur

The 2-dimensional Gaussian function is defined as follows:

$$G(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{7}$$

the graph of which takes the shape of a bell. When a Gaussian is used to convolute an image I, the new pixel at $I(x, y)$ becomes the weighted average of all pixels in its neighborhood, producing a smoothing or blurring effect.


0.2
0.15
0.1
0.05
0
G(x,y)
2
0
−2
Y
−4
−4
−2
0
2
X

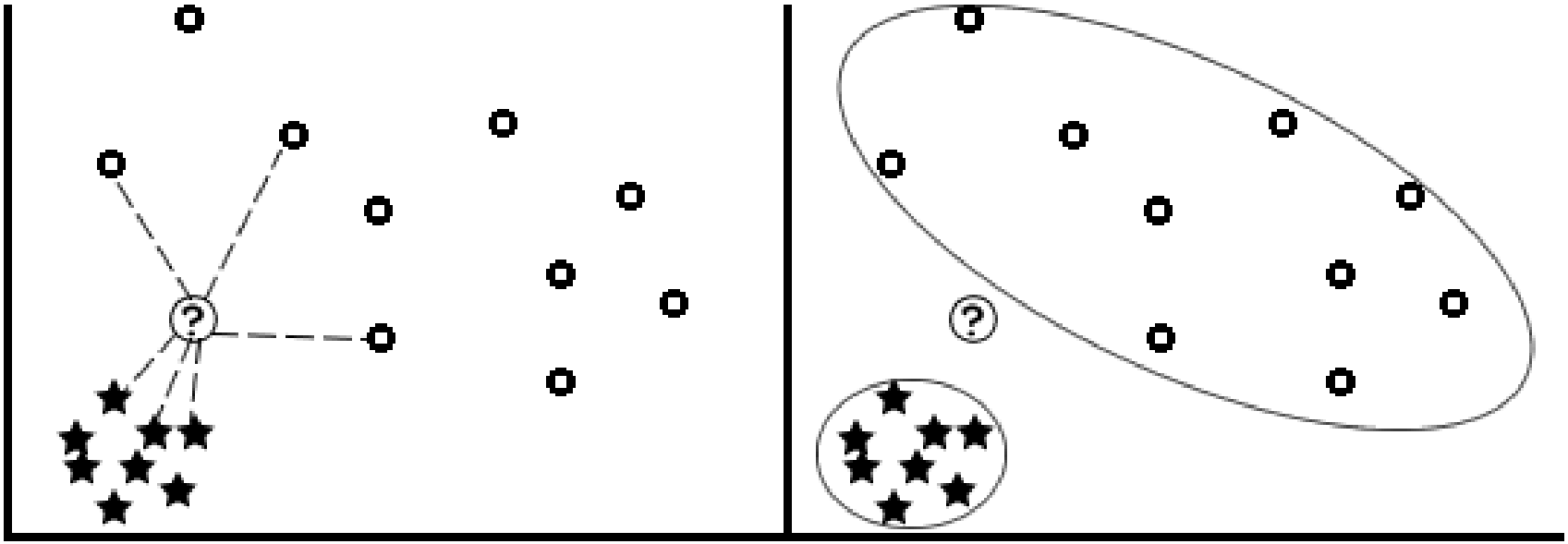
Euclidean and Mahalanobis Distances

Euclidean distance: for two given points p_i and q_i, the Euclidean distance is:

$$d(x, y) = \sum_{i=0}^{N} \sqrt{(p_i - q_i)^2}. \tag{8}$$

Mahalanobis distance: for a given multivariate vector $x = (x_1, x_2 \ldots x_n)$ the Mahalanobis distance from a group of values with mean $\mu = (\mu_1, \mu_2 \ldots \mu_n)$ is defined as:

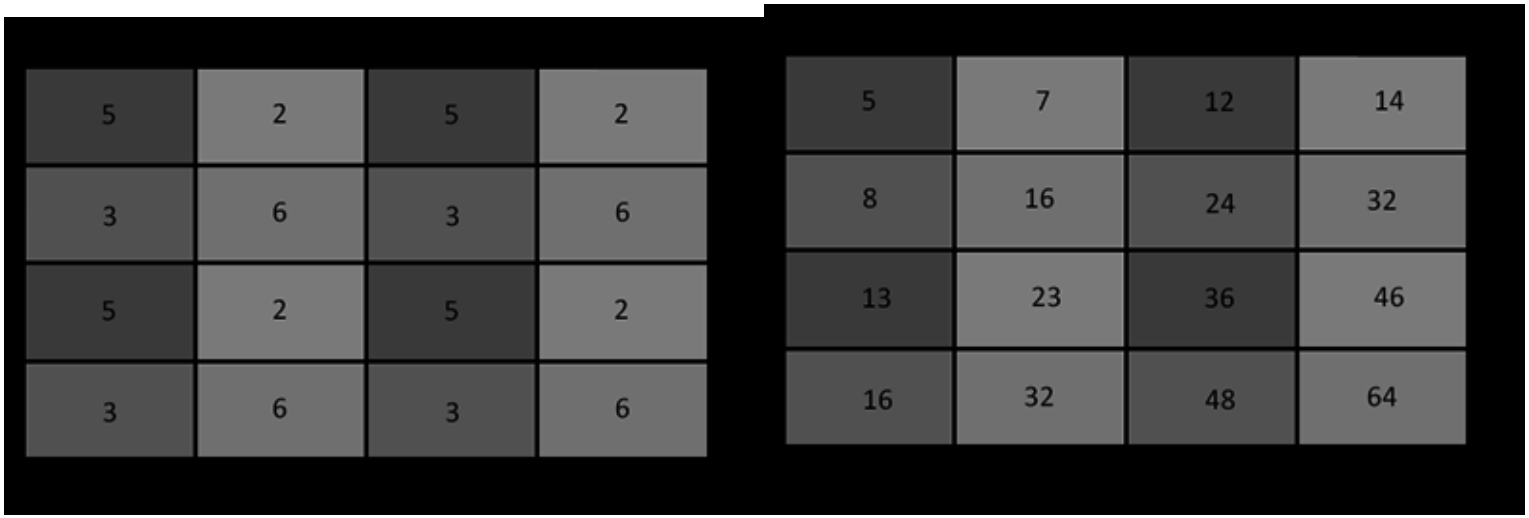
$$D_M(x) = \sqrt{(x - \mu)^T S^{-1} (x - \mu)}. \tag{9}$$

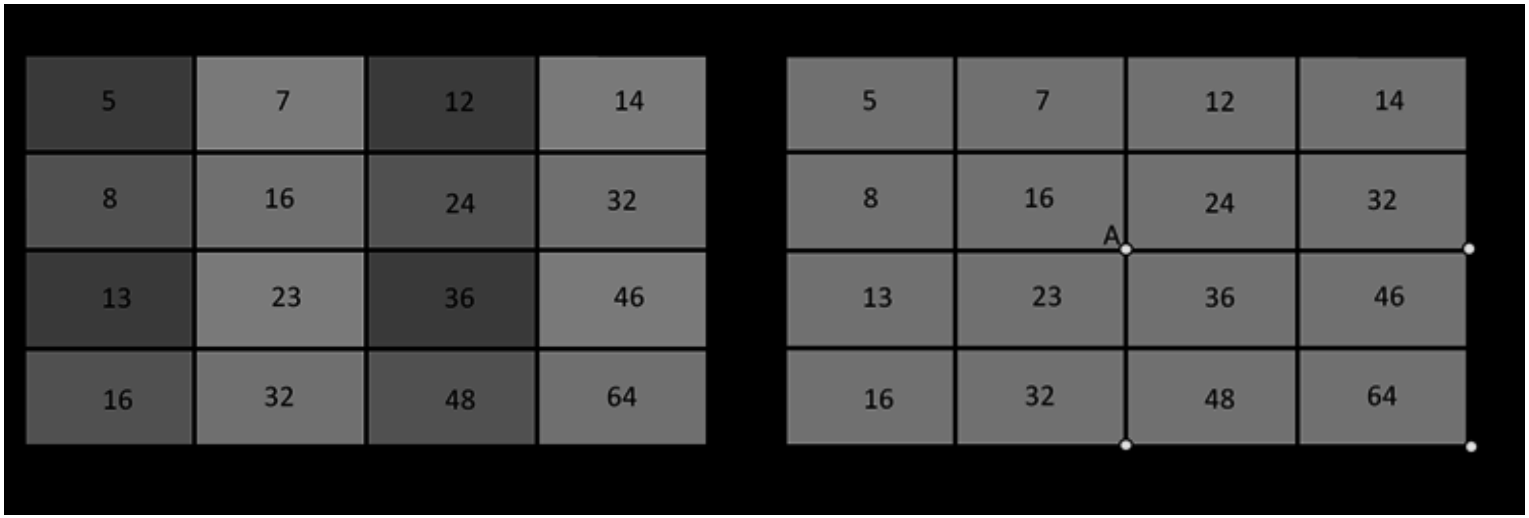


Integral Images

The integral image $I_\Sigma(x)$ at a location $\mathbf{x} = (x, y)^T$ is the sum of pixels in the input image I within a rectangular region formed by the origin and $\mathbf{x}$:

$$\sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j). \tag{10}$$


5 2 5 2
3 6 3 6
5 2 5 2
3 6 3 6
5 7 12 14
8 16 24 32
13 23 36 46
16 32 48 64


5
7
12
14
8
16
24
32
13
23
36
46
16
32
48
64
5
7
12
14
8
16
A
24
32
13
23
36
46
16
32
48
64

Frobenius Norm

- The Frobenius norm $|A|_F$ of a matrix A is simply defined as:

$$\sqrt{\sum_{i=0}^{n}\sum_{j=0}^{m} A_{ij}^2} \tag{11}$$

- So, if the matrix is $\begin{bmatrix} 2 & 5 \\ 3 & 4 \end{bmatrix}$
- then the Frobenius norm is $\sqrt{2^2 + 5^2 + 3^2 + 4^2} = \sqrt{54}$.

Clairaut's Theorem

- As a consequence of Clairaut's theorem:

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial^2 f}{\partial y \partial x}. \tag{12}$$

- E.g. if $f(x, y) = x^2 y^2$, then $\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial^2 f}{\partial y \partial x} = 4xy$.
- Check this for yourself on more complex functions.

Hessian Matrix

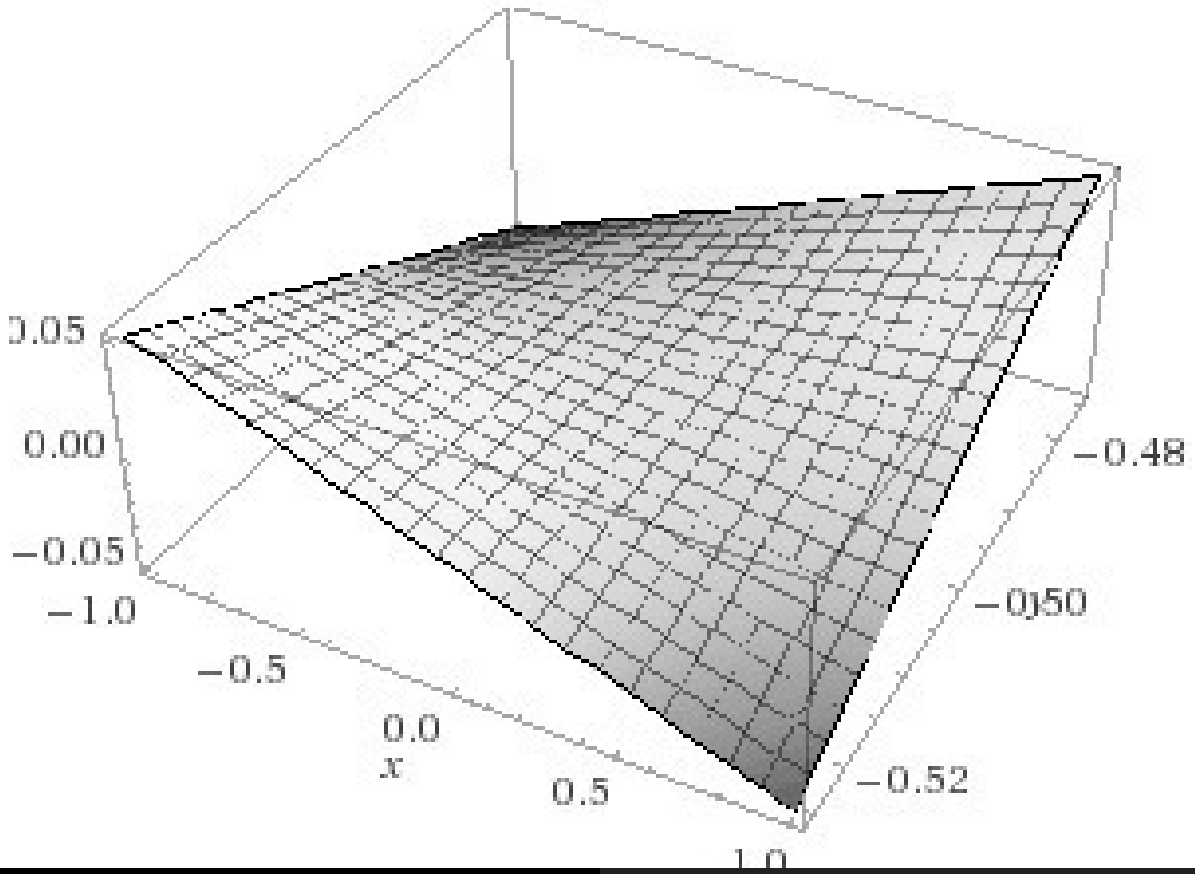
Given a point $\mathbf{x} = (x, y)$ in an image I, the Hessian matrix

$$\mathcal{H}(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \tag{13}$$

where $L_{xx}(x, y)$ is the convolution of the Gaussian second-order derivative $\frac{\partial^2}{\partial x^x} g(\sigma)$ with the image I in point $\mathbf{x}$; similarly for $L_{xy}(x, \sigma)$ and $L_{yy}(x, \sigma)$.

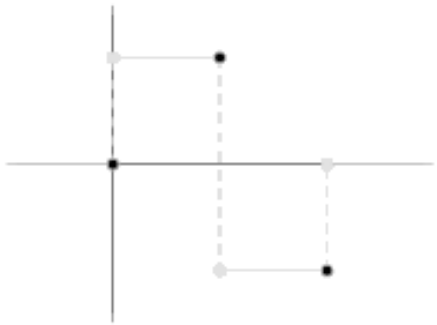
$$f = x(1 + 2y)$$

$$H(x) = \begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix}$$



Haar Wavelets

- A Haar wavelet is a graph of a sequence of discontinuous square-shaped functions:



- This particular Haar wavelet is described by the function:
  - $\Phi(t) = 1 for 0 \leq \frac{1}{2}$,
  - $\Phi(t) = -1 for \frac{1}{2} \leq 1$, and
  - 0 otherwise.