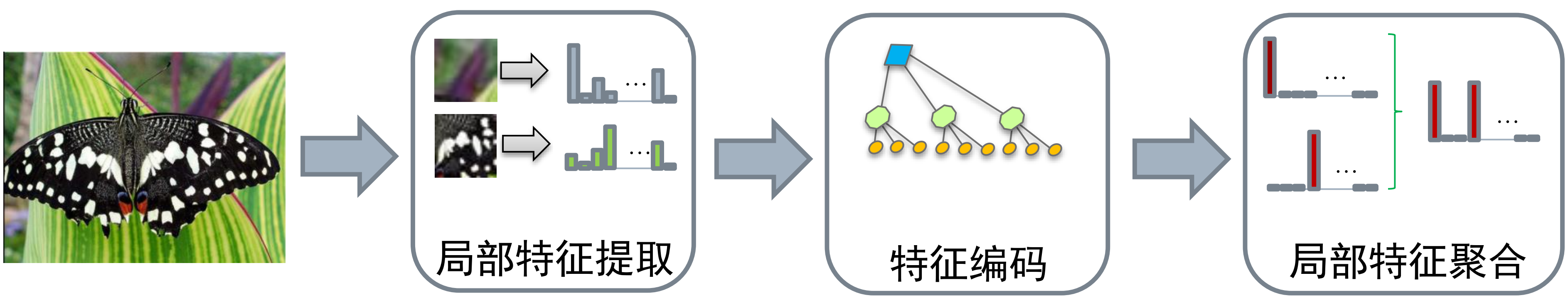
第九章 基于局部视觉特征的图像表达

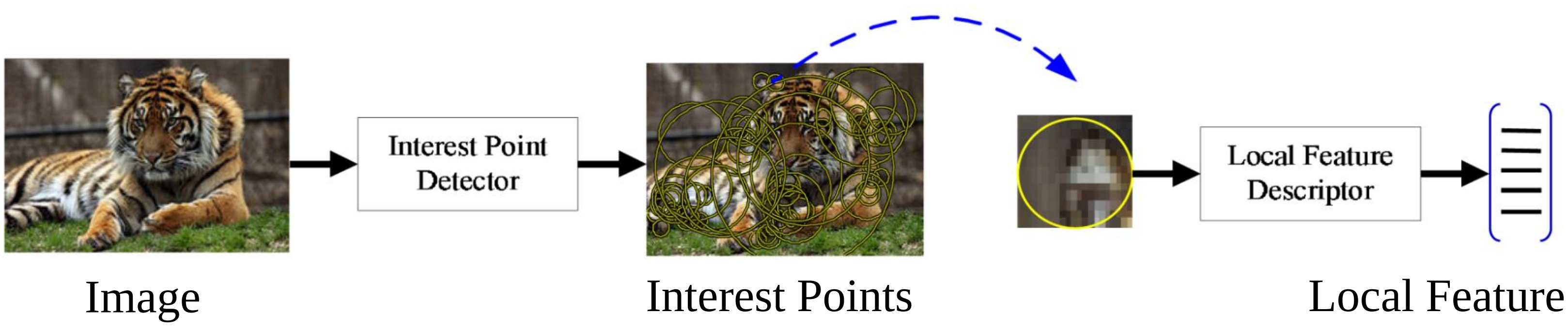
9.1 图像表达基本框架

- 在图像分析中，许多问题可归结为图像间的比较
- 局部视觉特征对视觉内容有良好的区分性和表达力
- 不同的图像中包含的视觉特征数量不同，有必要进行聚合，生成固定长度的矢量



9.2 局部视觉特征描述

- 局部关键点检测
  - 对**平移、旋转、缩放变化**具有不变性
- 局部区域描述
  - 将图像局部区域变换为固定长度的特征向量
    - ✓ 浮点型特征
    - ✓ 二值型特征
  - 对**旋转和亮度变化**具有不变性



局部特征性质

- 可重复性 （Repeatability）
  - 图像经过变换后，同样的特征仍然存在
- 显著性 （Saliency）
  - 每个特征都对应一个有区别性的描述
- 紧凑性和有效性 （Compactness and efficiency）
  - 特征数目远少于图像像素数
- 局部性 （Locality）
  - 每个特征对应着一个相对较小的图像区域;
  - 对于混乱背景和遮挡具有鲁棒性

目标:关键点的可重复性

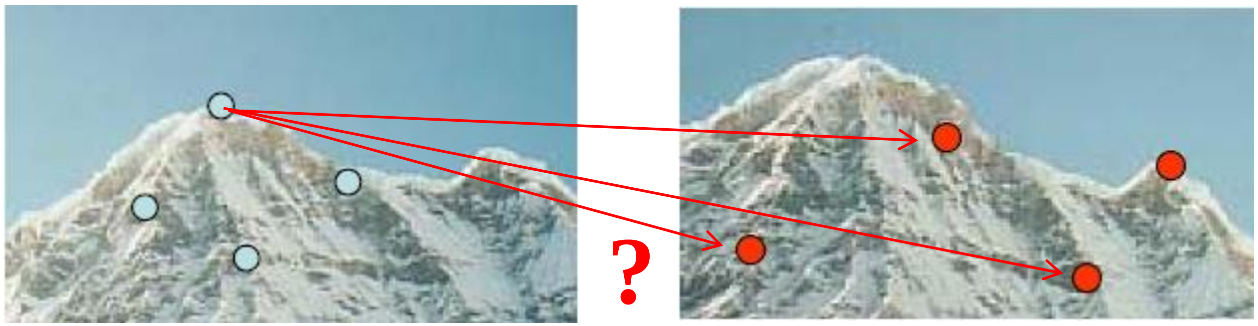
- 在两个图像中检测若干相同的关键点

No chance to find true matches!

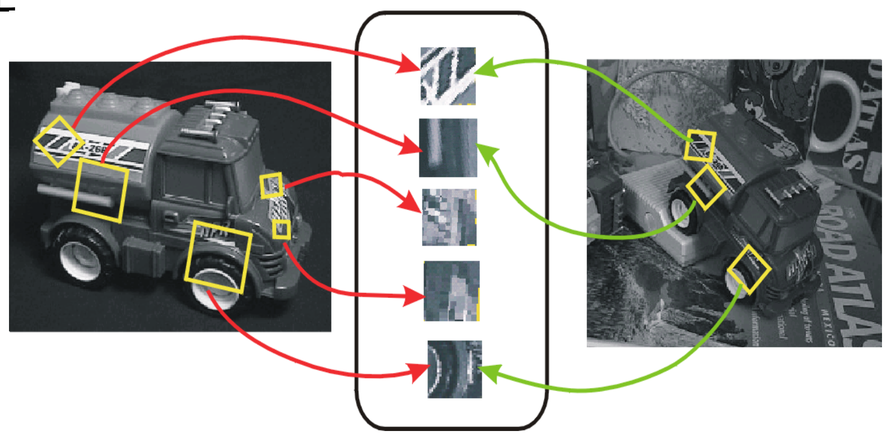
- 对每个图像，关键点检测独立进行

目标: 描述子的可区分性

- 可靠地确定兴趣点的对应关系



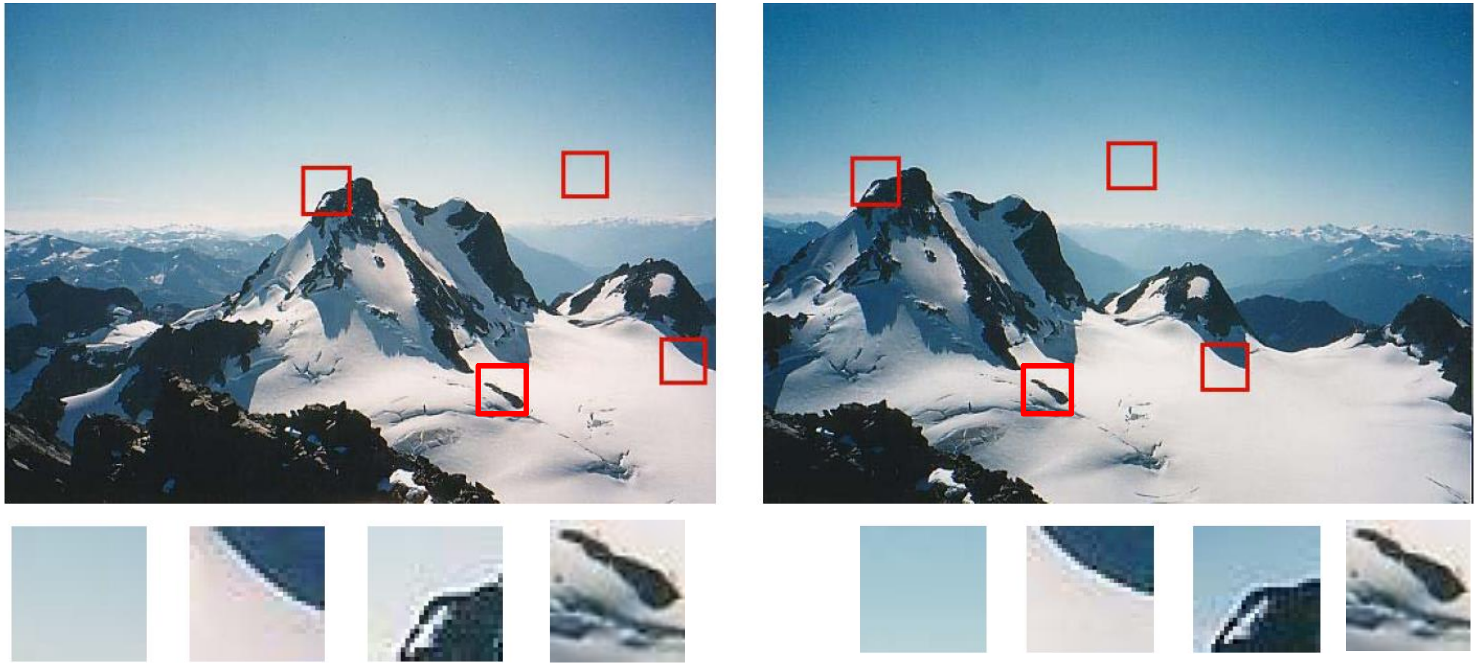
- 对于两个视角，描述子必须具有对几何和光照变化的不变性



选择那些点作为兴趣点？

局部视觉特征

- 在图像中寻找那些与众不同的区域？
  - Lead to unambiguous matches in other images
  - 如何定义这种“与众不同”？

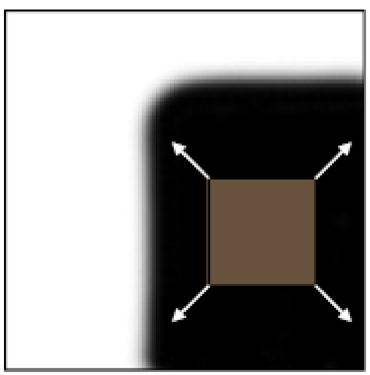


9.2.1 局部关键点检测

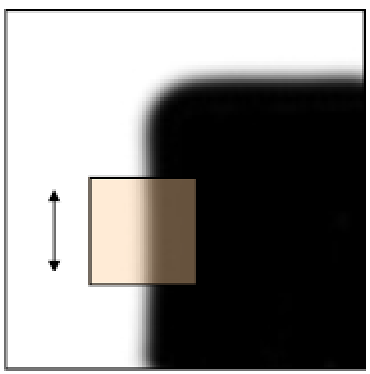
- 关键点检测子
  - 角点(corner)检测
    - ✓ Harris角点检测子
    - ✓ FAST角点检测子
  - 块(blob)检测
    - ✓ 高斯差分(DoG)检测子
    - ✓ SURF detector
    - ✓ MSER detector
    - ✓ Hessian-Affine detector

关键点检测：角点

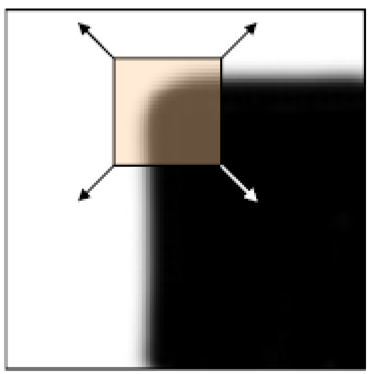
- 通过查看一个小窗口，即可简单的识别角点
- 在角点上，向任何一个方向移动窗口，都会产生灰度的较大变化



"平坦" 区域：任意方向灰度均无变化



"边缘"：沿边缘方向，灰度无变化



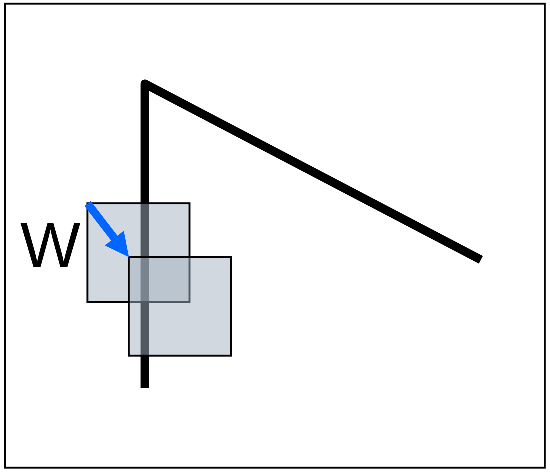
"角点"：所有方向上，灰度均有较大变化

Harris Corner Detector

Consider shifting the window W by (u,v)

- how do the pixels in W change?
- compare each pixel before and after by summing up the squared differences

$$E(u,v) = \sum_{(x,y)\in W} \left[I(x+u,y+v) - I(x,y)\right]^2$$



Taylor Series expansion of *I*:

$$I(x+u,y+v) = I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$

If the motion (u,v) is small, then first order approx is good

$$I(x+u,y+v) \approx I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v$$

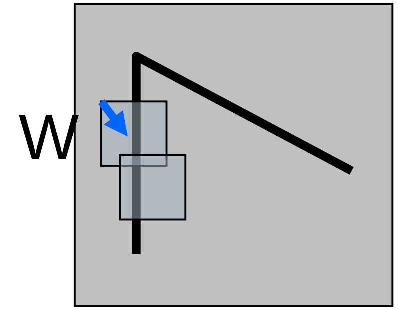
$$\approx I(x,y) + [I_x \;\; I_y]\begin{bmatrix} u \\ v \end{bmatrix}$$

shorthand: $I_x = \frac{\partial I}{\partial x}$

Harris Corner Detector

$$E(u,v) = \sum_{(x,y)\in W} [I(x+u, y+v) - I(x,y)]^2$$

$$\approx I(x,y) + [I_x \;\; I_y] \begin{bmatrix} u \\ v \end{bmatrix}$$



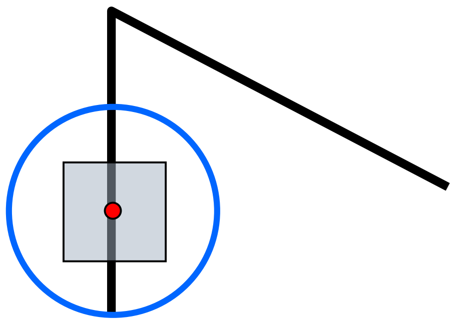
$$I(x+u, y+v)$$

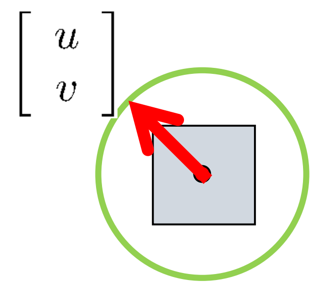
$$\begin{aligned} E(u,v) &= \sum_{(x,y)\in W} [I(x+u, y+v) - I(x,y)]^2 \\ &\approx \sum_{(x,y)\in W} \left[I(x,y) + [I_x \;\; I_y] \begin{bmatrix} u \\ v \end{bmatrix} - I(x,y)\right]^2 \\ &\approx \sum_{(x,y)\in W} \left[[I_x \;\; I_y] \begin{bmatrix} u \\ v \end{bmatrix}\right]^2 \end{aligned}$$

Harris Corner Detector

This can be rewritten as

$$E(u,v) = \sum_{(x,y)\in W} [u \ v] \underbrace{\begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}}_{M} \begin{bmatrix} u \\ v \end{bmatrix}$$





For the example above

- You can move the center of the blue window to anywhere on the blue unit circle
- Which directions will result in the largest and smallest E values?
- We can find these directions by looking at the eigenvectors of *M*

回顾：特征向量、特征值

- The eigenvectors of a matrix A are the vectors x that satisfy:

$$Ax = \lambda x$$

- The scalar λ is the eigenvalue corresponding to x
  - The eigenvalues are found by solving:

$$det(A - \lambda I) = 0$$

  - In our case, A = M is a 2x2 matrix, so we have

$$\det \begin{bmatrix} m_{11} - \lambda & m_{12} \\ m_{21} & m_{22} - \lambda \end{bmatrix} = 0$$

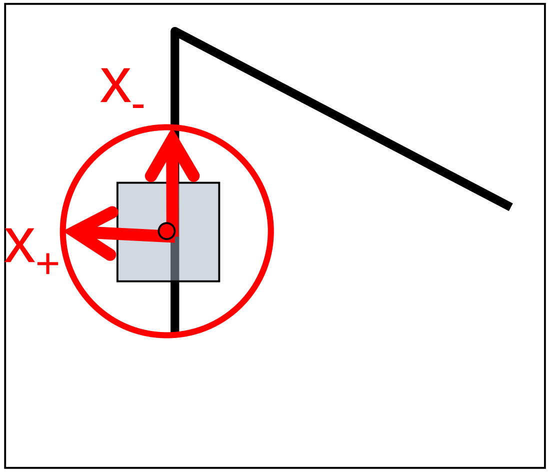
  - The solution:

$$\lambda_{\pm} = \frac{1}{2}\left[(m_{11} + m_{22}) \pm \sqrt{4m_{12}m_{21} + (m_{11} - m_{22})^2} \right]$$

- Once you know λ, you find x by solving $\begin{bmatrix} m_{11} - \lambda & m_{12} \\ m_{21} & m_{22} - \lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$

Harris Corner Detector

$$E(u,v) = \begin{bmatrix} u \\ v \end{bmatrix} \underbrace{\sum_{(x,y)\in W} \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}}_{M} \begin{bmatrix} u \\ v \end{bmatrix}$$



Eigenvalues and eigenvectors of *M*

- Define shifts with the smallest and largest change (E value)
- x_+ = direction of largest increase in E.
- λ_+ = amount of increase in direction x_+
- x_- = direction of smallest increase in E.
- λ- = amount of increase in direction x_+

$$Mx_+ = \lambda_+ x_+$$

$$Mx_- = \lambda_- x_-$$

Harris Corner Detector

Intensity change in shifting window: eigenvalue analysis

$$E(u,v) \cong [u,v] \;\; M \begin{bmatrix} u \\ v \end{bmatrix}$$

$\lambda_{min}, \lambda_{max}$ – eigenvalues of M

If we try every possible orientation **n**, the max. change in intensity is λ_{max}

$$M\mathbf{x}_{\max} = \lambda_{\max}\mathbf{x}_{\max}; \quad M\mathbf{x}_{\min} = \lambda_{\min}\mathbf{x}_{\min}; \quad \mathbf{x}_{\max}{}^{T}\mathbf{x}_{\min} = 0$$

Let $\mathbf{n} = [u, v]^T = a\mathbf{x}_{max} + b\mathbf{x}_{min}$, with $a^2 + b^2 = 1$

Then, $E(\mathbf{n}) = \mathbf{n}^T \mathrm{M} \mathbf{n}$

$$= (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})^{T} \mathrm{M} (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})$$

$$= (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})^{T} (\mathrm{a}\lambda_{\max}\mathbf{x}_{\max} + \mathrm{b}\lambda_{\min}\mathbf{x}_{\min})$$

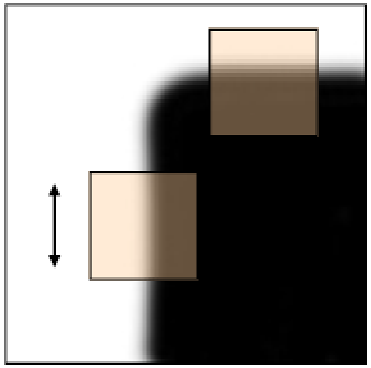
$$= a^2\lambda_{\max} + b^2\lambda_{\min}$$

$$= a^2\lambda_{\max} + (1 - a^2)\lambda_{\min}$$

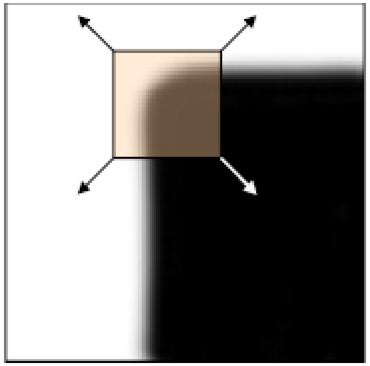
Therefore, $E_{\max} = \lambda_{\max}$, with $a = 1$ and $\mathbf{n} = [u, v]^T = \mathbf{x}_{\max}$;

$E_{\min} = \lambda_{\min}$, with $a = 0$ and $\mathbf{n} = [u, v]^T = \mathbf{x}_{min}$.

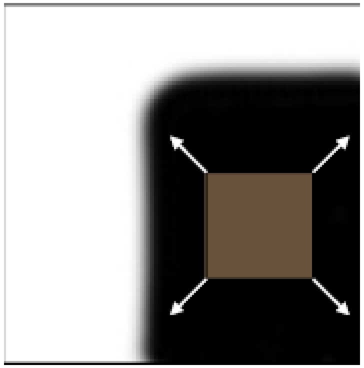
Corner 响应函数



“edge”:
λ1 >> λ2
λ2 >> λ1



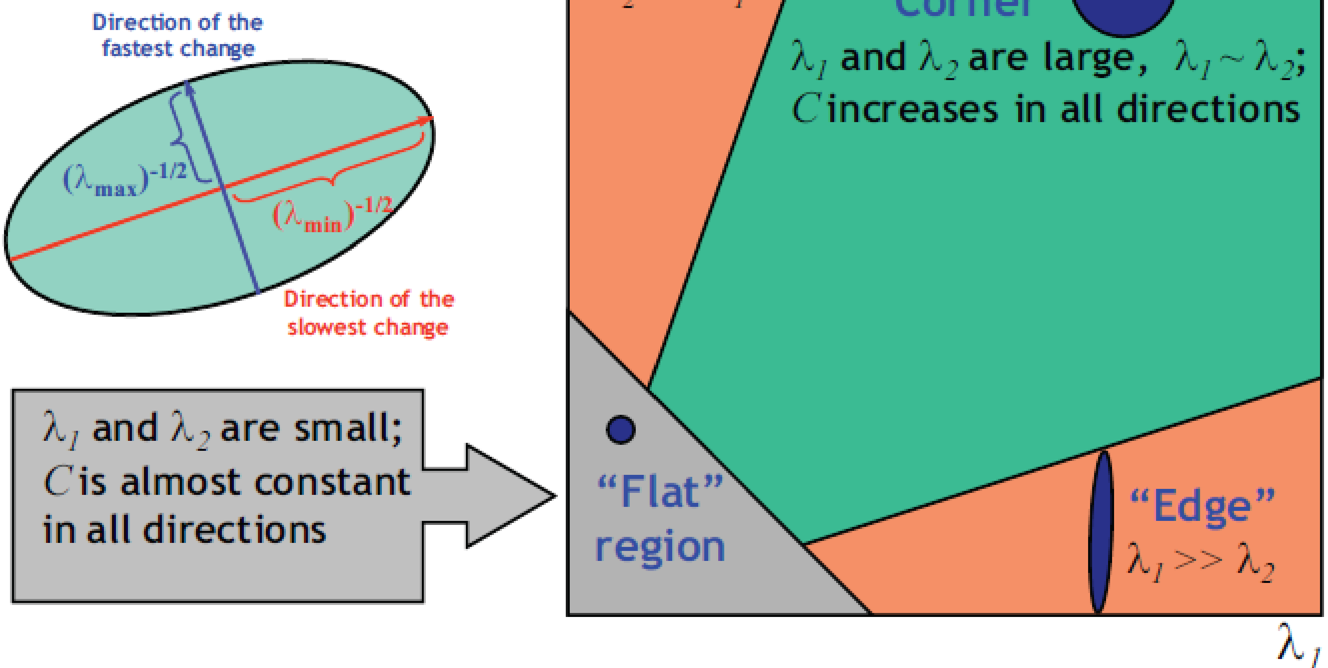
“corner”:
λ1 和 λ2 均较大,
λ1 ~ λ2;



“flat” region：
λ1 和 λ2 均较小;

Corner 响应函数

$$C = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$



Harris 角点检测算子

- 1) 对每个图像窗，计算 M 矩阵，得到相应的角点响应 R 值.
- 2) 找到角点响应较大的值所对应的图像位置点：

$$R > threshold$$

- 3) 取 R 的局部极大值点，例如 进行非最大抑制

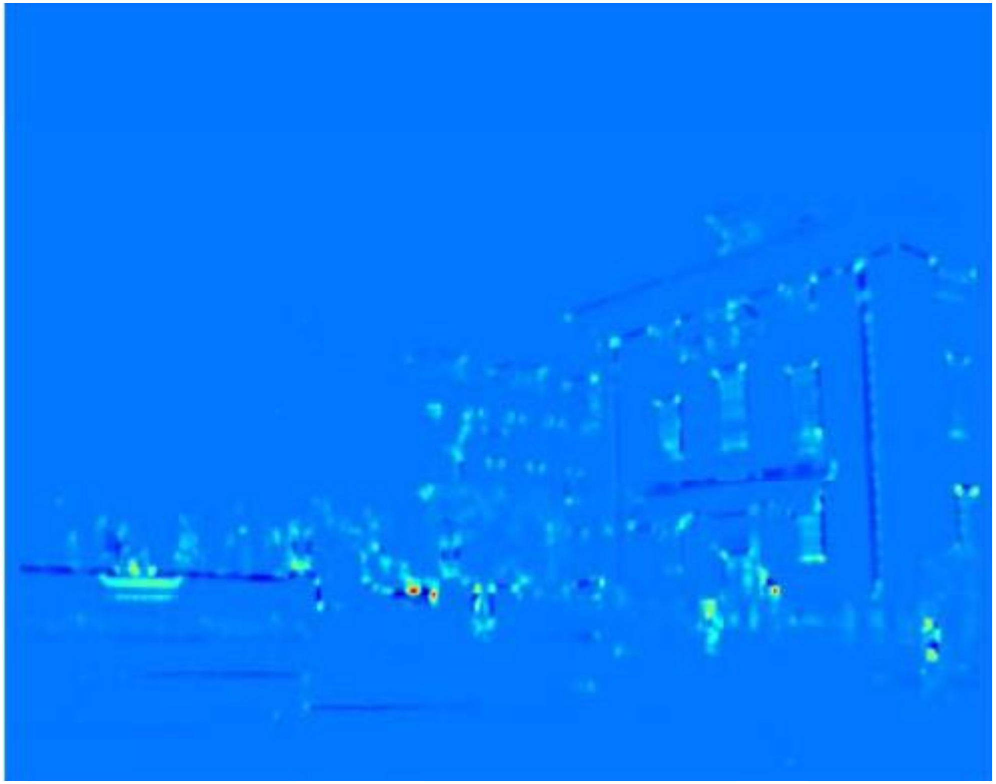
$$R(x,y) = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

$$\begin{aligned} \det(\mathbf{C}) &= \lambda_1\lambda_2 \\ \operatorname{trace}(\mathbf{C}) &= \lambda_1 + \lambda_2 \end{aligned}$$

Harris 应用示例

角点响应图 (R)

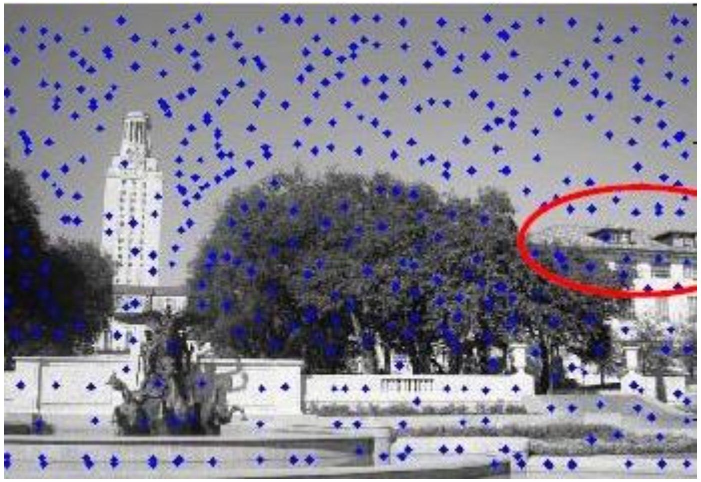
在每个像素处，计算角点响应值 R.



检测的 corner-like 点

对 R 进行门限处理, 取极大值.

对两张图像独立进行检测处理



Harris 角点检测算子的性质

- 旋转不变?
  - Yes

$$M = X\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T$$

- 尺度不变?
  - No

校正窗口尺度函数

Window scale = 10

Window scale = 15

Window scale = 30

Harris 检测算子检测到的特征点 取决于所选的窗的大小.

9.2.1 局部关键点检测

- 关键点检测子
  - 角点(corner)检测
    - ✓ Harris角点检测子
    - ✓ FAST角点检测子
  - 块(blob)检测
    - ✓ 高斯差分(DoG)检测子
    - ✓ SURF detector
    - ✓ MSER detector
    - ✓ Hessian-Affine detector

角点检测II:
FAST: Features from Accelerated Segment Test

基本思想：

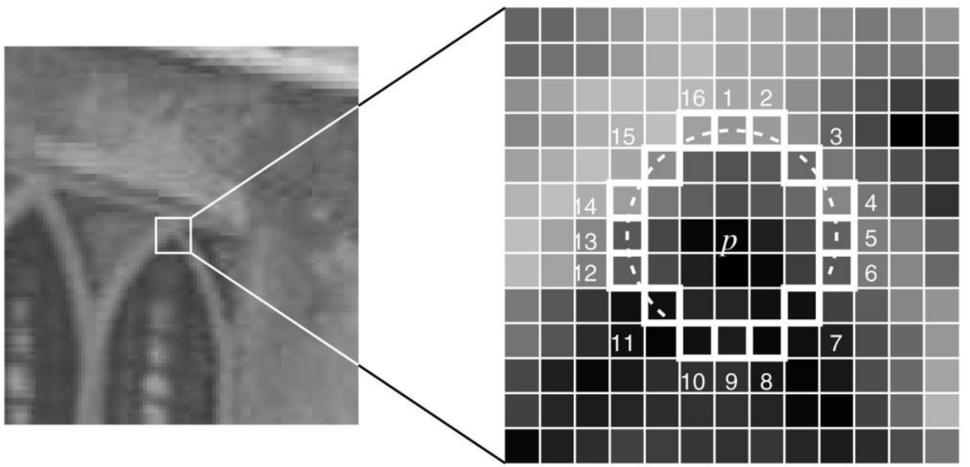
- 考察以候选点p（灰度值为I_p）为圆心的16个像素的灰度值
- 若连续n个像素的灰度值大于（I_p+t)，或者小于（I_p-t)，则像素点p为角点

快速算法

◆首先考察像素点1和9，若其灰度值与I_p之差均小于t，则候选的p不为角点；

◆若上述测试通过，增加考察像素点5和13；如果p为角点，则上述四个点的灰度值至少有三个大于（I_p+t)或者小于（I_p-t)；

◆若上述测试通过，则考察是否有连续n个像素的灰度值大于（I_p+t)，或者小于（I_p-t)，如符合，则为角点。

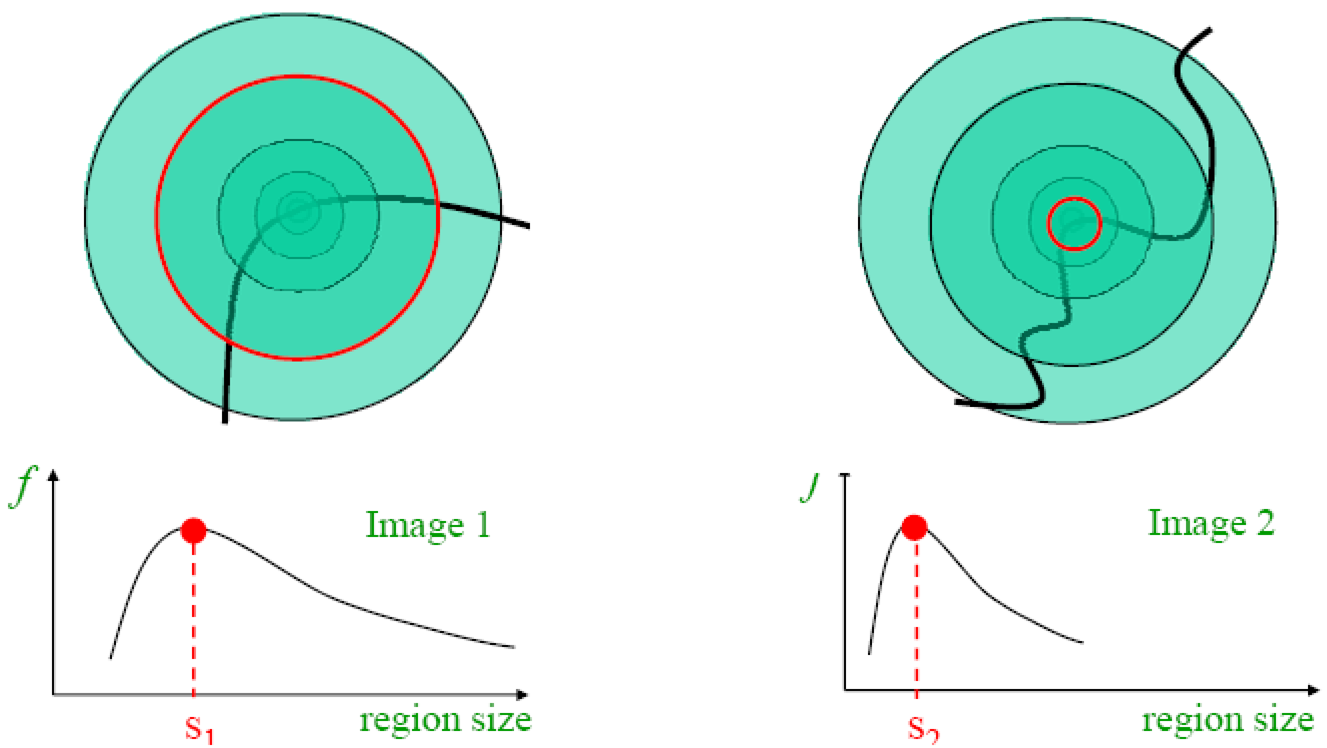


9.2.1 局部关键点检测

- 关键点检测子
  - 角点(corner)检测
    - Harris角点检测子
    - FAST角点检测子
  - 块(blob)检测
    - 高斯差分(DoG)检测子
    - SURF detector
    - MSER detector
    - Hessian-Affine detector

自动的尺度选择

- **Intuition: 对于某个函数 f, 寻找尺度，使得其在尺度和位置 产生局部极大值。**



自动尺度选择

- How to find corresponding patch size independently?

Signature Function

$$f(I_{i_1\ldots i_m}(x,\sigma)) = f(I_{i_1\ldots i_m}(x',\sigma'))$$

- Intuition: Find scale yielding local maxima of a signature function *f* in both position and scale.

自动尺度选择

- Function responses for increasing scale (scale signature)

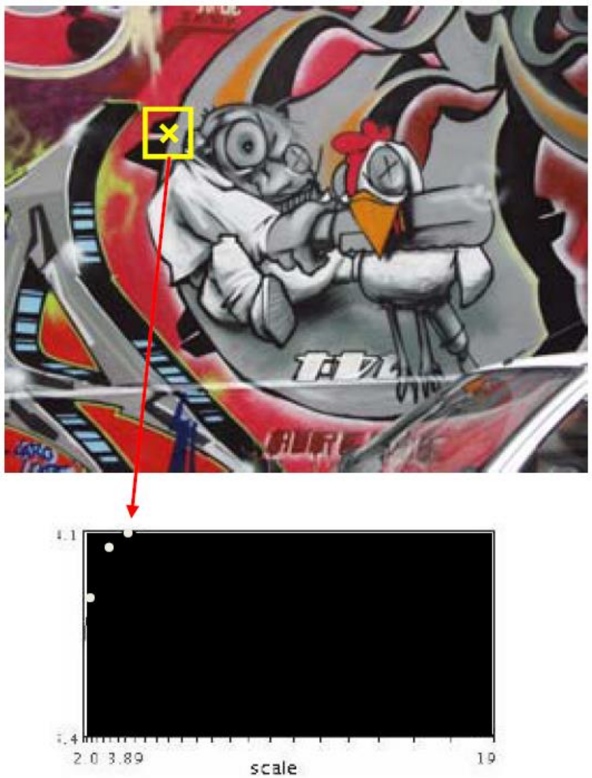
$f(I_{i_1 \ldots i_m}(x,\sigma))$

$f(I_{i_1 \ldots i_m}(x',\sigma))$

Signature Function

自动尺度选择

- Function responses for increasing scale (scale signature)

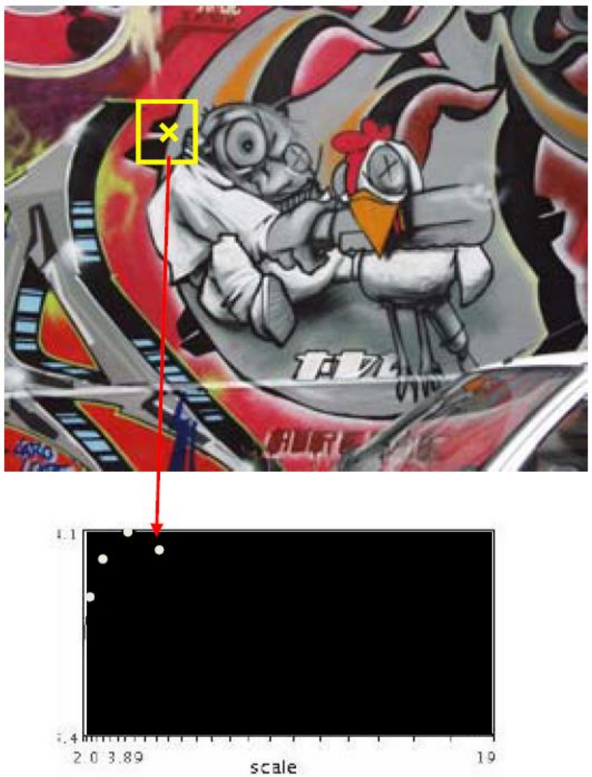


$f(I_{i_1 \ldots i_m}(x, \sigma))$

$f(I_{i_1 \ldots i_m}(x', \sigma))$

自动尺度选择

- Function responses for increasing scale (scale signature)

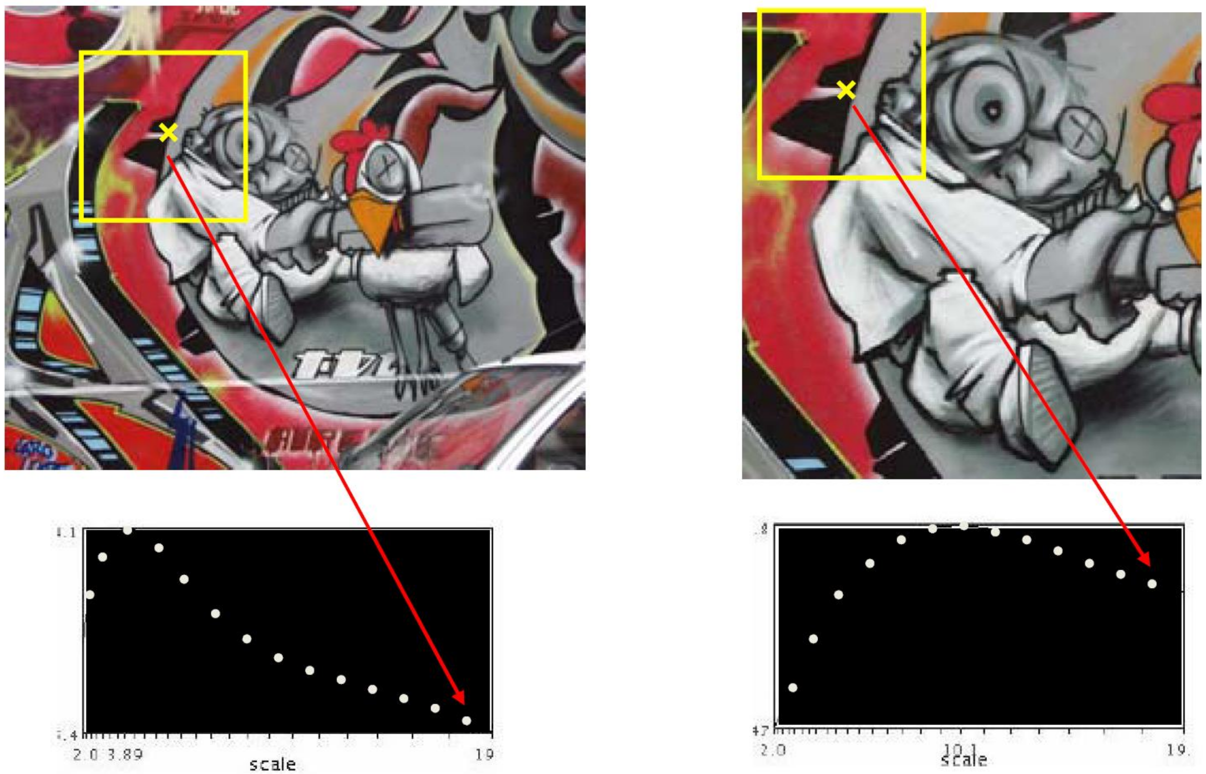


$f(I_{i_1 \ldots i_m}(x, \sigma))$

$f(I_{i_1 \ldots i_m}(x', \sigma))$

自动尺度选择

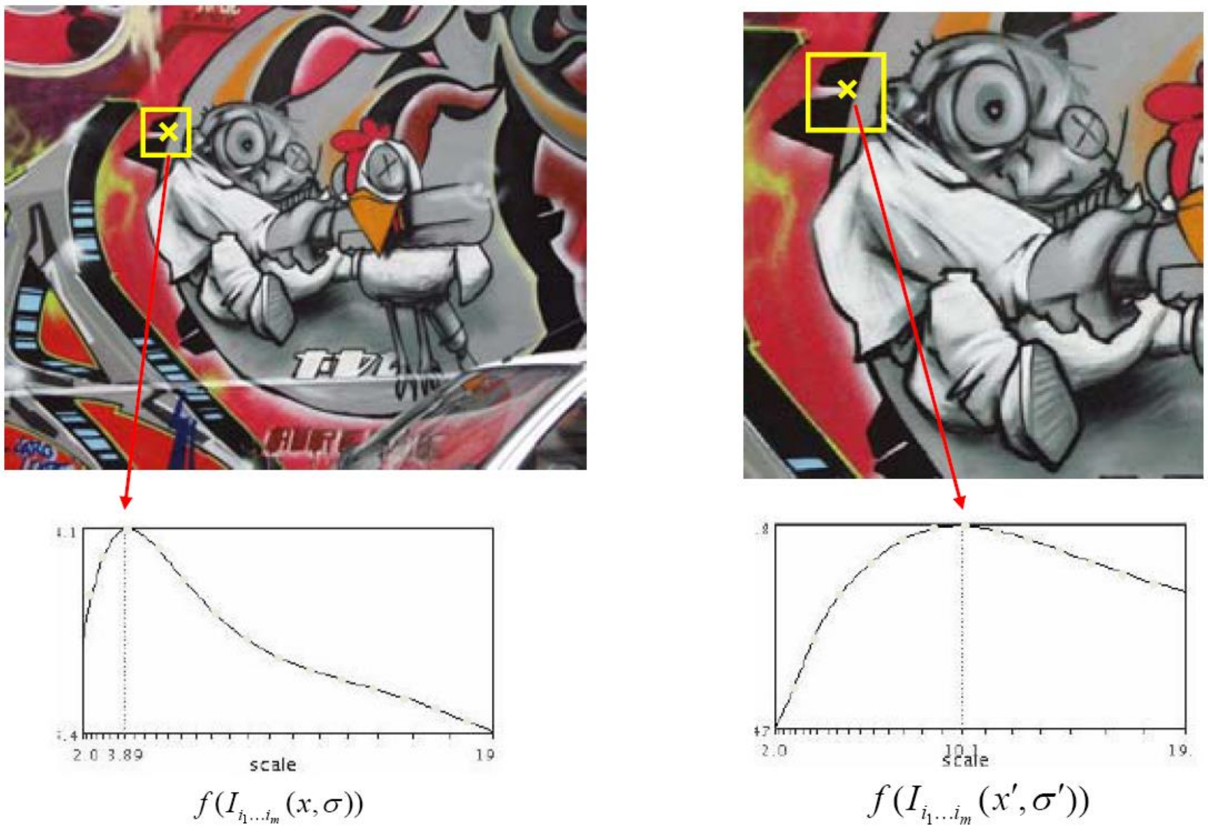
- Function responses for increasing scale (scale signature)



$f(I_{i_1 \ldots i_m}(x,\sigma))$ $f(I_{i_1 \ldots i_m}(x',\sigma))$

自动尺度选择

- Function responses for increasing scale (scale signature)

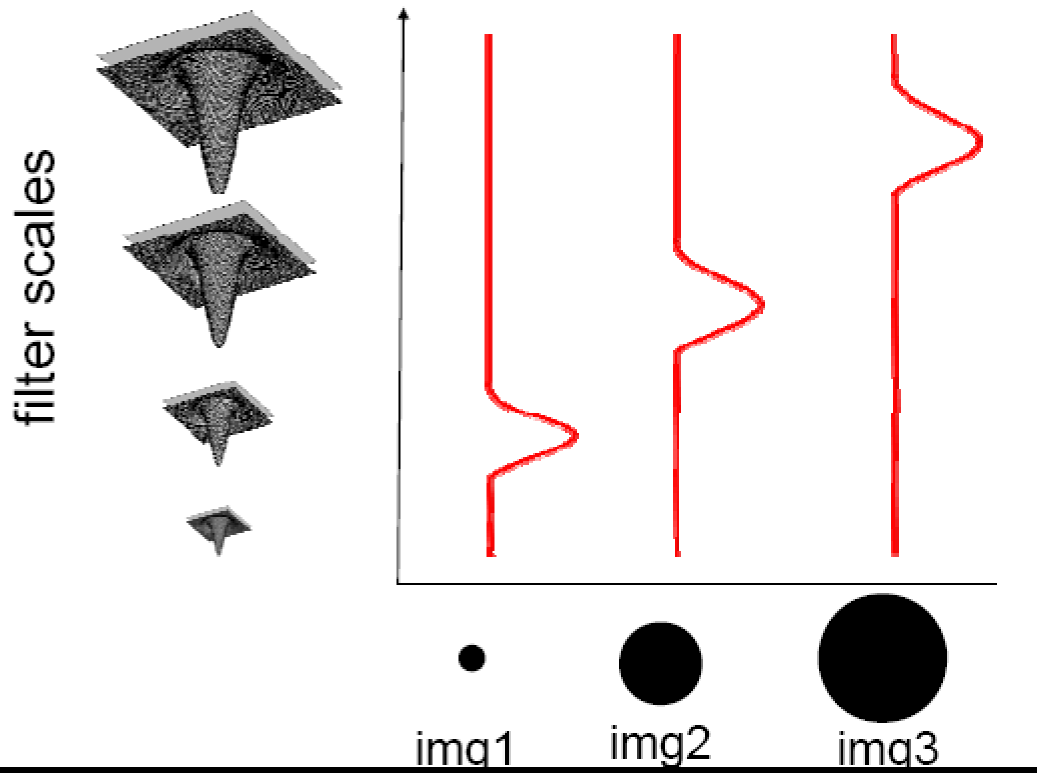


9.2.1 局部关键点检测

- 关键点检测子
  - 角点(corner)检测
    - ✓ Harris角点检测子
    - ✓ FAST角点检测子
  - 块(blob)检测
    - ✓ 高斯差分(DoG)检测子
    - ✓ SURF detector
    - ✓ MSER detector
    - ✓ Hessian-Affine detector

如何确定“signature” function *f* ?

Laplacian-of-Gaussian = “blob” detector

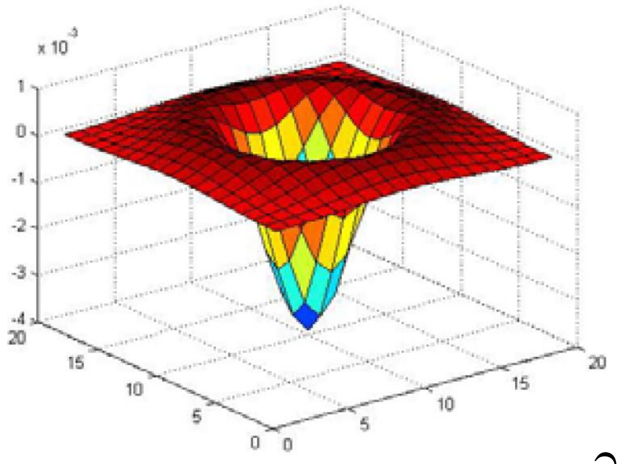


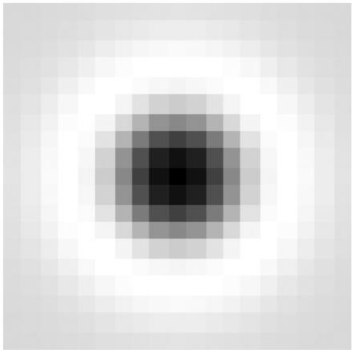
$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Laplacian-of-Gaussian (LoG)

- Circularly symmetric operator for blob detection in 2D.

LoG: "***blob***" detector



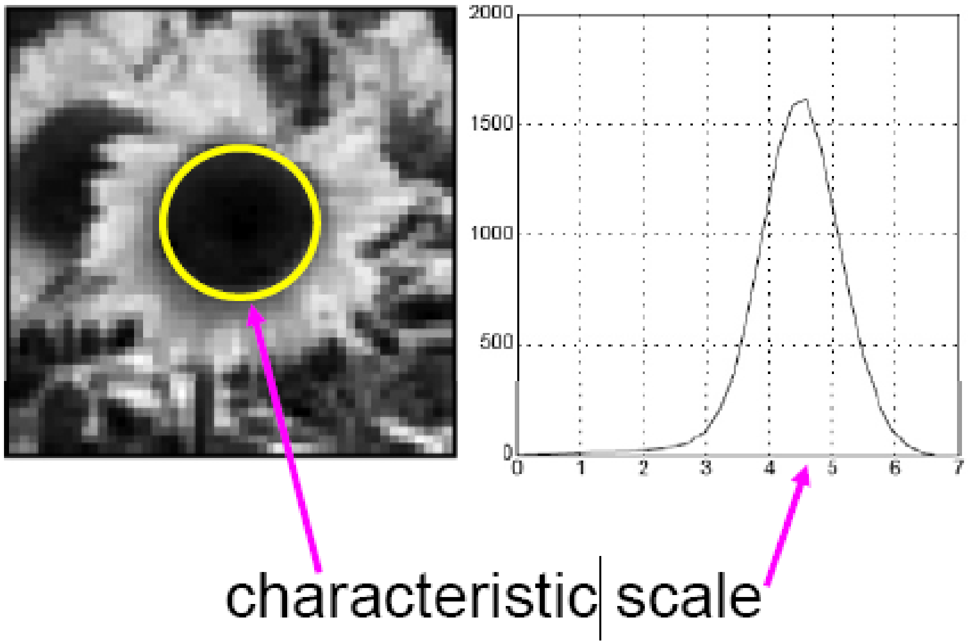


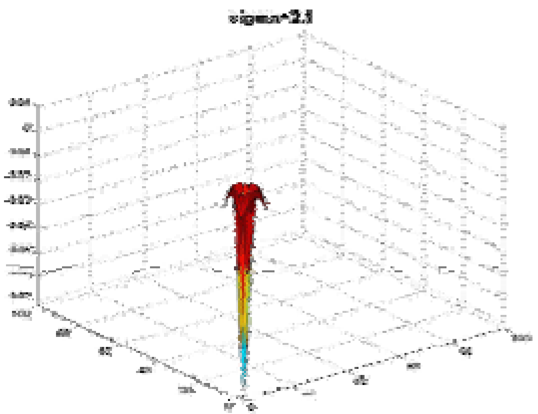
$$LoG = \nabla^2 G_\sigma(x,y) = \frac{\partial^2}{\partial x^2} G_\sigma(x,y) + \frac{\partial^2}{\partial y^2} G_\sigma(x,y)$$

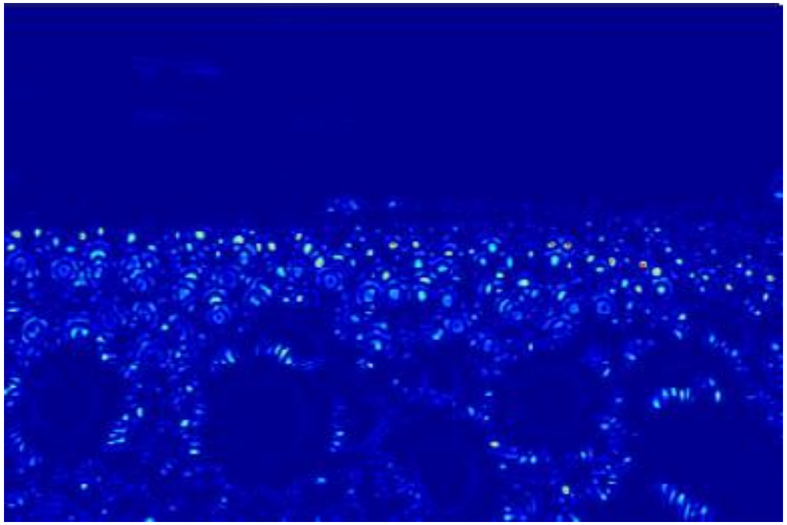
$$G_\sigma(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp[\frac{x^2+y^2}{2\sigma^2}]$$

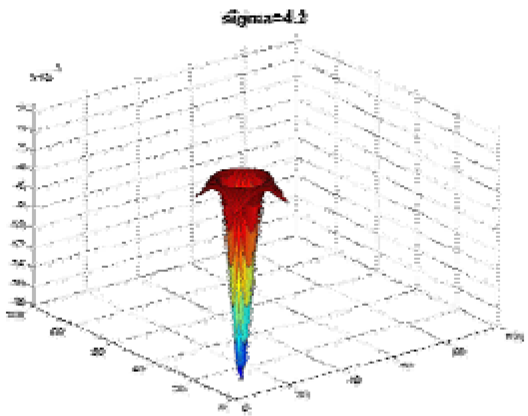
对于图像中的一个给定点:

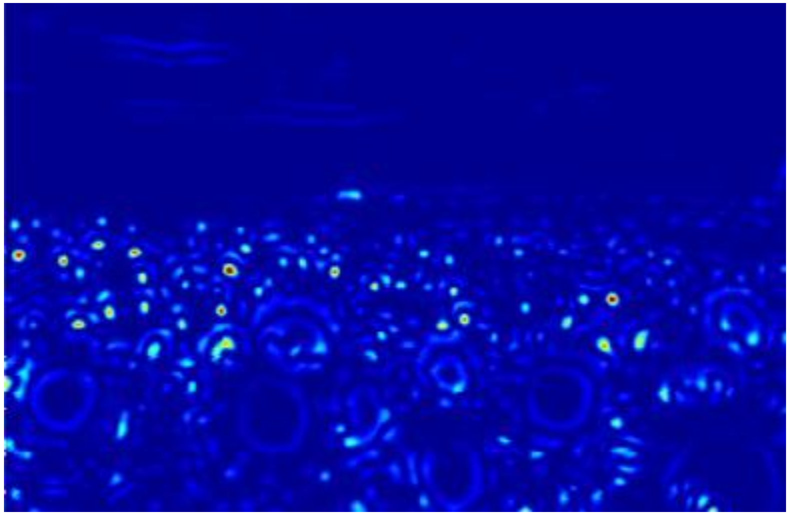
- 特征尺度 （*characteristic scale*）：Laplacian 响应的极值所对应的尺度

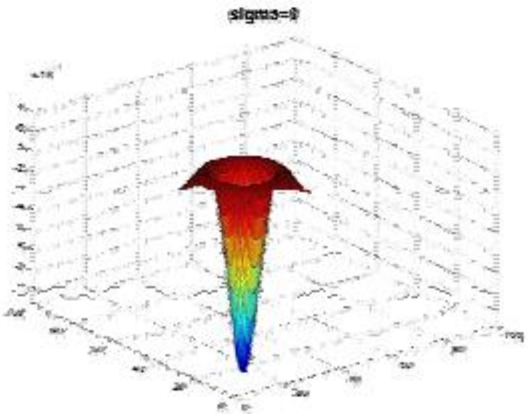


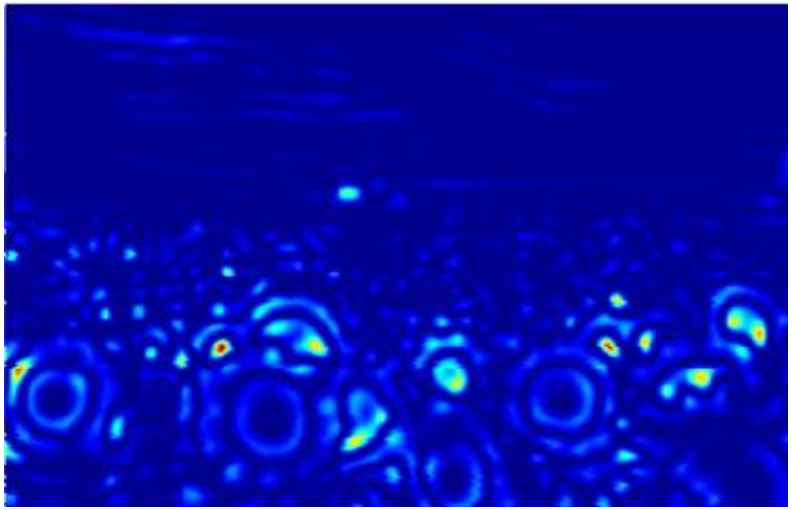

sigma=2.1

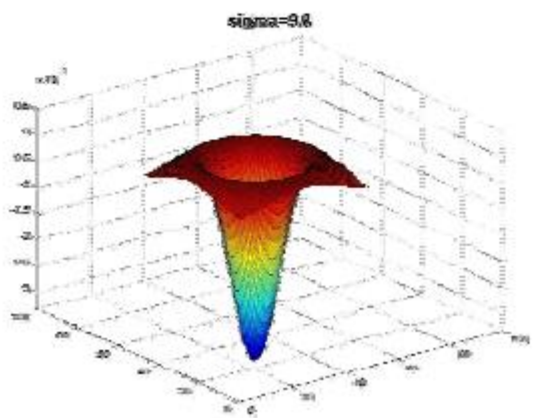


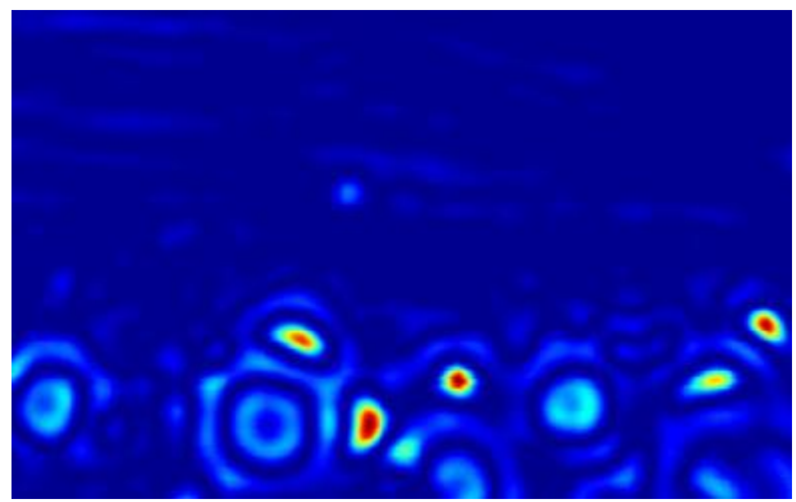

sigma=4.2

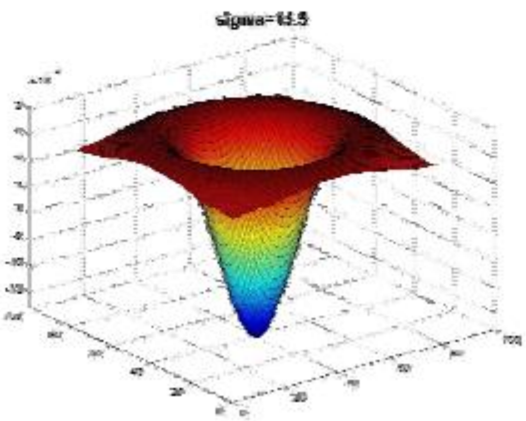


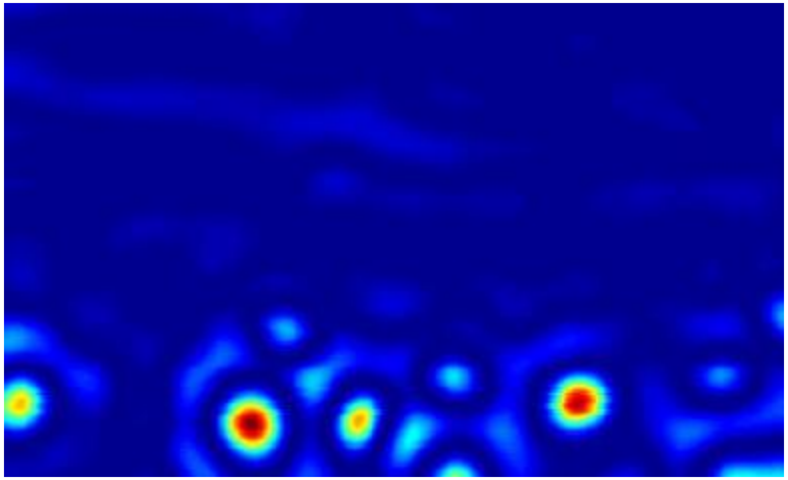


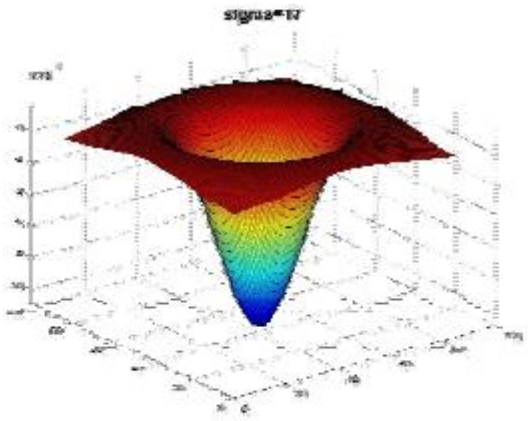


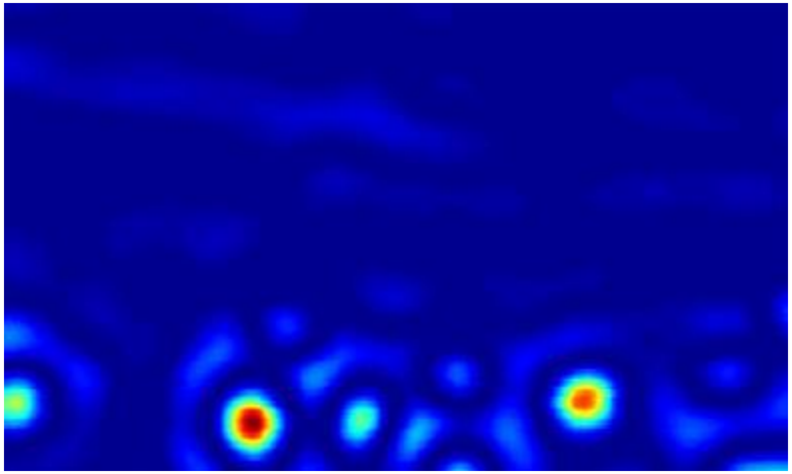




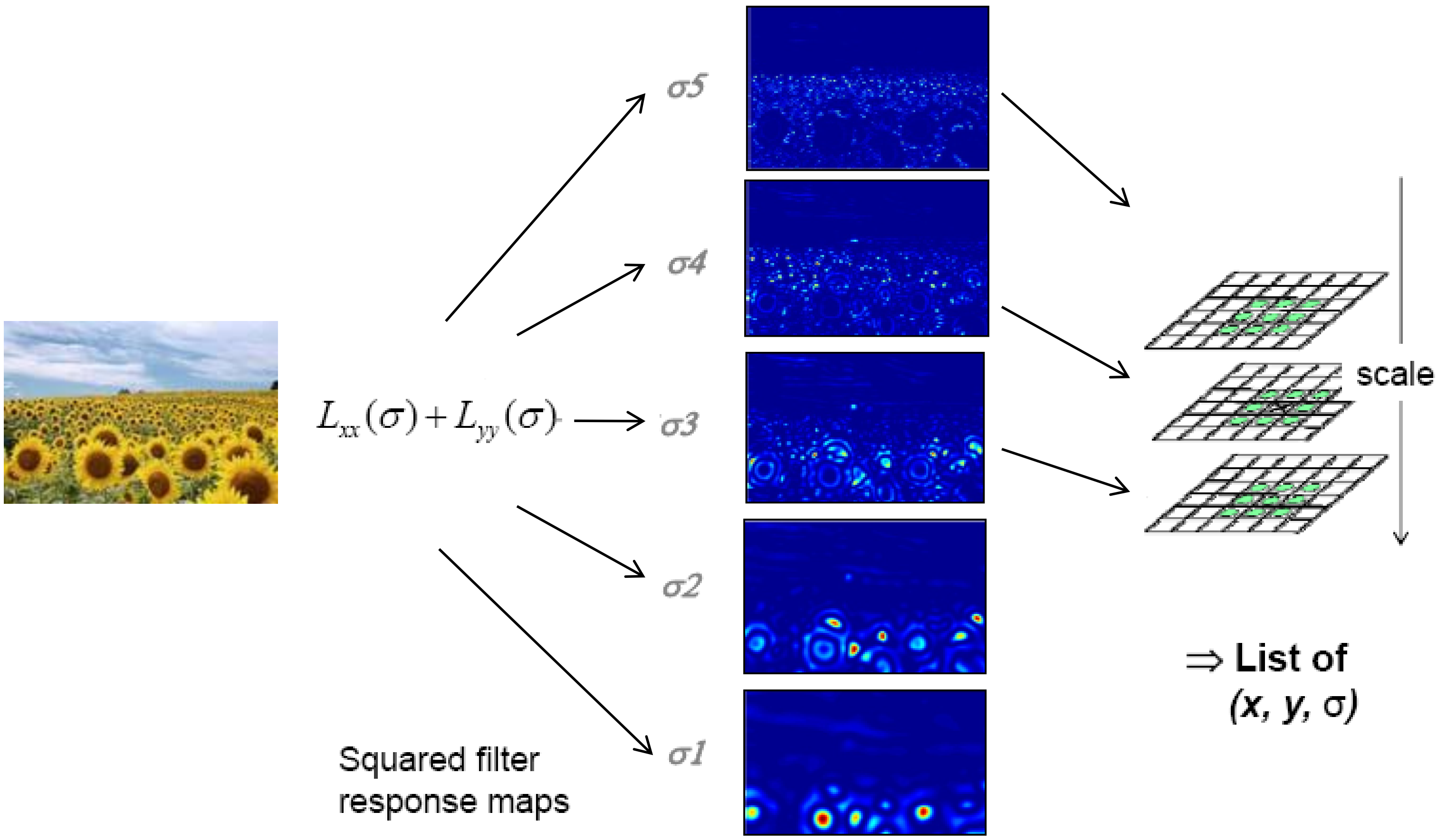




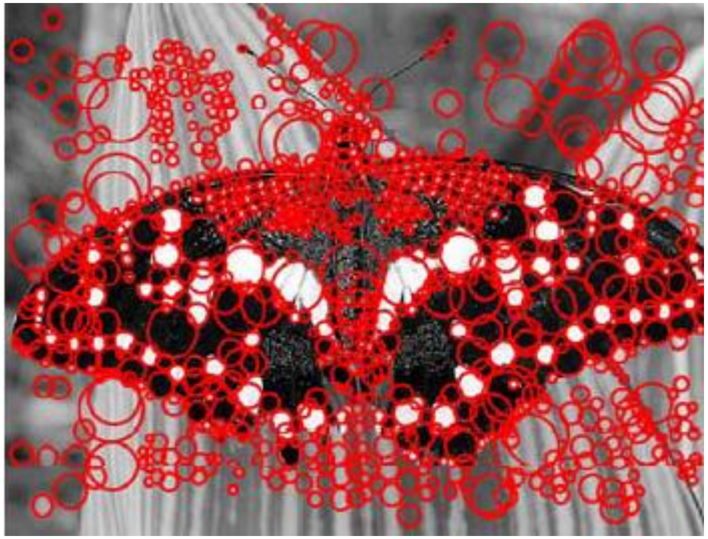




尺度空间的 blob 检测



尺度空间的 blob 检测: 示例



算法细节

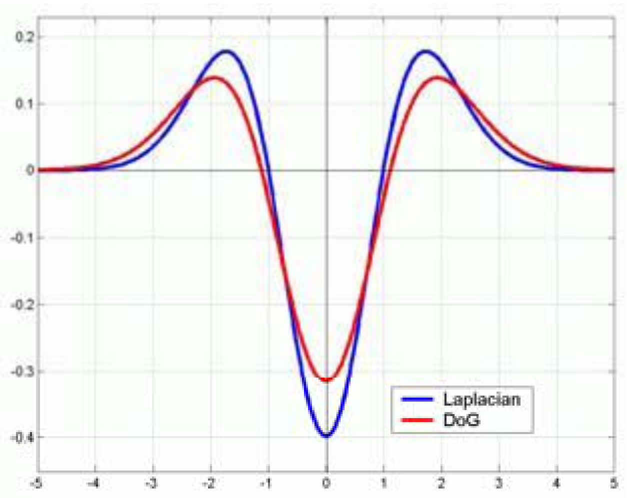
- 采用 Difference of Gaussians （DoG）去近似 Laplacian： more efficient to implement.

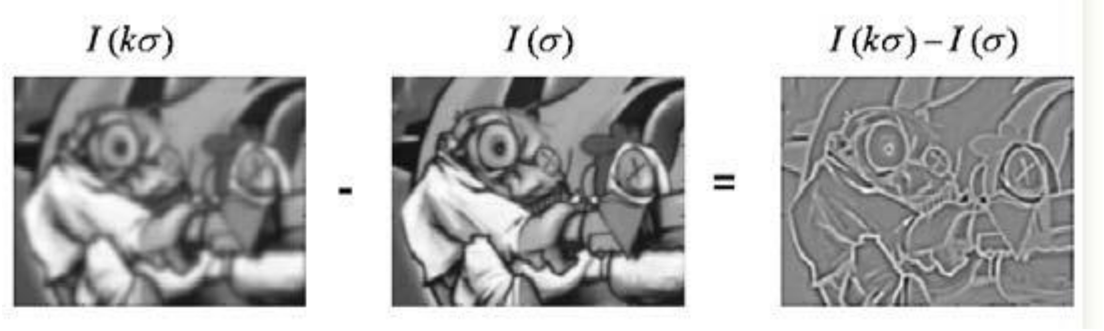
$$L = \sigma^2 \left(G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)$$

(Laplacian)

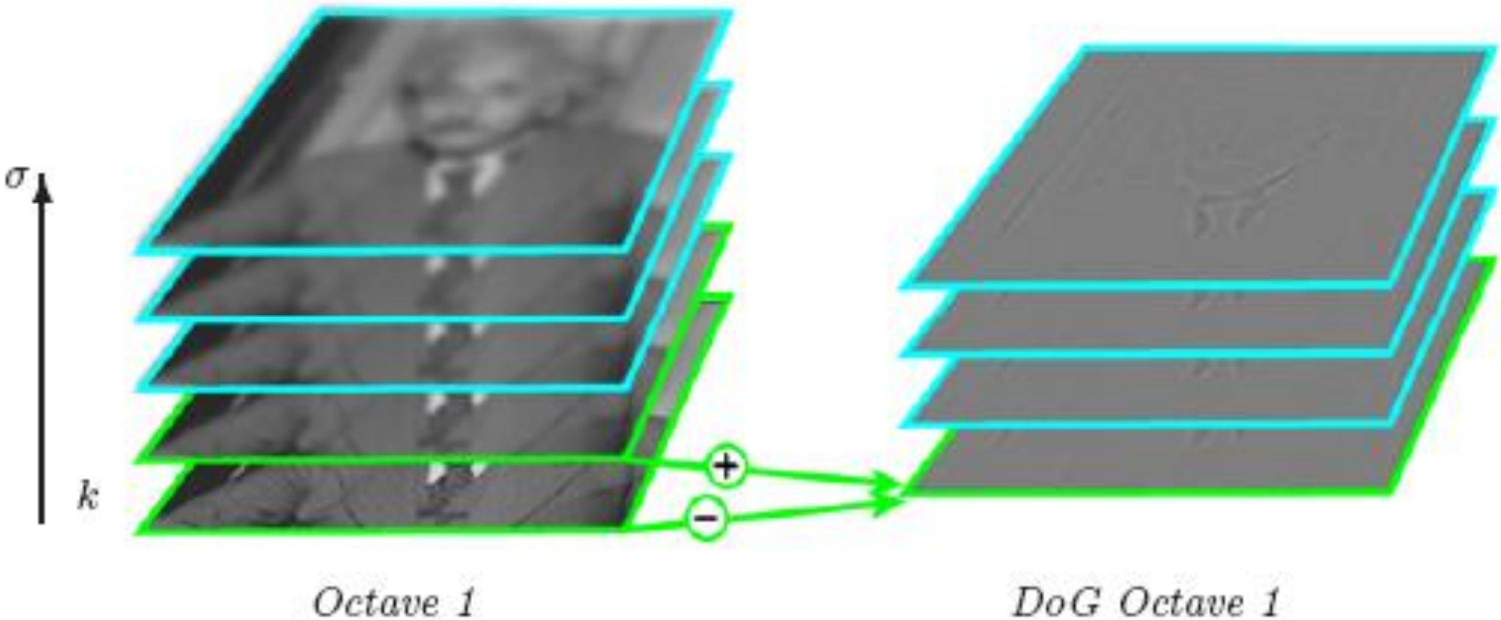
$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

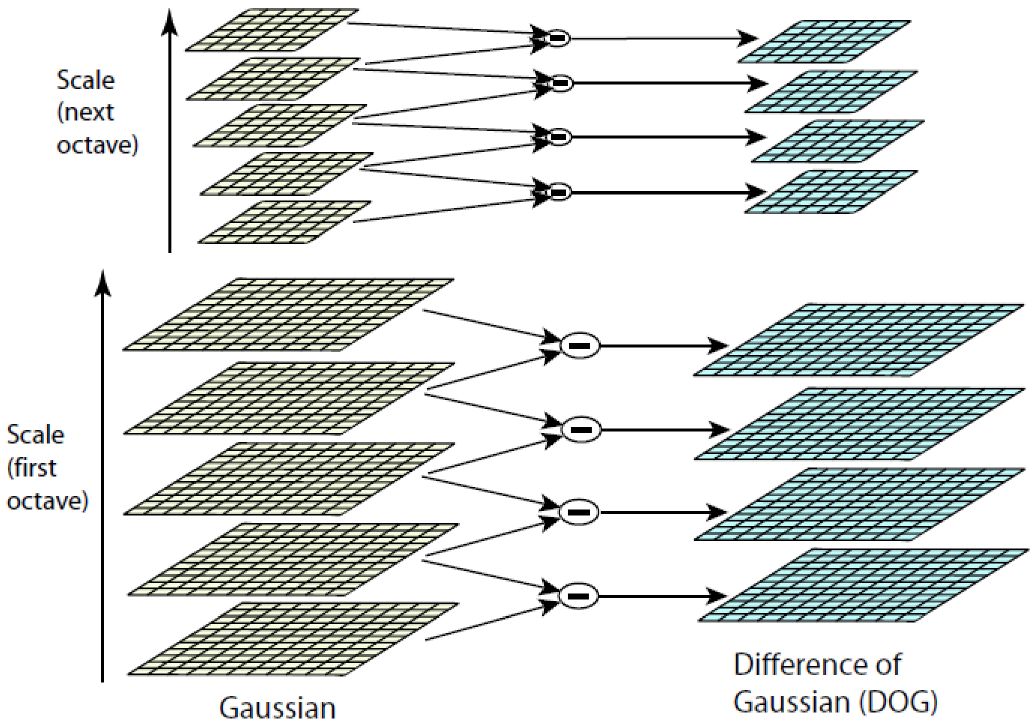
(Difference of Gaussians)



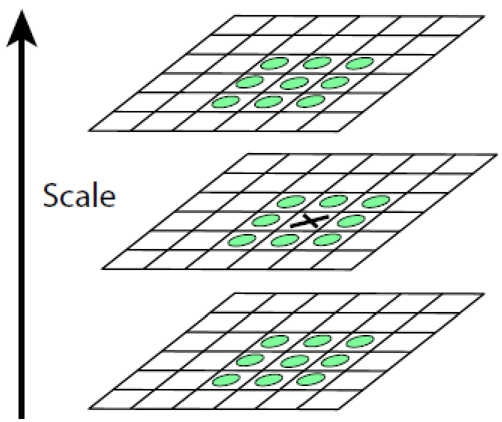


DoG Image Pyramid




Scale
(next
octave)
Scale
(first
octave)
Gaussian
Difference of
Gaussian (DOG)

(a)


Scale

(b)

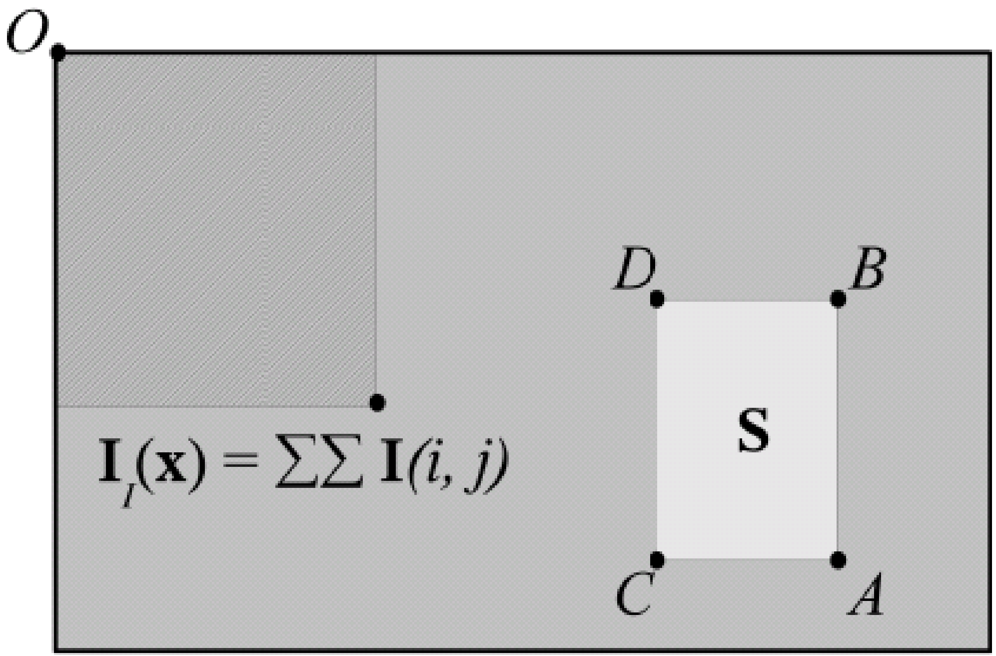
9.2.1 局部关键点检测

- 关键点检测子
  - 角点(corner)检测
    - Harris角点检测子
    - FAST角点检测子
  - 块(blob)检测
    - 高斯差分(DoG)检测子
    - SURF detector
    - MSER detector
    - Hessian-Affine detector

SURF: Speeded Up Robust Features

ECCV 2006, CVIU 2008

- Using integral images for major speed up
  - Integral Image (summed area tables) is an intermediate representation for the image and contains the sum of gray scale pixel values of image
  - They allow for fast computation of box type convolution filters.



$$\boxed{\mathbf{S} = A - B - C + D}$$

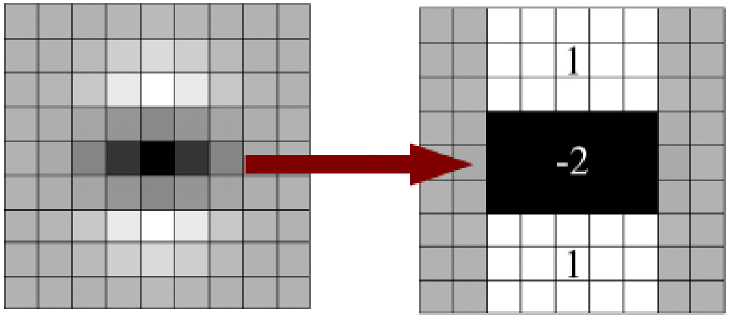
Cost *four* additions operation only to calculate the sum of the intensities over any upright, rectangular area

Detection

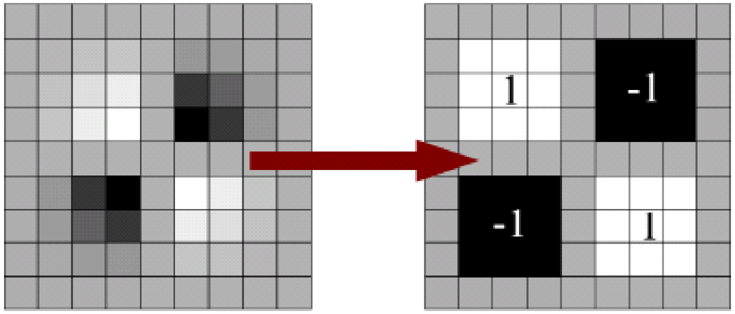
- Hessian-based interest point localization

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

- Approximated second order derivatives with box filters (mean/average filter)



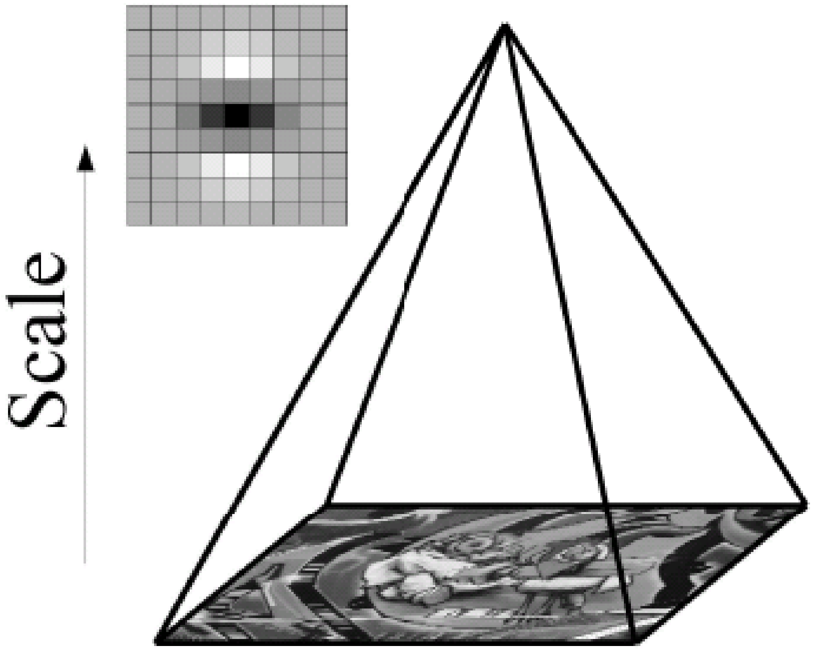
L_{yy}

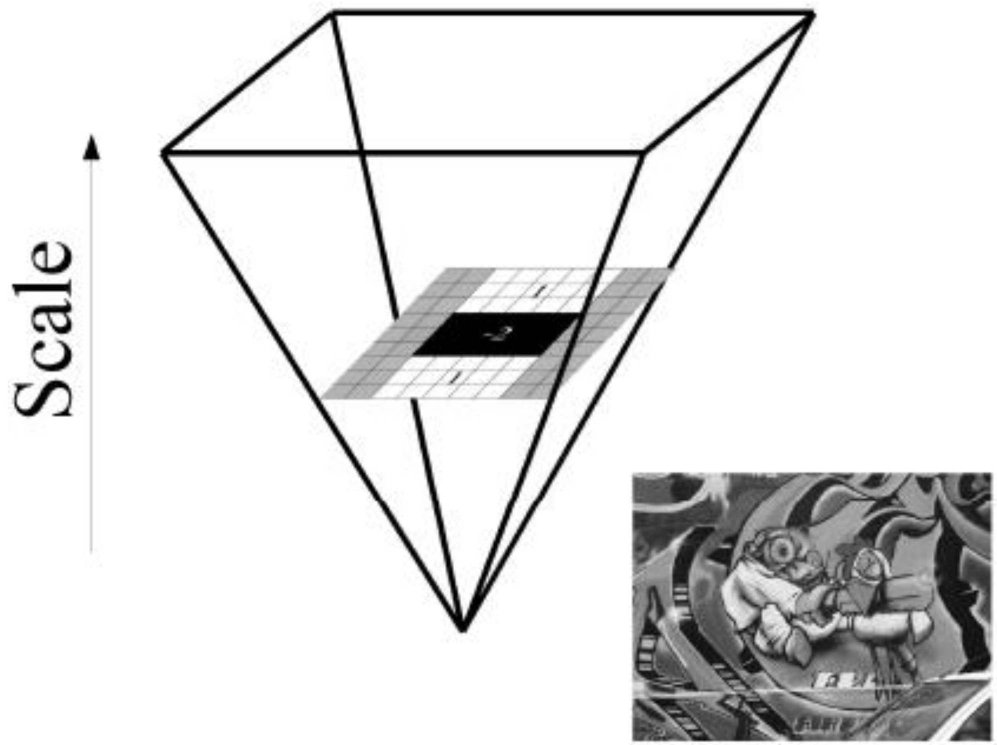


L_{xy}

Detection

- Scale analysis with constant image size



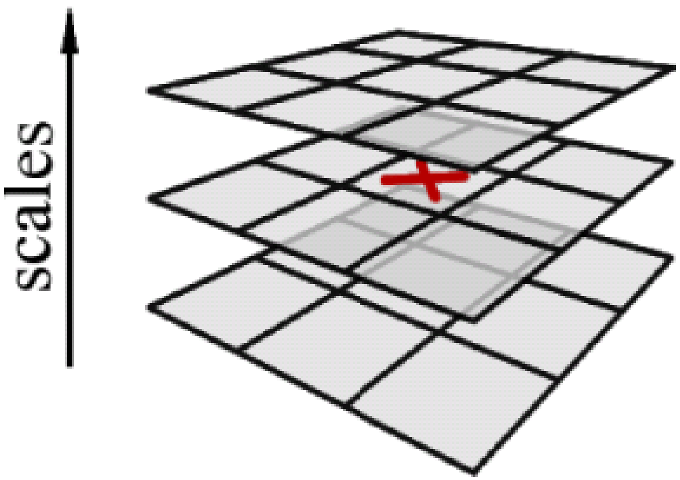


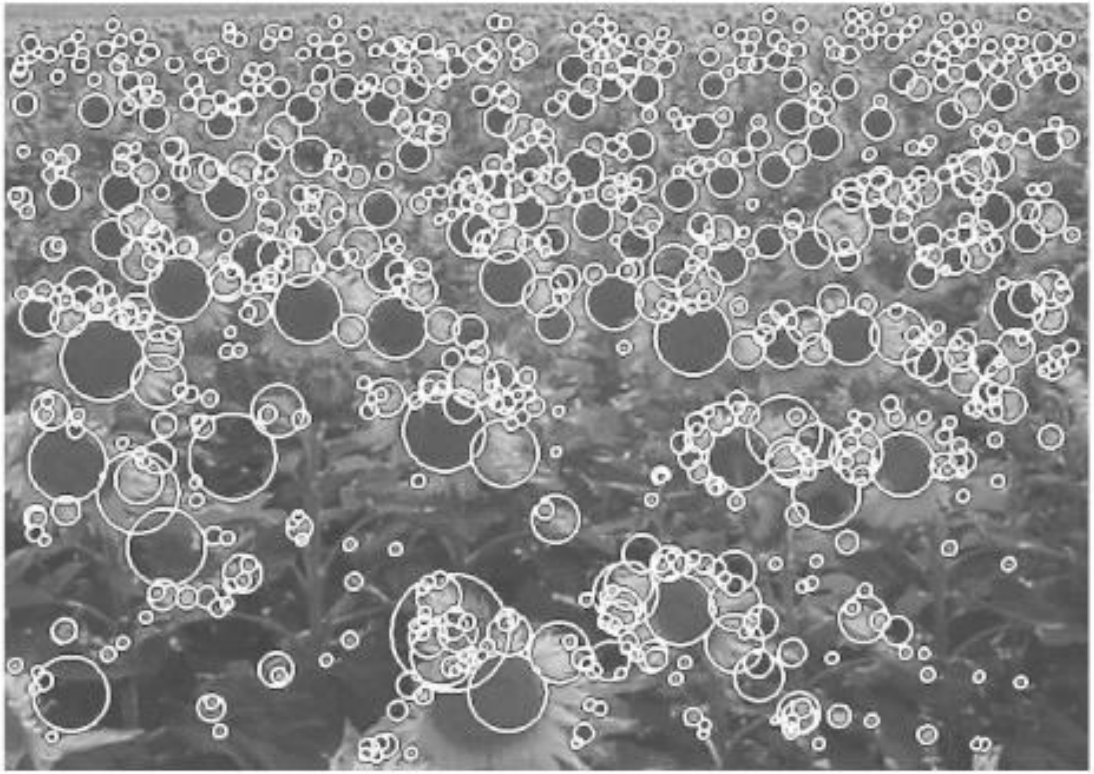
9 x 9, 15 x 15, 21 x 21, 27 x 27 → 39 x 39, 51 x 51 …

1st octave *2nd octave*

Detection

- Non-maximum suppression and interpolation
  - Blob-like feature detector





9.2.1 局部关键点检测

- 关键点检测子
  - 角点(corner)检测
    - ✓ Harris角点检测子
    - ✓ FAST角点检测子
  - 块(blob)检测
    - ✓ 高斯差分(DoG)检测子
    - ✓ SURF detector
    - ✓ MSER detector
    - ✓ Hessian-Affine detector

Blob Detector: MSER

- Maximally Stable Extremal Region

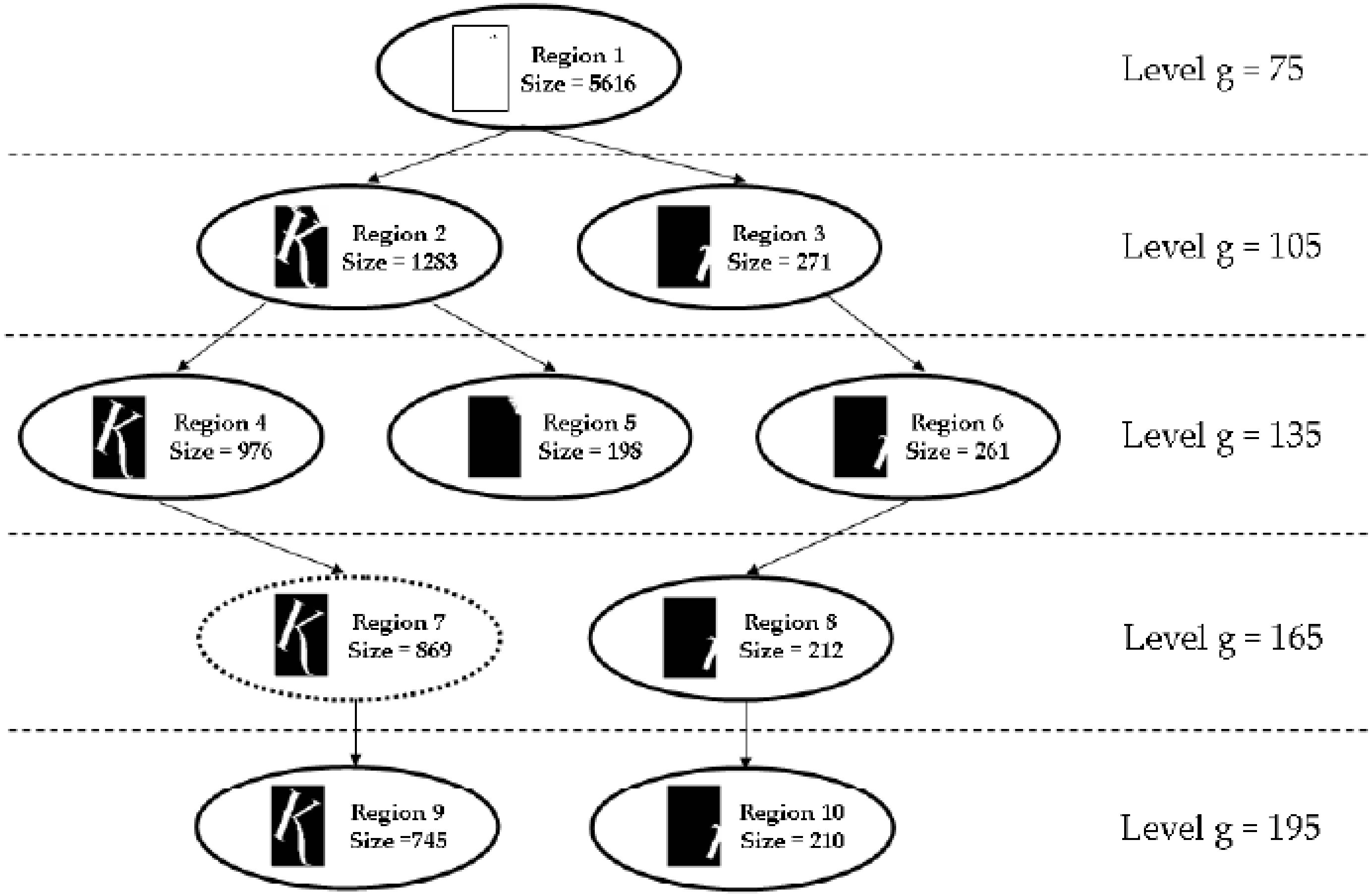
(a) Input (b) $g = 75$ (c) $g = 105$ (d) $g = 135$

(e) $g = 165$ (f) $g = 195$ (g) $g = 225$ (h) $g = 255$

$$\forall p \in R_i \ , \forall q \in boundary(R_i) \rightarrow I_{in}(p) \geq I_{in}(q)$$

$$I_{bin}^{g} = \begin{cases} 1 & I_{in} \geq g \\ 0 & otherwise \end{cases}$$

Blob Detector: MSER



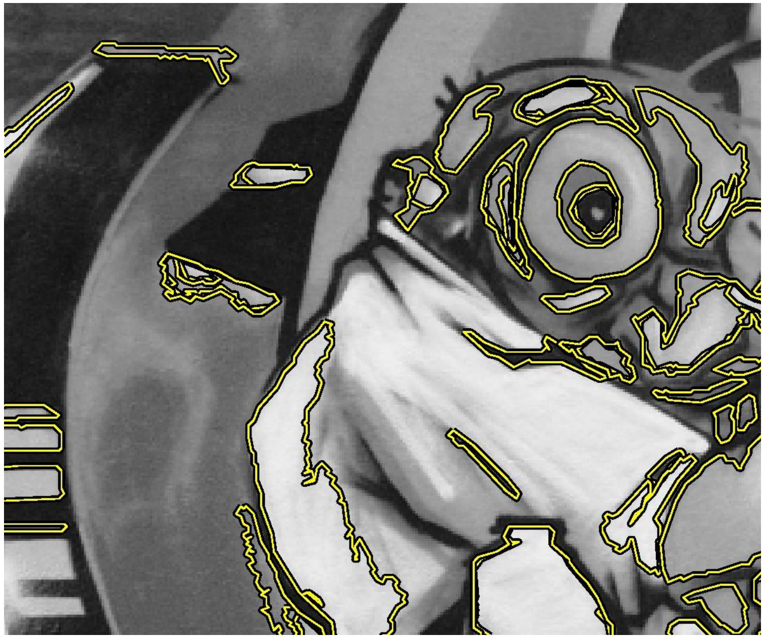
Maximally stable extremal regions (MSER)

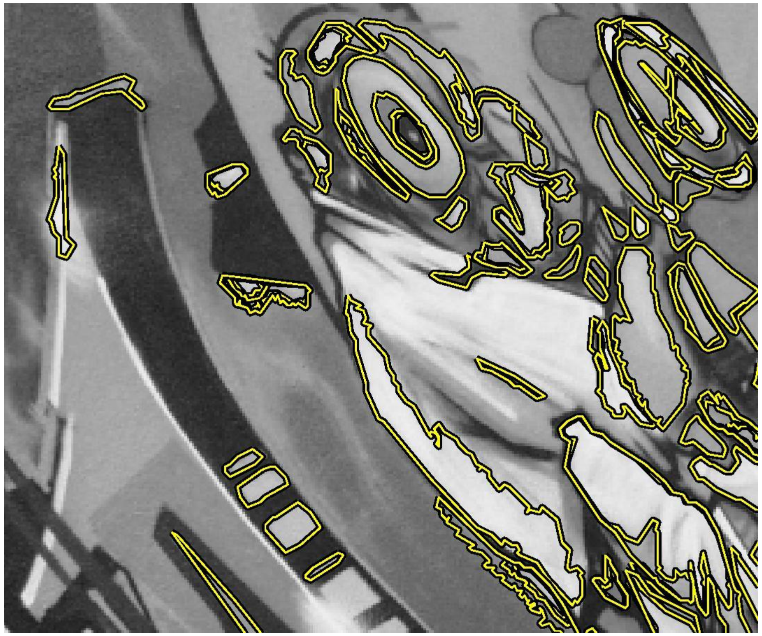
- Examples of thresholded images

high threshold

low threshold

MSER



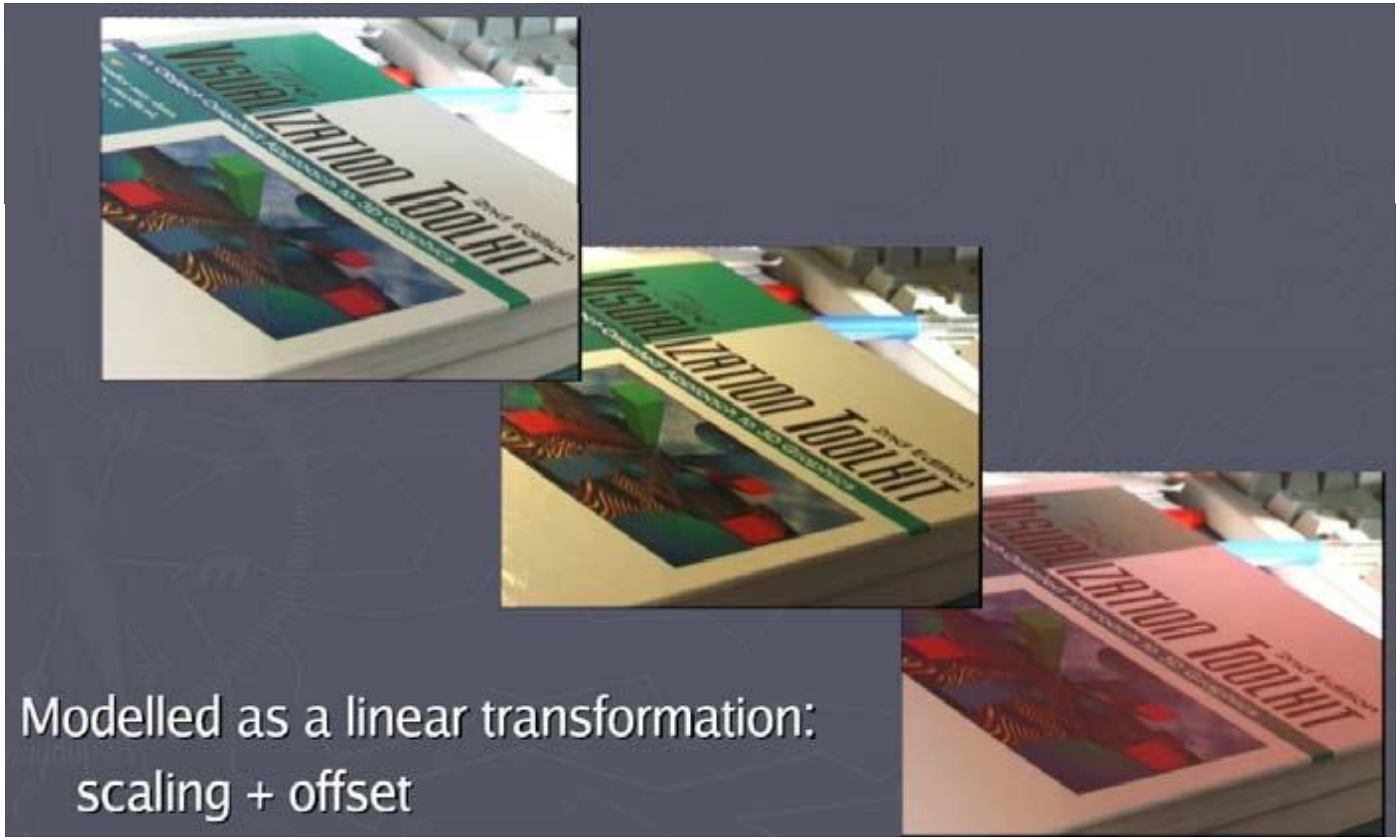


9.2.1 图像局部区域描述

- 目标
  - 将图像局部区域变换为固定长度的特征向量
  - 不变性：旋转、亮度
- 特征类型
  - 浮点型特征
    - ✓ SIFT, GLOH, SURF, LIOP,
  - 二值型特征
    - ✓ BRIEF, ORB, FREAK, BRISK, CARD, Edge-SIFT

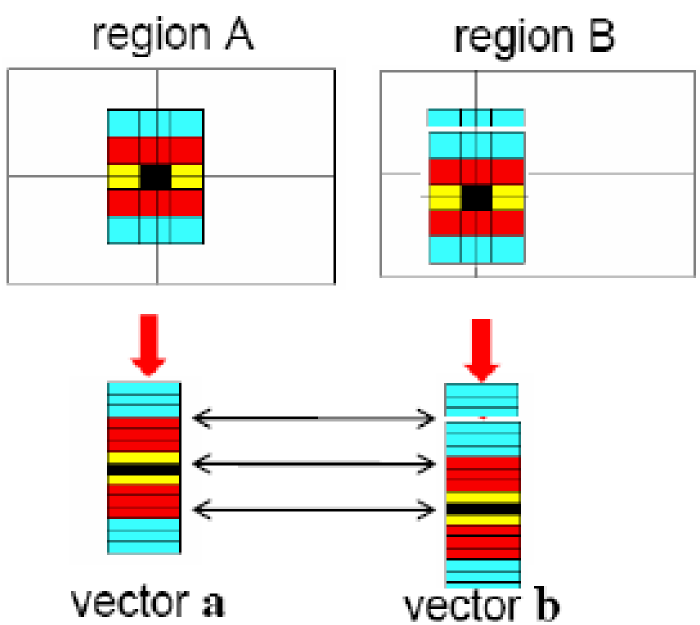
几何变换

光照变化



原始的灰度块作为局部描述?

- 最简单的方式：将特征点邻域的像素灰度值逐个排列，得到一个特征向量
- 但是，这种方式对微小平移和旋转非常敏感！！！

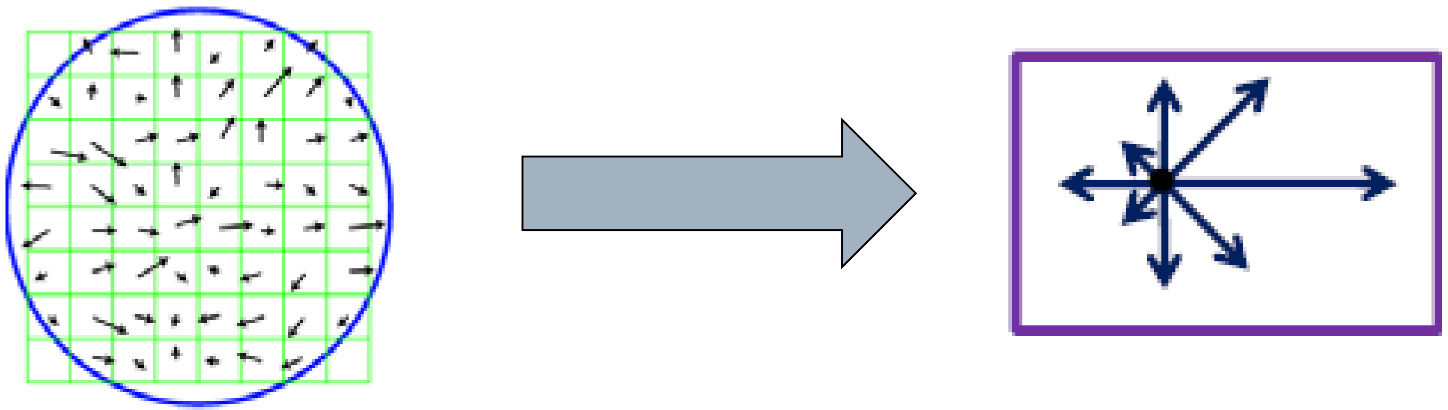


9.2.1 图像局部区域描述

- 目标
  - 将图像局部区域变换为固定长度的特征向量
  - 不变性：旋转、亮度
- 特征类型
  - 浮点型特征
    - ✓ SIFT, GLOH, SURF, LIOP
  - 二值型特征
    - ✓ BRIEF, ORB, FREAK, BRISK, CARD, Edge-SIFT

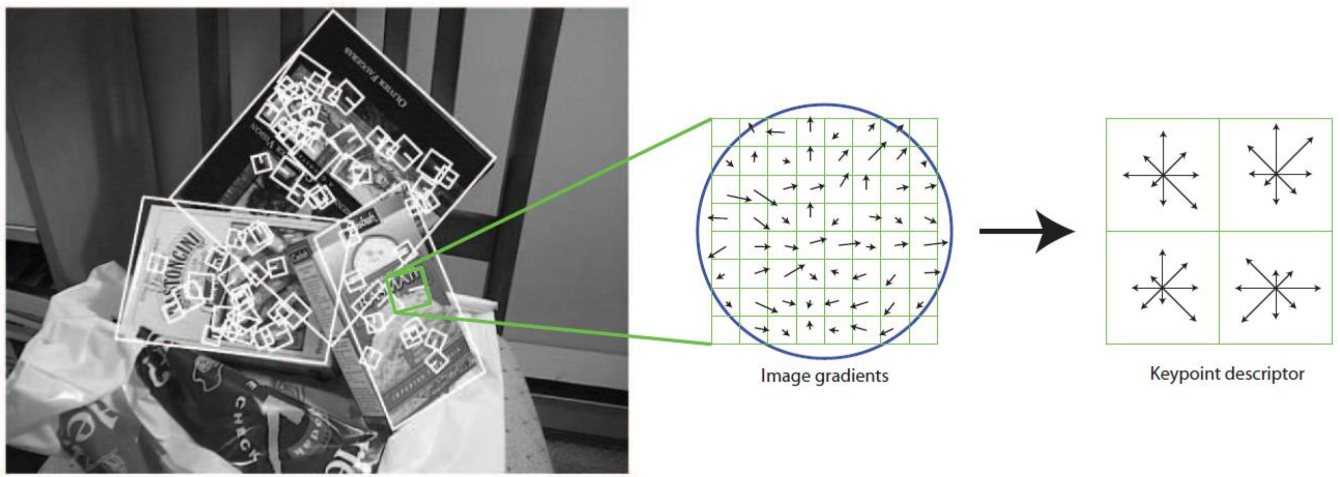
SIFT描述子：方向归一化

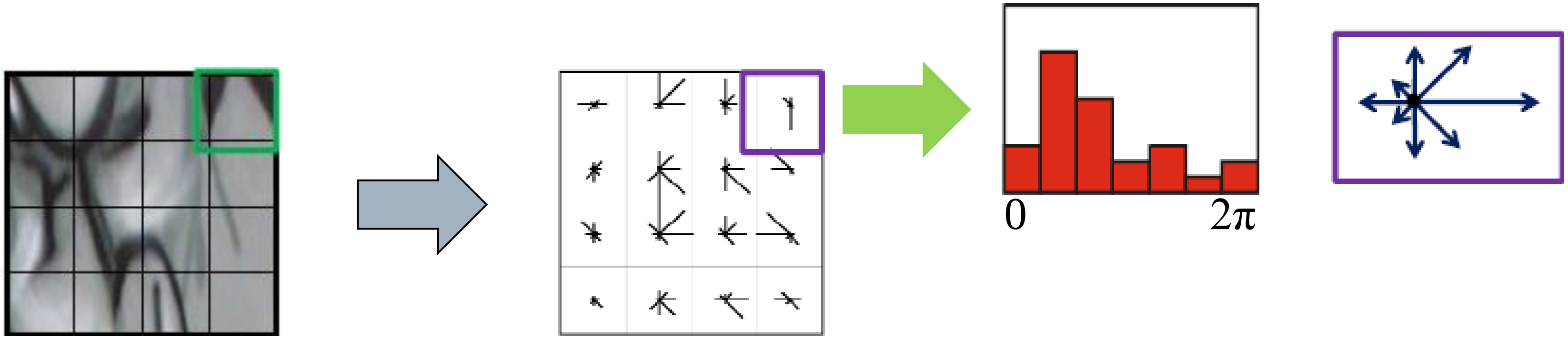
- 方向旋转不变: 主方向(dominant orientation)对齐
  - 36个方向
  - 根据梯度幅值和与特征点的距离加权(1.5σ)
  - 多个主方向



SIFT描述子[Lowe 2004]

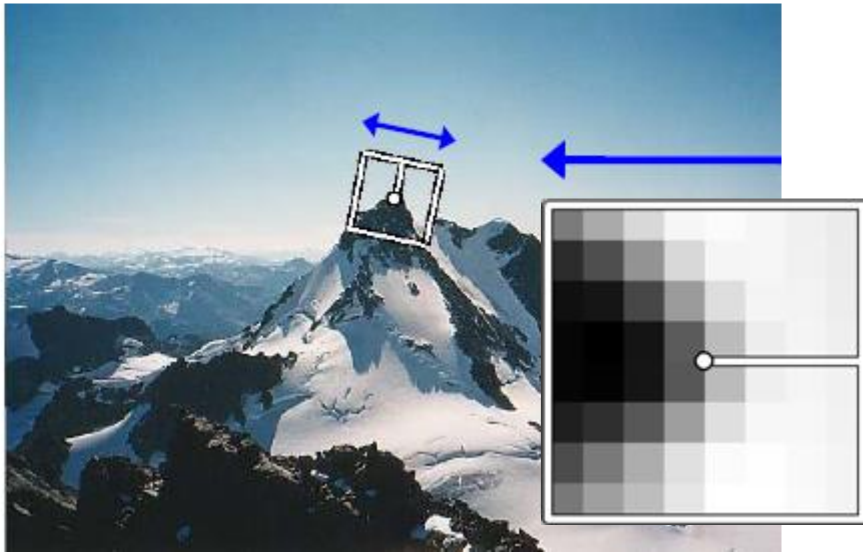
- 对每个子块，采用基于梯度方向的直方图表示特征.





SIFT描述子：旋转不变性

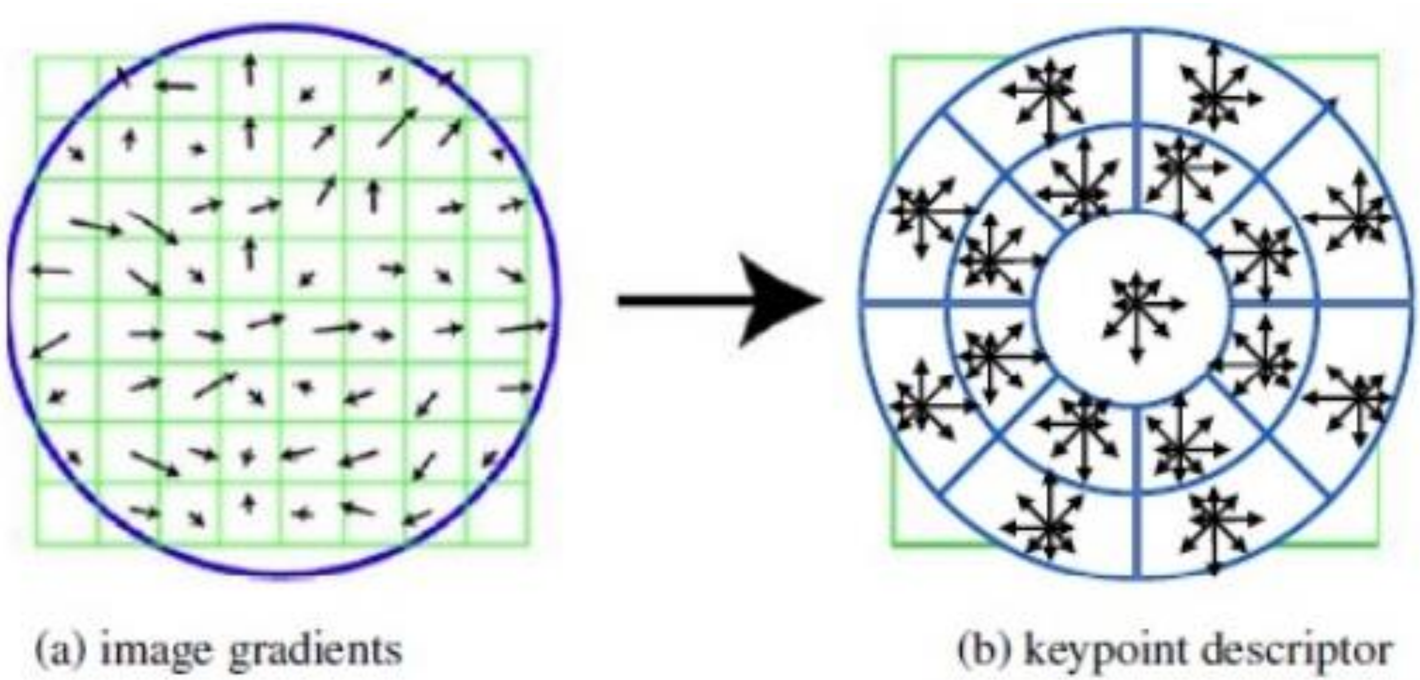
- 根据patch的梯度主方向，对patch进行旋转
- 将patches 修正到一个canonical orientation.



SIFT 描述子 [Lowe 2004]

- 非常鲁棒的匹配技术
  - 良好的不变性
    - ✓ 平移、旋转、缩放（尺度变化）
  - 可以处理视角变化
    - ✓ Up to about 60 degree out of plane rotation
  - 可以处理显著的光照明暗变化
    - ✓ Sometimes even day vs. night
- 网上代码下载
  - http://people.csail.mit.edu/albert/ladypack/wiki/index.php/Known_implementations_of_SIFT

GLOH: Gradient location-orientation histogram (Mikolajczyk and Schmid 2005)

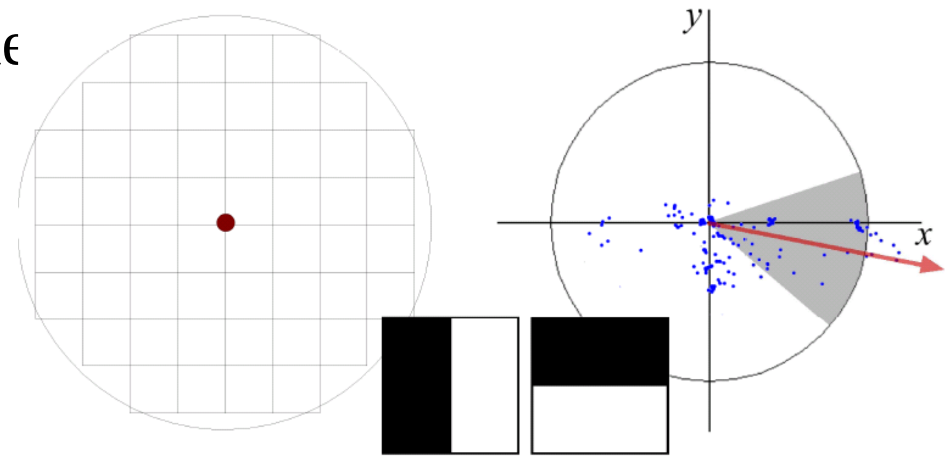


SIFT GLOH

272D → 128D by PCA

SURF: Speeded Up Robust Features

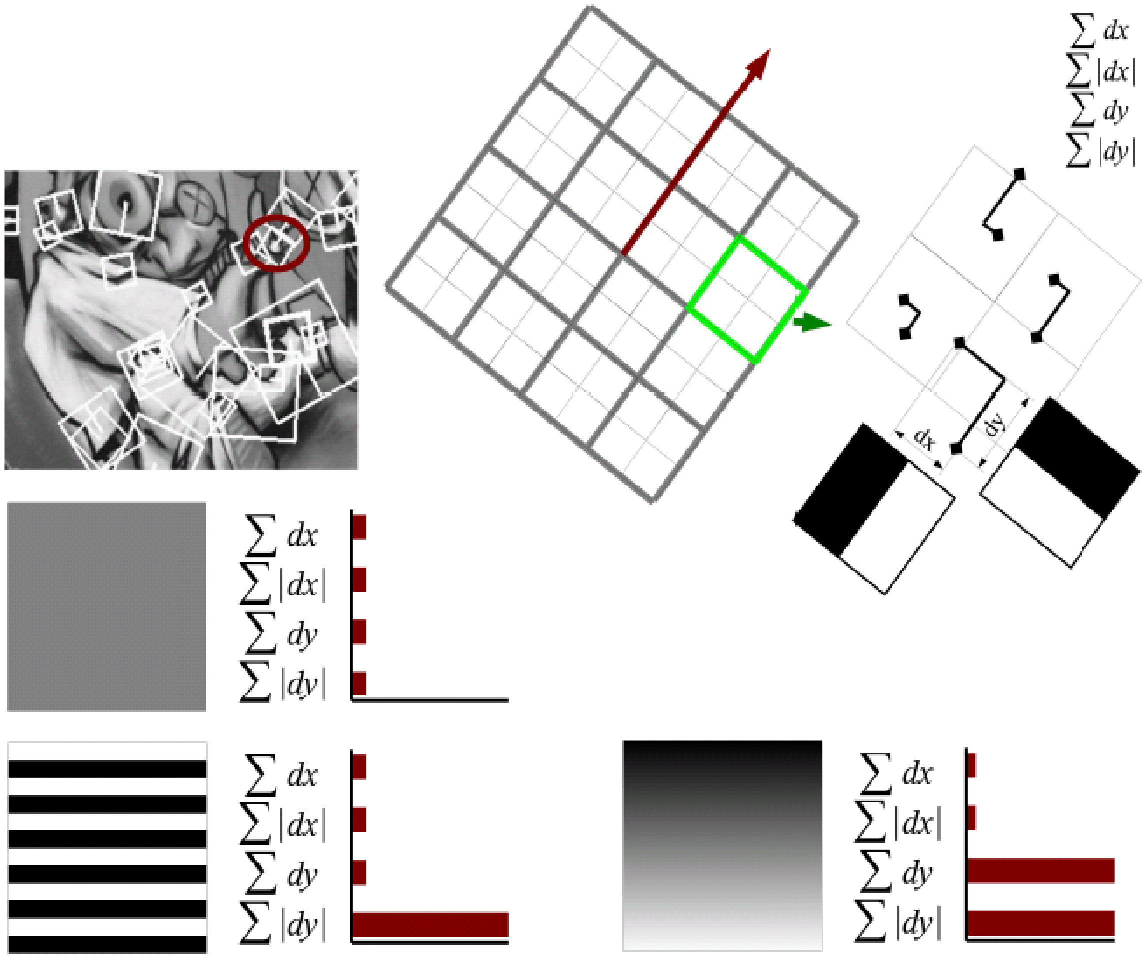
- Dominant orientation
  - The Haar wavelet responses are represented as vectors
  - Sum all responses within a sliding orientation window covering an angle of 60 degree
  - The two summed response yield a new vector
  - The longest vector is the dominant orientation
  - Second longest is … ignored



SURF描述子

- Split the interest region up into 4 x 4 square sub-regions with 5 x 5 regularly spaced sample points inside
- Calculate Haar wavelet response dx and dy
- Weight the response with a Gaussian kernel centered at the interest point
- Sum the response over each sub-region for dx and dy separately → feature vector of length 32
- In order to bring in information about the polarity of the intensity changes, extract the sum of absolute value of the responses → feature vector of length 64
- Normalize the vector into unit length

SURF描述子



SURF描述子

- SURF-128
  - The sum of dx and |dx| are computed separately for dy < 0 and dy >0
  - Similarly for the sum of dy and |dy|
  - This doubles the length of a feature vector

LIOP: Local Intensity Order Pattern for Feature Description (ICCV 2011)

- Motivation
  - Orientation estimation error in SIFT

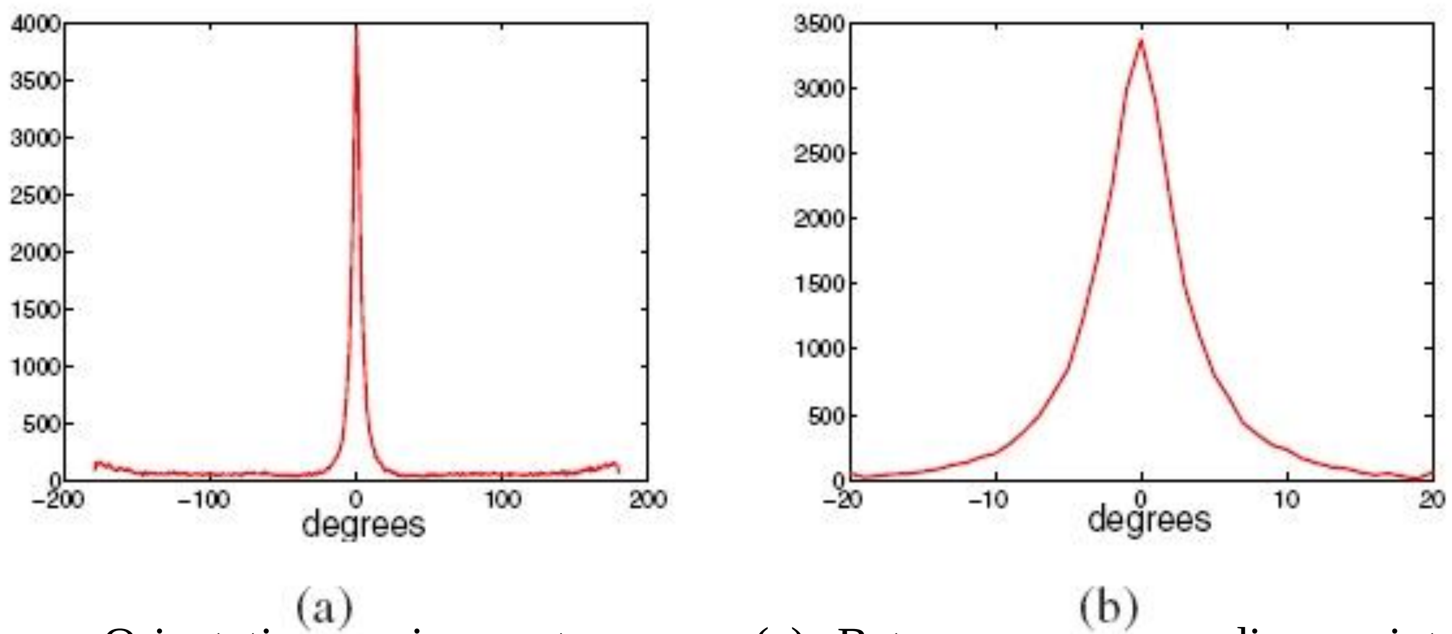
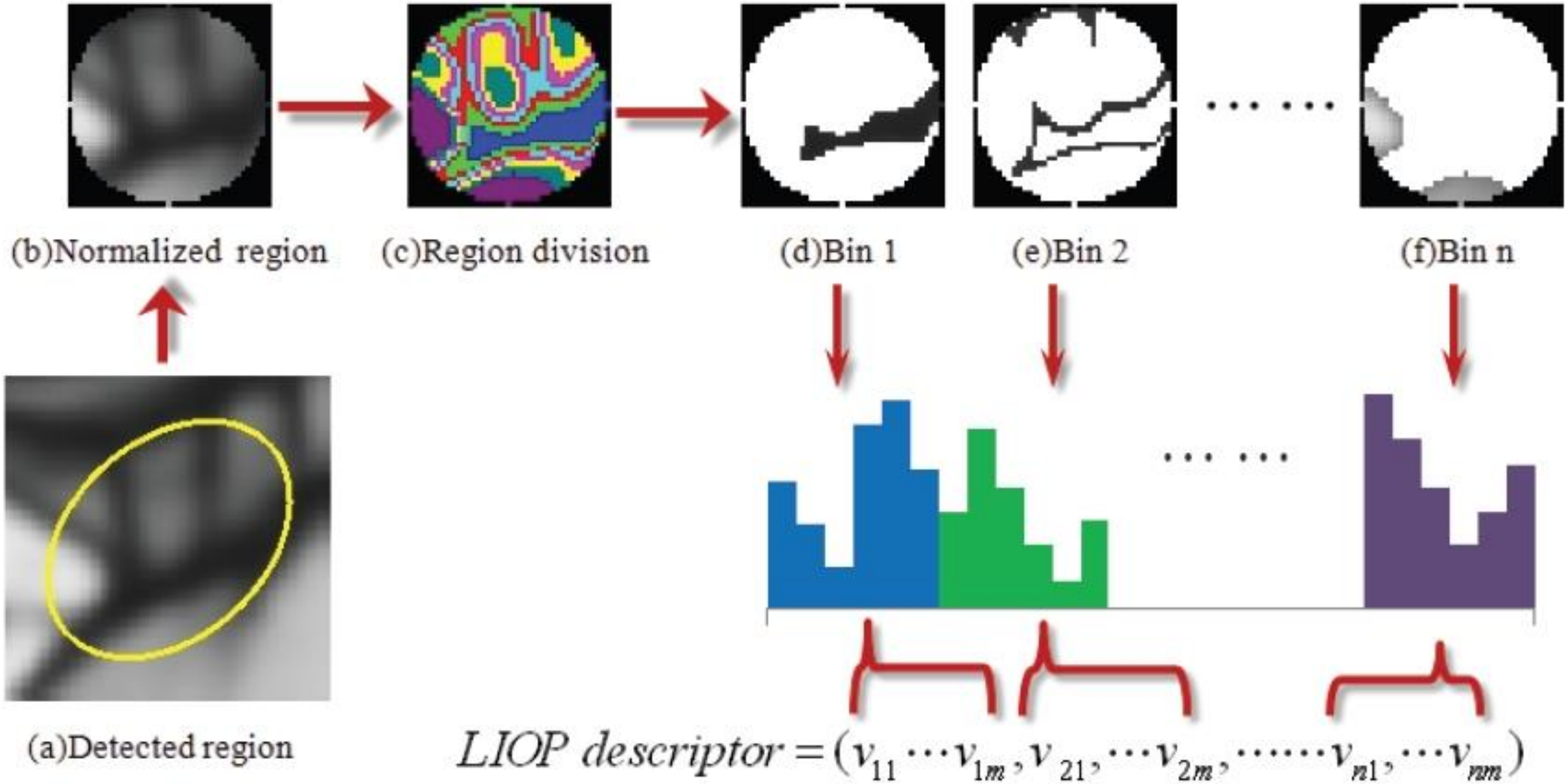
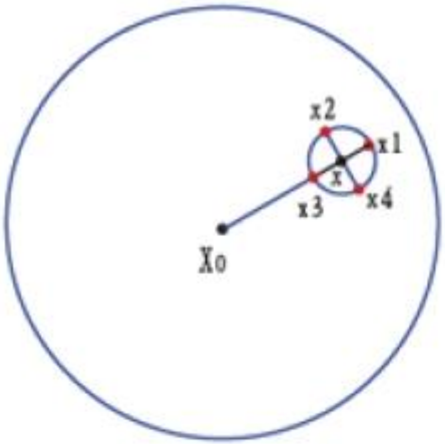


Figure. Orientation assignment errors. (a) Between corresponding points, only 63.77% of errors are in the range of [-20,20]. (b) Between corresponding points that are also matched by SIFT descriptors.

LIOP: Local Intensity Order Pattern for Feature Description



LIOP: Local Intensity Order Pattern for Feature Description



(a) Original patch

For a point x:

$$P(x) = (I(x_1), I(x_2), I(x_3), I(x_4)) = (86, 217, 152, 101)$$

$$\gamma(P(x)) = (1,4,3,2)$$

$$LIOP(x) = \phi(\gamma(P(x))) = (0,0,0,0,0,1,0,\cdots,0)$$

histogram of bin i

π	$Ind(\pi)$
1,2,3,4	1
1,2,4,3	2
1,3,2,4	3
1,3,4,2	4
1,4,2,3	5
1,4,3,2	6
2,1,3,4	7
2,1,4,3	8
⋮	⋮
4,3,1,2	23
4,3,2,1	24

9.2.1 图像局部区域描述

- 目标
  - 将图像局部区域变换为固定长度的特征向量
  - 不变性：旋转、亮度
- 特征类型
  - 浮点型特征
    - ✓ SIFT, GLOH, SURF, LIOP,
  - 二值型特征
    - ✓ BRIEF, ORB, FREAK, BRISK, CARD, Edge-SIFT

BRIEF: Binary Robust Independent Elementary Features (2010)

- Binary test

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

- BRIEF descriptor

$$f_{n_d}(p) = \sum_{1 \le i \le n_d} 2^{i-1} \tau(p; x_i, y_j)$$

- For each S*S patch
  - Smooth it
  - Pick pixels using pre-defined binary tests

Smoothing kernels

- De-noising with Gaussian kernels

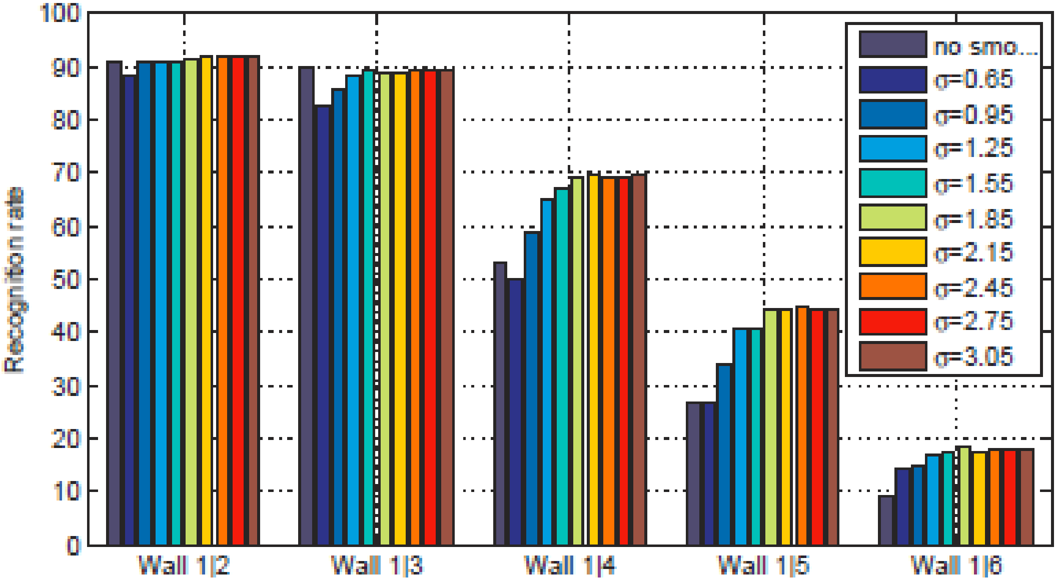


Fig. 1. Each group of 10 bars represents the recognition rates in one specific stereo pair for increasing levels of Gaussian smoothing. Especially for the hard-to-match pairs, which are those on the right side of the plot, smoothing is essential in slowing down the rate at which the recognition rate decreases.

Spatial arrangement of the binary tests

- (X,Y)~i.i.d. Uniform
- (X,Y)~i.i.d. Gaussian $(-\frac{S}{2},\frac{S}{2})$
- X~i.i.d. Gaussian $(0,\frac{1}{25}S^2)$ Y~i.i.d. Gaussian $(0,\frac{1}{25}S^2)$
- Randomly sampled from discrete locations of a coarse polar grid introducing a spatial quantization. $(x_i,\frac{1}{100}S^2)$
- $\forall i: x_i = (0,0)^T$ and y_i takes all possible values on a coarse polar grid containing n_d points

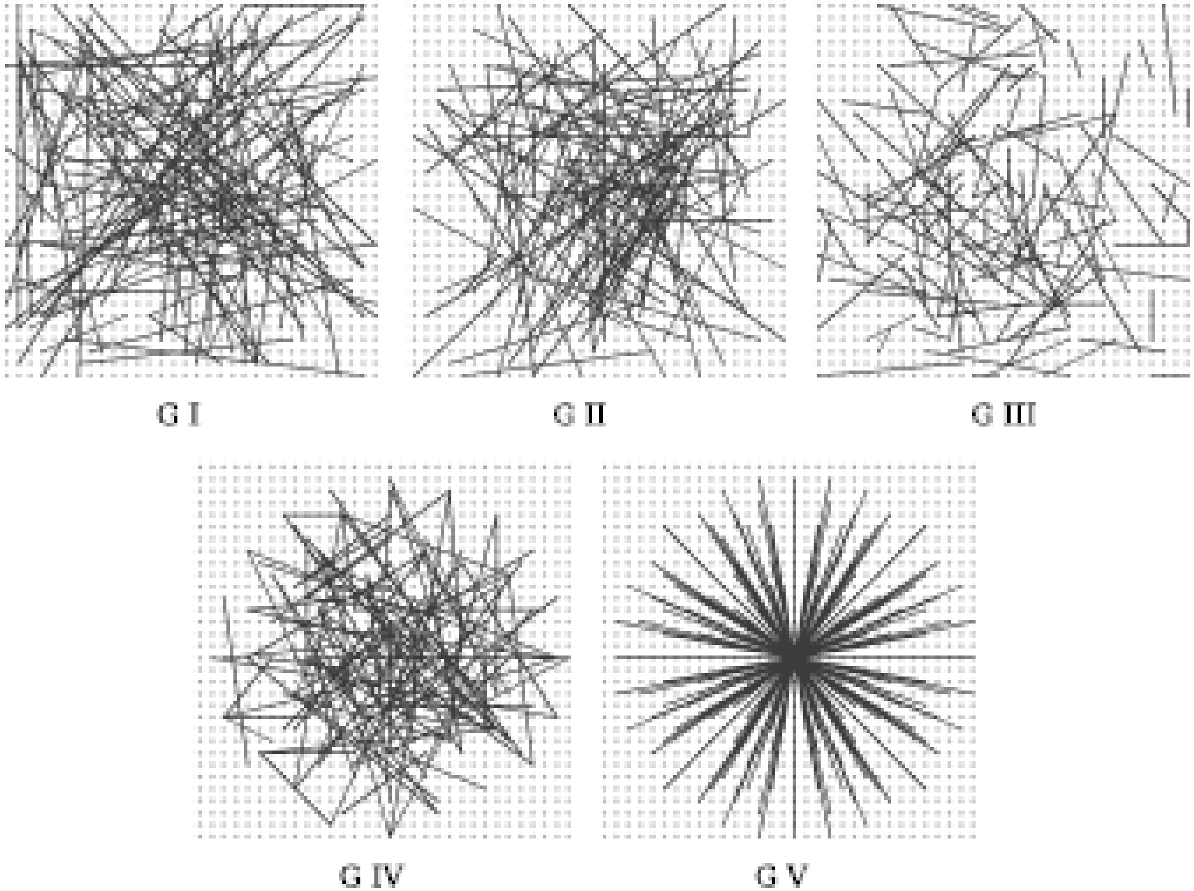


Fig. 2. Different approaches to choosing the test locations. All except the righmost one are selected by random sampling. Showing 128 tests in every image.

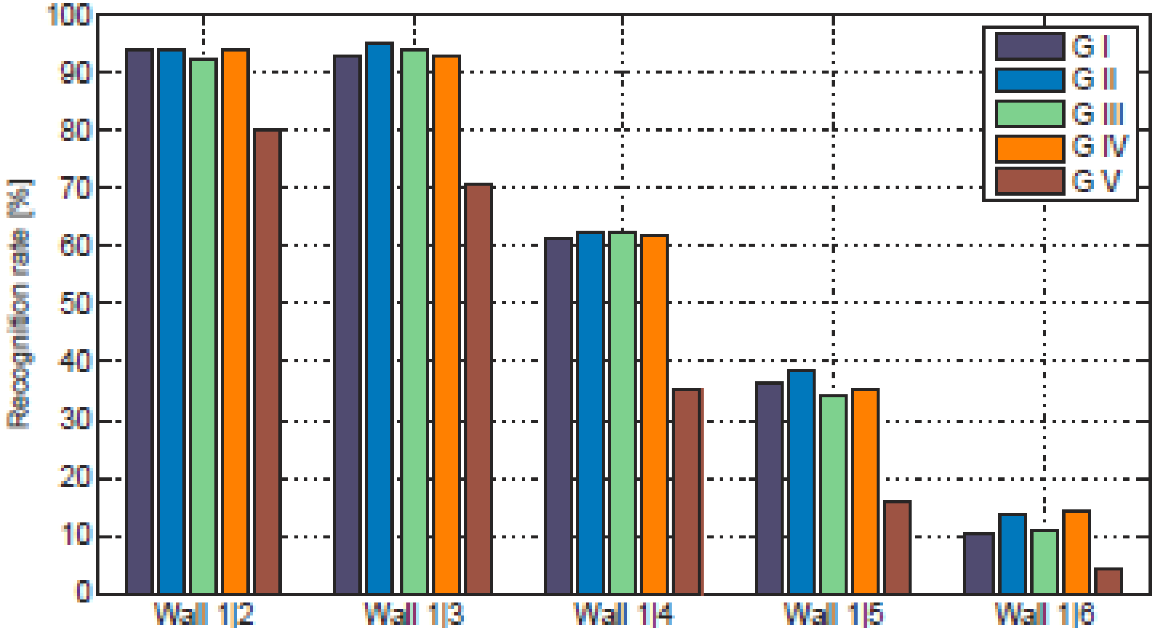
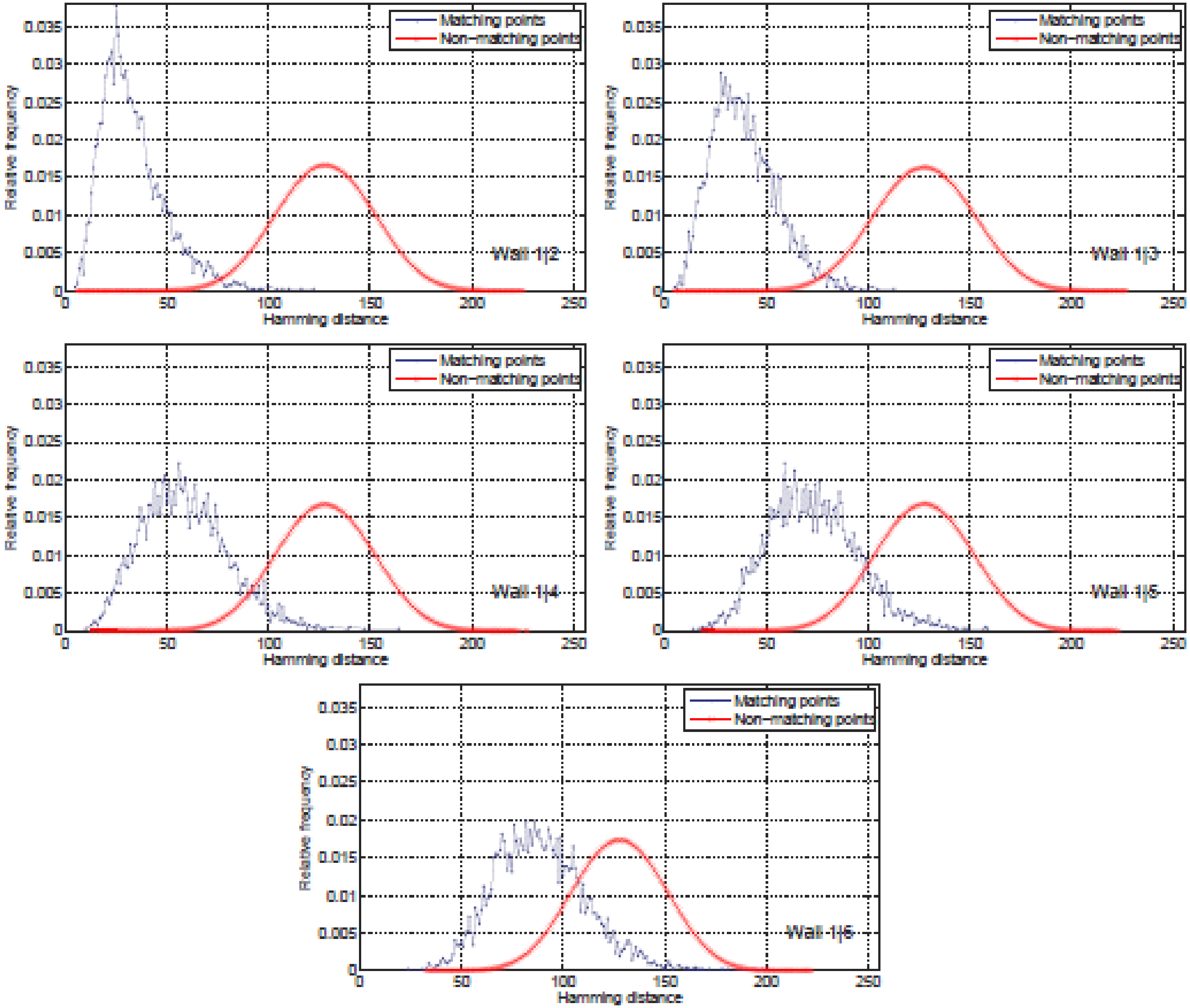
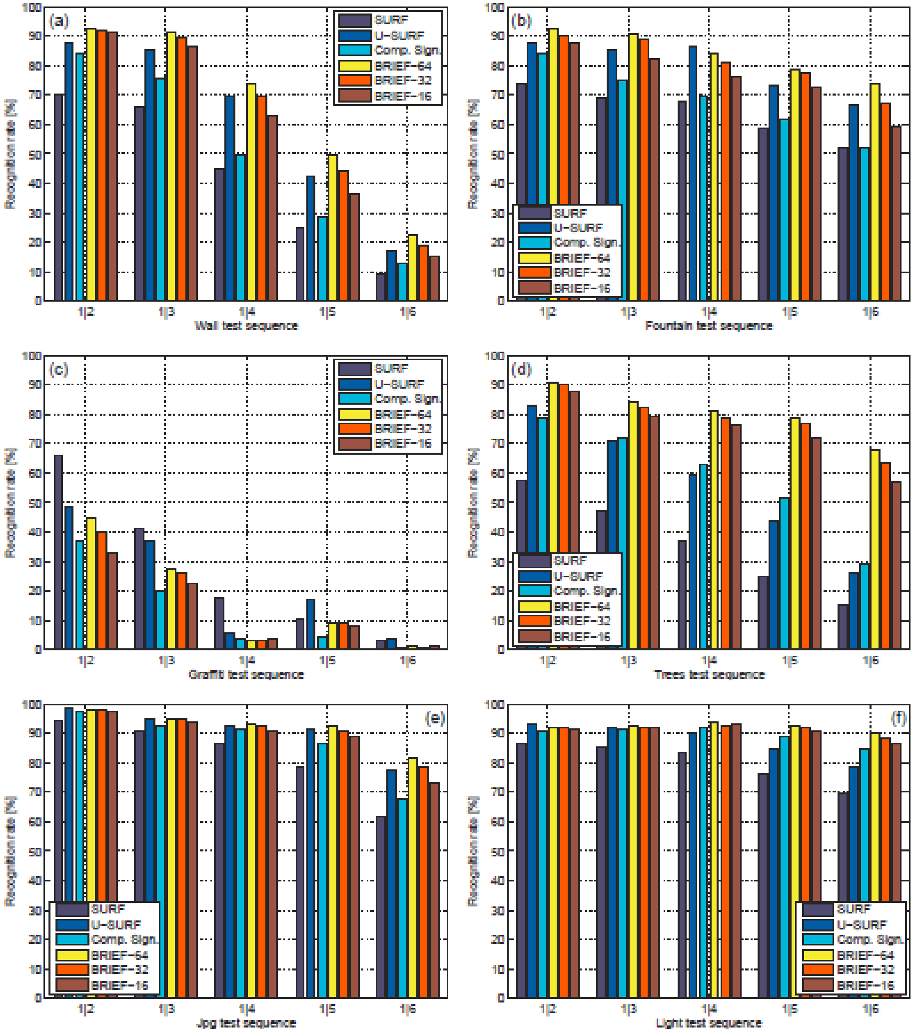


Fig. 3. Recognition rate for the five different test geometries introduced in section 3.2.

Distance Distributions



Experiments



Oriented FAST and Rotated BRIEF (ORB)

- 主方向：质心与几何中心的偏移
- 计算方法
  - 定义特征点（x， y）的邻域像素的矩

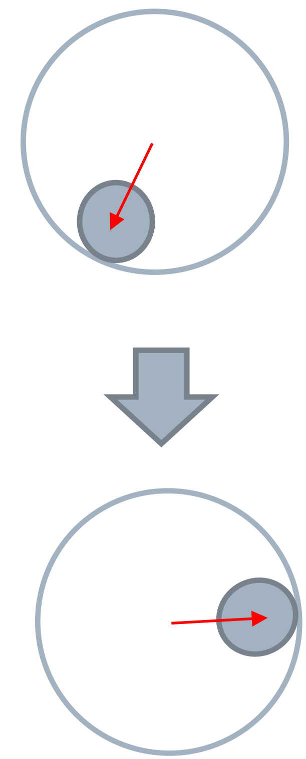
$$m_{pq} = \sum_{x,y} x^p y^q I(x,y)$$

  - 得到质心：

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

  - 特征点与质心的夹角定义为：

$$\theta = arctan(m_{01}, m_{10})$$



Binary Robust Invariant Scalable Keypoints (BRISK)

主方向计算方法

- 特征点局部梯度

$$\mathbf{g}(\mathbf{p}_i, \mathbf{p}_j) = (\mathbf{p}_j - \mathbf{p}_i) \cdot \frac{I(\mathbf{p}_j, \sigma_j) - I(\mathbf{p}_i, \sigma_i)}{\|\mathbf{p}_j - \mathbf{p}_i\|^2}.$$

- 定义短距离点对子集、长距离点对子集：

$$\mathcal{A} = \{(\mathbf{p}_i, \mathbf{p}_j) \in \mathbb{R}^2 \times \mathbb{R}^2 \mid i < N \wedge j < i \wedge i, j \in \mathbb{N}\}$$

$$\mathcal{S} = \{(\mathbf{p}_i, \mathbf{p}_j) \in \mathcal{A} \mid \|\mathbf{p}_j - \mathbf{p}_i\| < \delta_{max}\} \subseteq \mathcal{A}$$

$$\mathcal{L} = \{(\mathbf{p}_i, \mathbf{p}_j) \in \mathcal{A} \mid \|\mathbf{p}_j - \mathbf{p}_i\| > \delta_{min}\} \subseteq \mathcal{A}.$$

- 局部梯度均值

$$\mathbf{g} = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{L} \cdot \sum_{(\mathbf{p}_i, \mathbf{p}_j) \in \mathcal{L}} \mathbf{g}(\mathbf{p}_i, \mathbf{p}_j).$$

- 主方向：

$$\alpha = \arctan2(g_y, g_x)$$

主要内容

- 9.1 图像表达基本框架
- 9.2 局部视觉特征描述
  - 9.2.1 局部关键点检测
  - 9.2.2 局部区域描述
- 9.3 特征编码与聚合(coding & aggregation)

图像表达方法：Fisher Vector Representation

- General idea
  - Represent a set of features by a single vector: the gradient of the log-likelihood function
- Generative model

$$\nabla_{\lambda} \log p(X \mid \lambda)$$

X=$\{x_t\}, t = 1, \cdots, K$
$p(\cdot)$: the PDF function,
$\boldsymbol{\lambda}$: a set of parameters

- Independence assumption of features

$$L(X|\lambda) = \log p(X|\lambda) = \log \prod_{t=1}^{T} p(x_t|\lambda) = \sum_{t=1}^{T} \log p(x_t|\lambda)$$

F. Perronnin and C. Dance, "Fisher Kernels on Visual Vocabularies for Image Categorization", CVPR, 2007

Fisher Vector Representation

- Approximate distribution by Gaussian mixture model (GMM)

$$p(x_t \mid \lambda) = \sum_{i=1}^{N} w_i p_i(x_t \mid \lambda) \qquad \sum_{i=1}^{N} w_i = 1$$

$$p_i(x \mid \lambda) = \frac{\exp\{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\}}{(2\pi)^{D/2}\left|\Sigma_i\right|^{1/2}} \qquad \lambda = \{w_i, \mu_i, \Sigma_i, i = 1, \cdots, N\}$$

- The gradient of the log-likelihood function

$$\nabla_{\alpha_i} \log p(X \mid \lambda) = \sum_{t=1}^{T}\left[\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1}\right]$$

$$w_k = \frac{\exp(\alpha_k)}{\sum_{j=1}^{N}\exp(\alpha_j)}$$

$$\gamma_t(i) = p(i \mid x_t, \lambda) = \frac{w_i p_i(x_t \mid \lambda)}{\sum_{j=1}^{N} w_j p_j(x_t \mid \lambda)}$$

$$\nabla_{\mu_i^d} \log p(X \mid \lambda) = \sum_{t=1}^{T}\gamma_t(i)\left[\frac{x_t^d - \mu_t^d}{(\sigma_i^d)^2}\right]$$

$$\nabla_{\sigma_i^d} \log p(X \mid \lambda) = \sum_{t=1}^{T}\gamma_t(i)\left[\frac{(x_t^d - \mu_i^d)^2}{(\sigma_i^d)^2} - \frac{1}{\sigma_i^d}\right]$$

$$w_i = \frac{\exp(\alpha_k)}{\sum_{j=1}^{N}\exp(\alpha_j)}$$

Relationship to BoW and VLAD

- BOW (bag-of-words model)
  - The gradient with respect to the weight of GMM:

$$\nabla_{\alpha_i} \log p(X \mid \lambda) = \sum_{t=1}^{T} \left[\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1} \right]$$

  - Soft version of BoW:

$$b_i \propto \sum_{t=1}^{T} \gamma_t(i) \text{ for the } i\text{-th visual word}$$

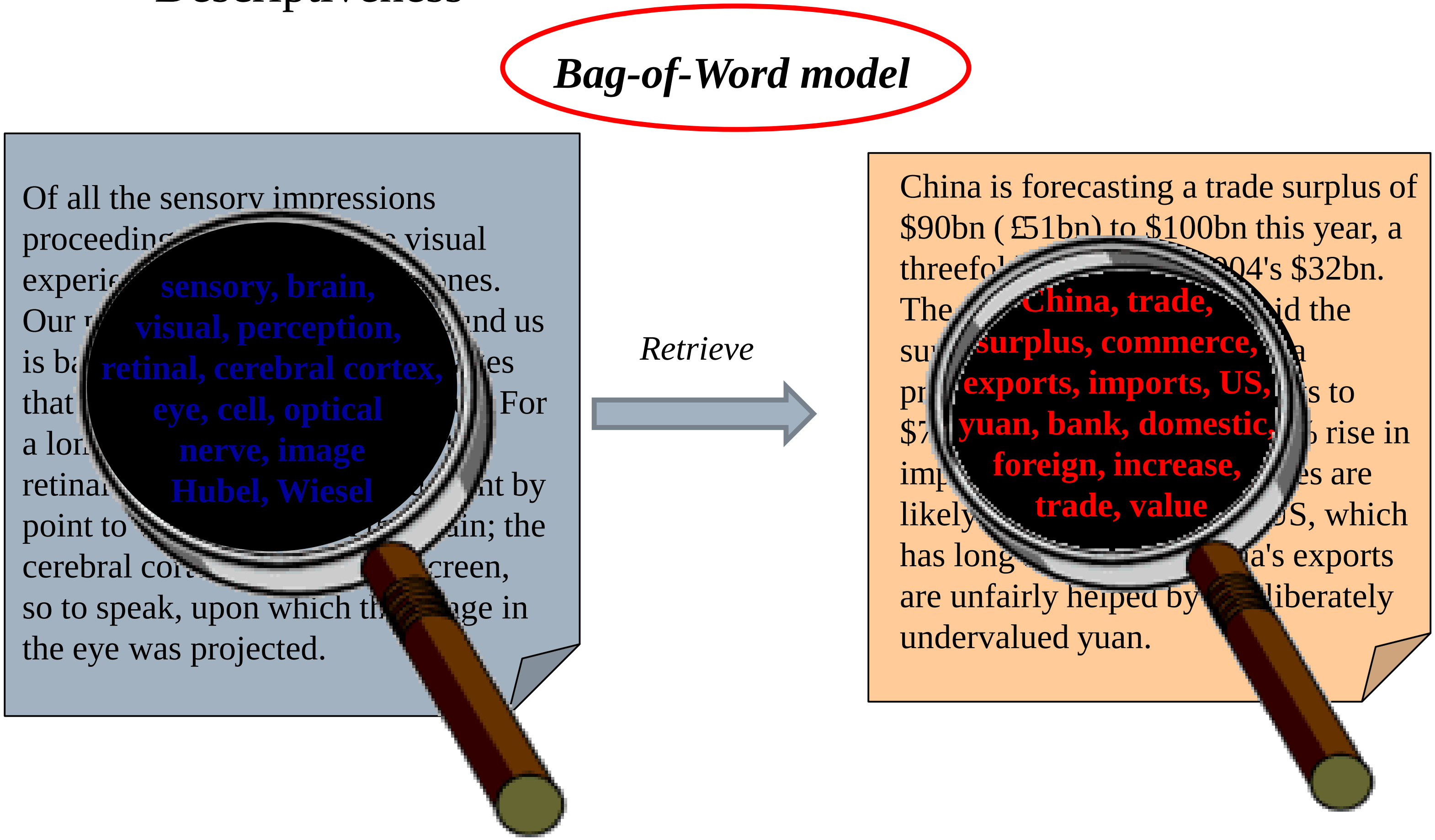
  - Sparseness is ensured with large N
    - ✓ Suitable for the inverted index
- VLAD (vector of locally aggregated descriptor)
  - The gradient with respect to the mean vector of GMM

$$\nabla_{\mu_i^d} \log p(X \mid \lambda) = \sum_{t=1}^{T} \gamma_t(i) \left[\frac{x_t^d - \mu_t^d}{(\sigma_i^d)^2} \right]$$

  - Disadvantage
    - ✓ Non-sparse: unsuitable to apply the inverted index

Bag-of-Words (BoW) Model

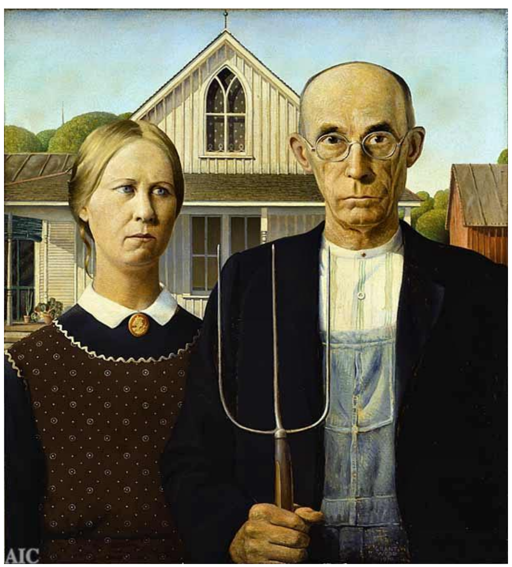
- Text Words in Information Retrieval (IR)
  - Compactness
  - Descriptiveness



Bag of Visual Words

- Could images be represented as Bag-of-Visual Words?

Image → **Bag of 'visual words'**



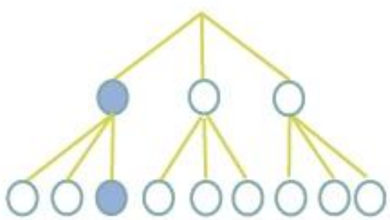
?

Bag of Visual Words

- Three questions
  - **Q1:** 如何定义视觉单词和视觉码本？
    - ✓ **A1:** 对大量的局部视觉特征进行聚类（如k-means），聚类中心视为视觉单词，所有的聚类中心构成了视觉码本。
  - **Q2:** 如何将每个局部视觉特征与视觉单词对应？
    - ✓ **A2：** 基于视觉码本，通过**矢量量化**，将局部视觉特征量化到最近的视觉单词。
    - ✓ 有损压缩（lossy compression），表达紧凑。
  - **Q3:** 如何基于对一幅图像的局部视觉特征集合进行紧凑表达?
    - ✓ **A2：** 基于视觉单词在图像中出现的频率，构造视觉单词直方图

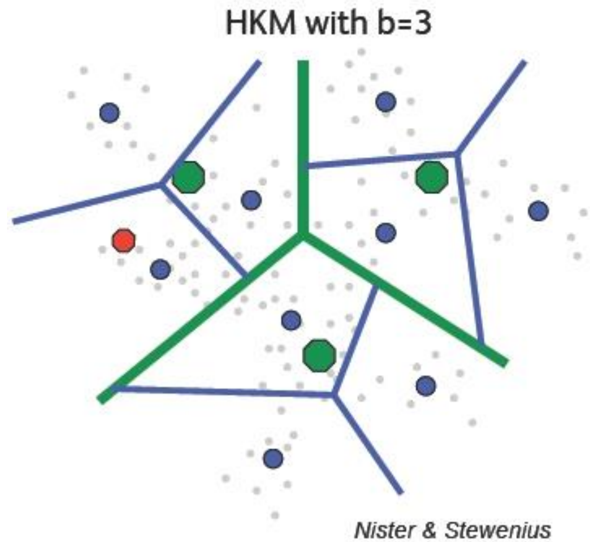
Large Vocabularies with Learned Quantizer

- Feature matching by quantization: $ANN(\mathbf{x}, \mathbf{q}) = \{\mathbf{y} | \mathbf{q}(\mathbf{x}) = \mathbf{q}(\mathbf{y})\}$.
- Hierarchical k-means [Nister 06]
  - K-means tree of height h



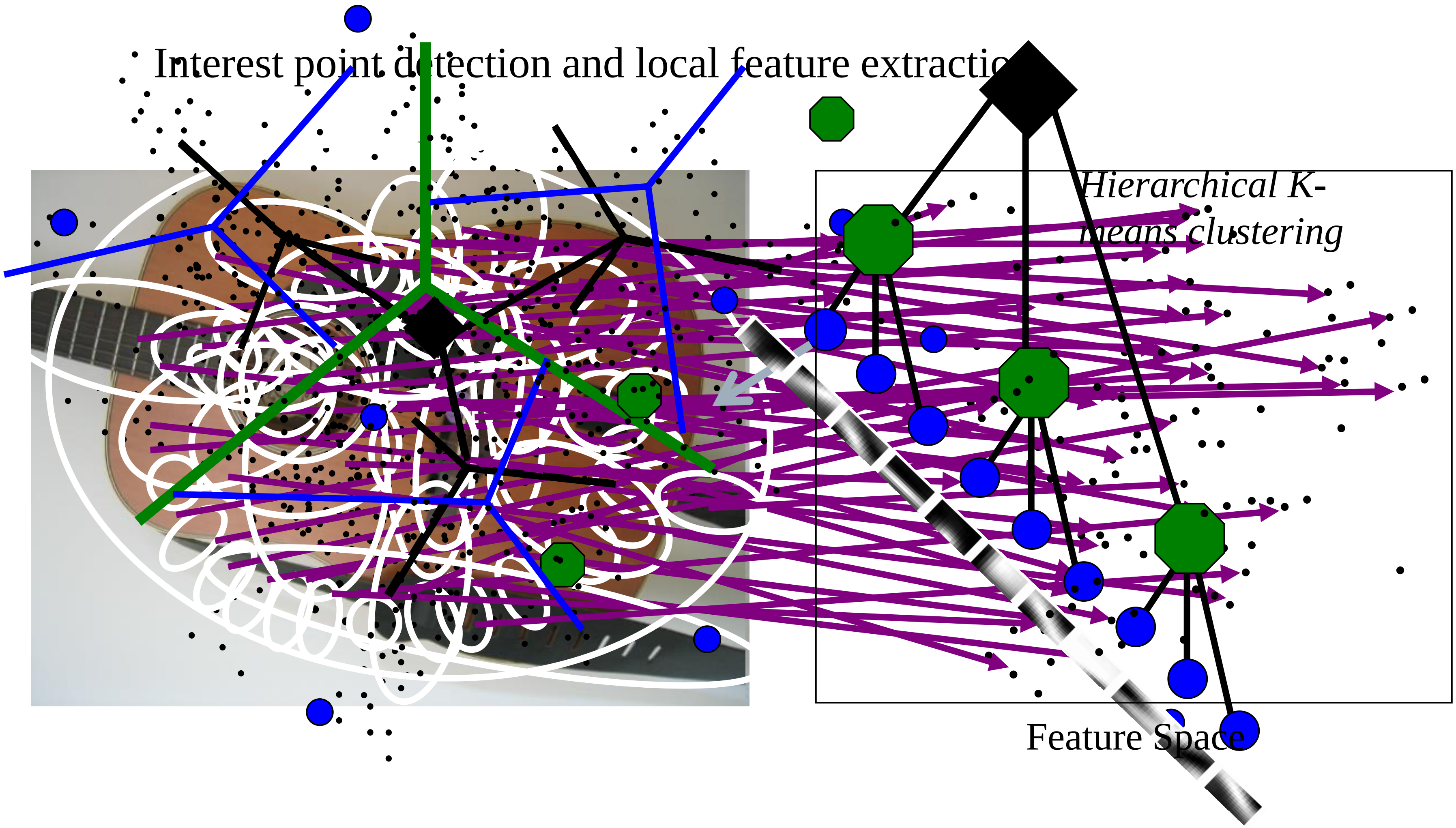
  - Branching factor b: $k = b^h$
  - Assignment Complexity:

$$\mathcal{O}(dhb) = \mathcal{O}(dhk^{\frac{1}{h}})$$



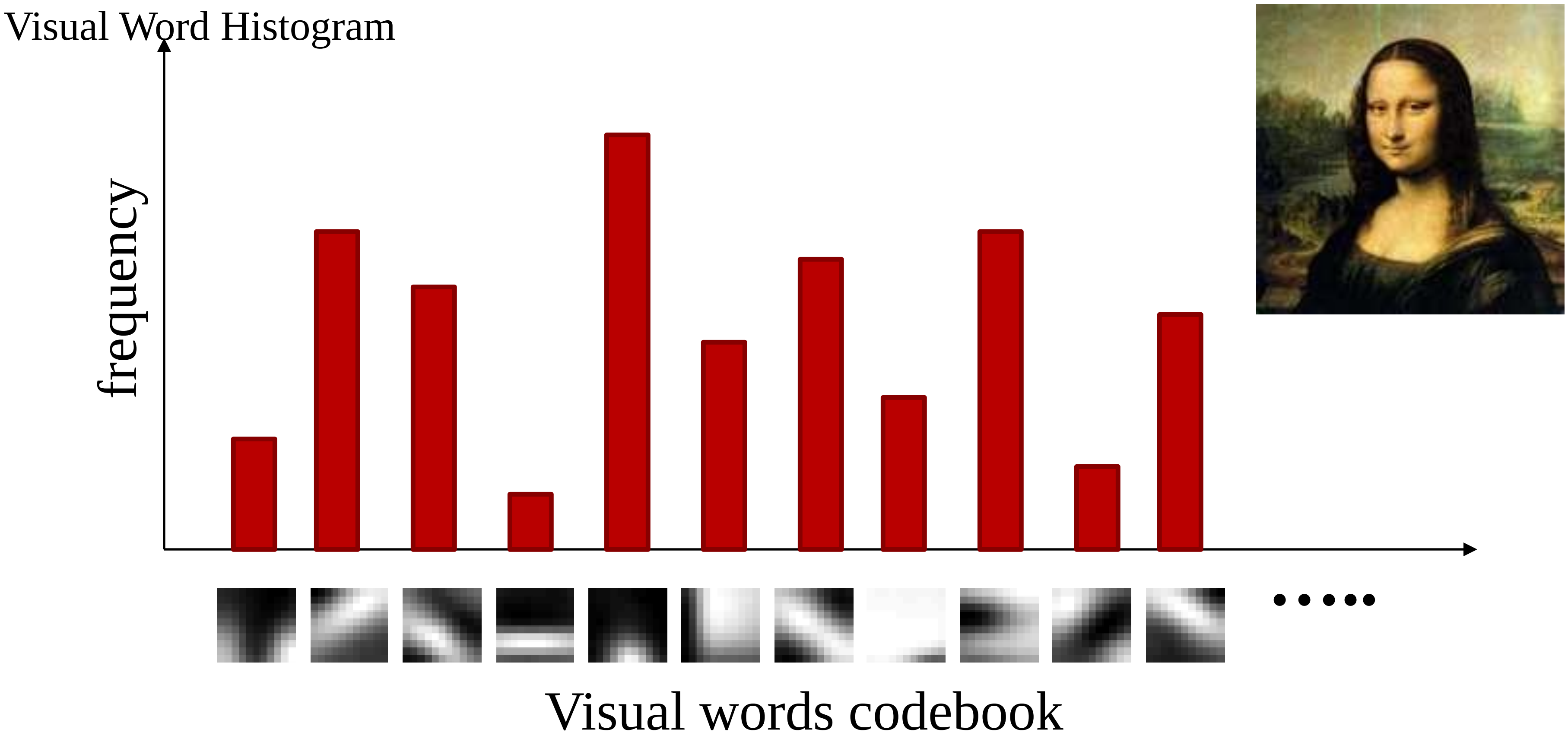
- Approximate k-means [Philbin 07]
  - Based on approximate nearest neighbor search
  - With parallel k-tree

Bag of Visual Words with Vocabulary Tree



Bag of Visual Words

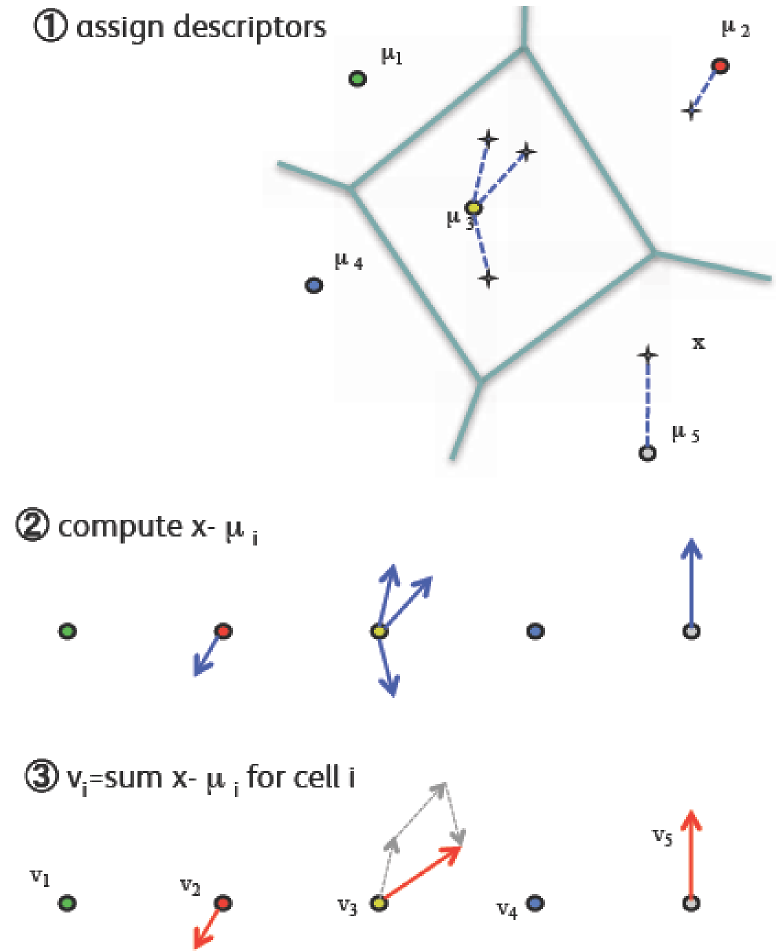
- Summarize entire image based on its distribution (histogram) of word occurrences.



VLAD (Vector of Locally Aggregated Descriptor)

Given a codebook $\{\mu_i, i = 1 \dots N\}$,
e.g. learned with K-means, and a set of
local descriptors $X = \{x_t, t = 1 \dots T\}$:

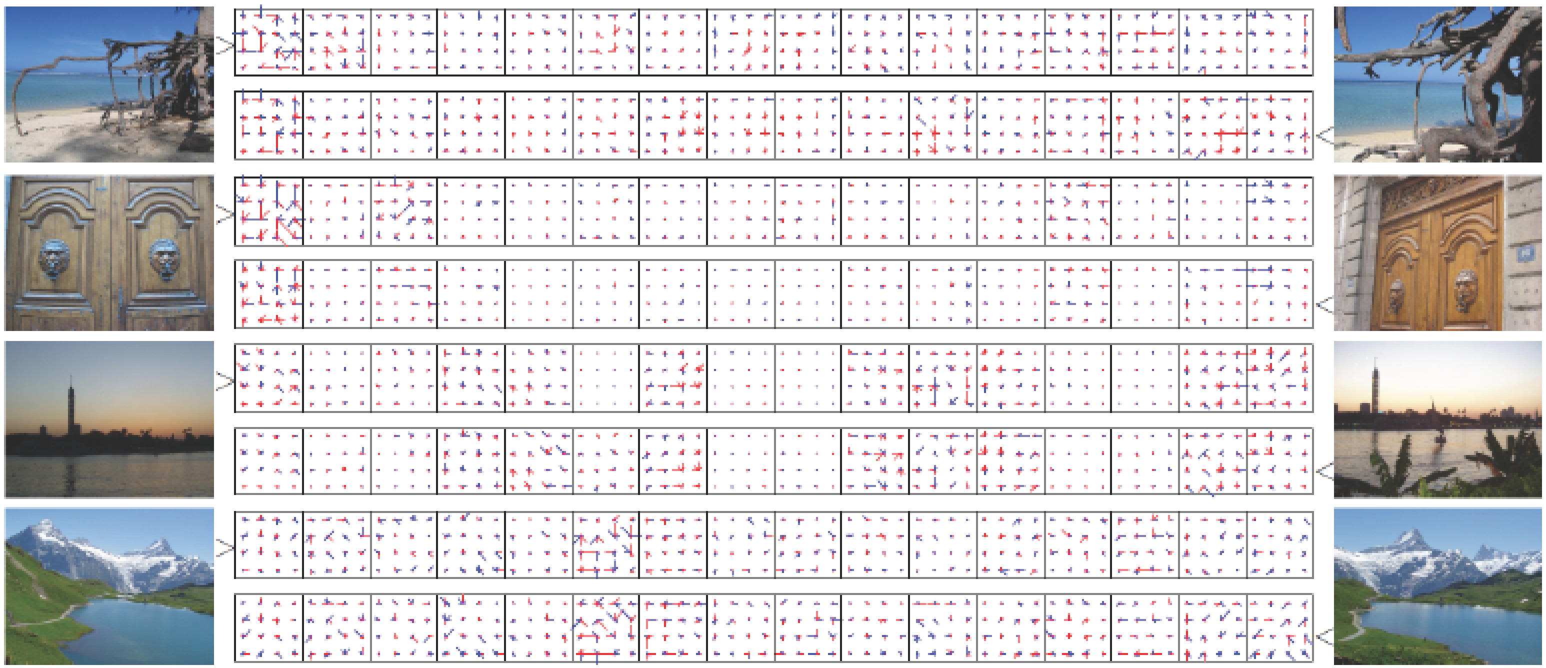
- ① assign: $\mathrm{NN}(x_t) = \arg\min_{\mu_i} ||x_t - \mu_i||$
- ②③ compute: $v_i = \sum_{x_t : \mathrm{NN}(x_t) = \mu_i} x_t - \mu_i$
- concatenate v_i's + ℓ_2 normalize



Jégou, Douze, Schmid and Pérez, "Aggregating local descriptors into a compact image representation", CVPR'10.

An example of VLAD

A graphical representation of $v_i = \sum_{x_t : \mathrm{NN}(x_t) = \mu_i} x_t - \mu_i$



Jégou, Douze, Schmid and Pérez, "Aggregating local descriptors into a compact image representation", CVPR'10.

乘积量化 (Product Quantization)

- Basic idea
  - Partition feature vector into *m* sub-vectors
  - Quantize sub-vectors with *m* distinct quantizers

$$\underbrace{x_1,\ldots,x_{D^*}}_{u_1(x)},\ldots,\underbrace{x_{D-D^*+1},\ldots,x_D}_{u_m(x)} \rightarrow q_1(u_1(x)),\ldots,q_m(u_m(x)),$$

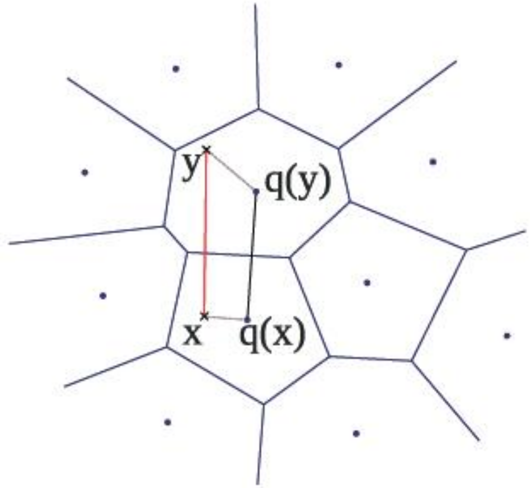
- Advantage
  - Low complexity in quantization
  - Extremely fine quantization of the feature space
- Vector distance approximation

$$\|\mathbf{x}-\mathbf{y}\|_2^2=\sum_{i=1}^{D}(x_i-y_i)^2 \approx \sum_{k=1}^{m}\|u_k(\mathbf{x})-u_k(\mathbf{y})\|_2^2=\sum_{k=1}^{m}\|\mathbf{c}(a_k)-\mathbf{c}(b_k)\|_2^2=\sum_{k=1}^{m}d_{a_k b_k}$$

where $a_k=q_k(u_k(\mathbf{x})),\ \ b_k=q_k(u_k(\mathbf{y}))$

Distance table

d_{11}	d_{12}	…	d_{1K}
$\vdots$			
d_{K1}			d_K K



$$d(x, q(y)) - d(y, q(y)) \leq d(x, y) \leq d(x, q(y)) + d(y, q(y)),$$

$$(d(x, y) - d(x, q(y)))^2 \leq d(y, q(y))^2.$$

$$\begin{aligned}\mathrm{MSDE}(q) &\triangleq \iint \left(d(x, y) - \tilde{d}(x, y)\right)^2 p(x)dx\, p(y)dy. \\ &\leq \int p(x)\left(\int d\big(y, q(y)\big)^2 p(y)\, dy\right)dx \\ &\leq \mathrm{MSE}(q),\end{aligned}$$