CS664 Computer Vision

6. Features

Dan Huttenlocher

Cornell University
Faculty of Computing and Information Science

Local Invariant Features

- Similarity- and affine-invariant keypoint detection
  - Sparse using non-maximum suppression
  - Stable under lighting and viewpoint changes
    - Recall 2D affine transform corresponds to 3D motion of plane under weak perspective
- Similarity- and affine-invariant, or at least stable, descriptors for keypoints
  - Enable accurate matching of keypoints between images
    - Object recognition, image registration

Local Feature Detectors

- Harris corner detector covered earlier
  - And related detectors such as KLT, estimates of image derivatives aggregated over local regions
- Based on magnitudes of eigenvalues of Hessian matrix (partial second derivatives)
  - Aggregated over window (weighted)

$$M = \sum \sum \; w(x,y) \begin{bmatrix} I_x^2 & I_{xy} \\ I_{xy} & I_y^2 \end{bmatrix}$$

- More recent detectors build on this to gain varying degrees of invariance

Geometric Invariance

- Local detectors sensitive to geometric transformations
- Several investigations of scale invariance
  - Multi-scale Harris corners
  - Harris-Laplacian
  - SIFT (Scale Invariant Feature Transform)
- Some investigations of similarity and affine invariance
  - Tradeoff of degree of invariance and amount of information in descriptors

Scale Invariant Detection

- Kernels for determining scale

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

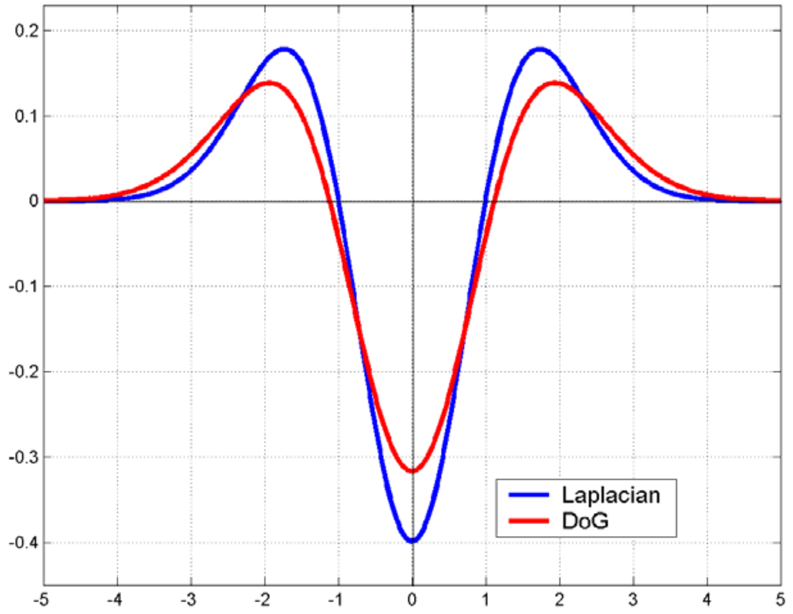
(Scale-normalized Laplacian)

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

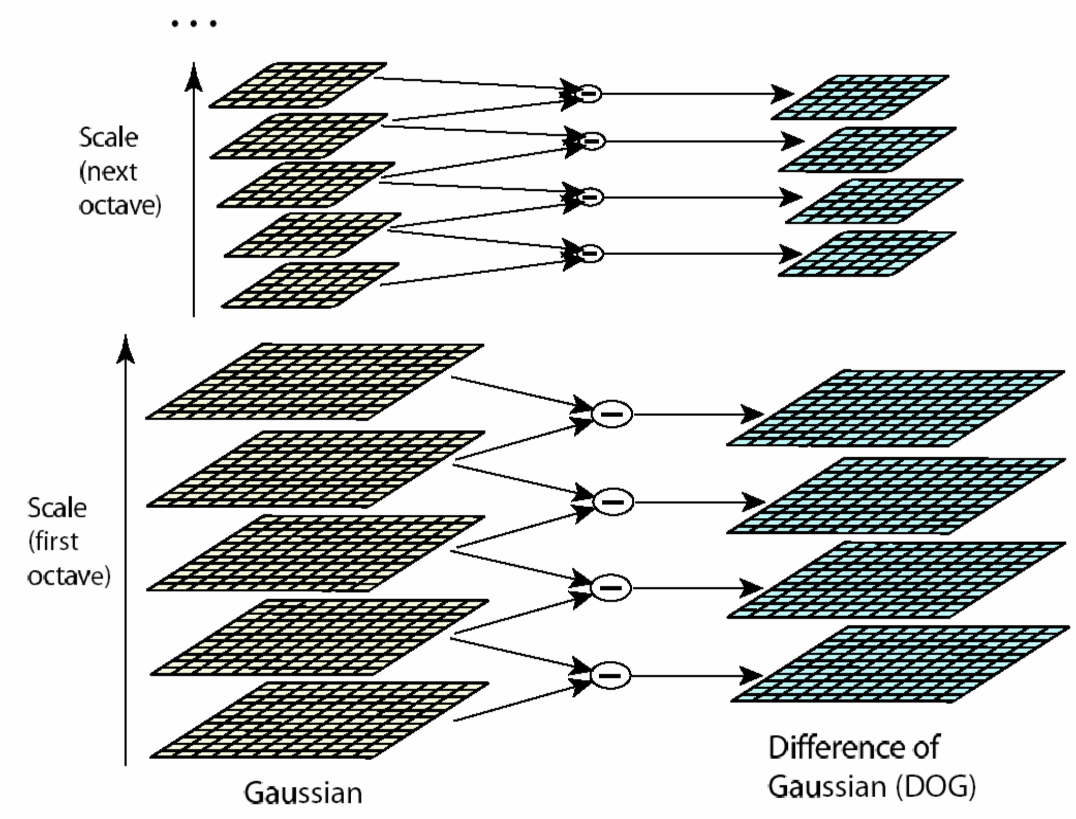
(Difference of Gaussians)

where Gaussian

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



Scale-Space Octaves

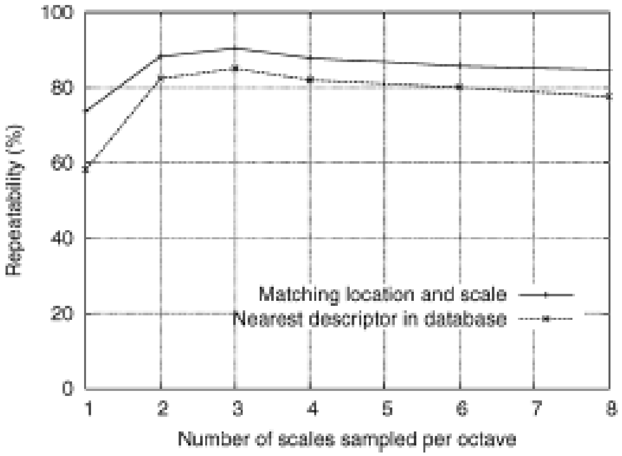


- Each octave is doubling of scale
  - Halve image dimensions
- Within octave several scales
  - In SIFT paper Lowe uses 3
  - Same dimension for all images within octave

Lowe advocated DoG as efficient

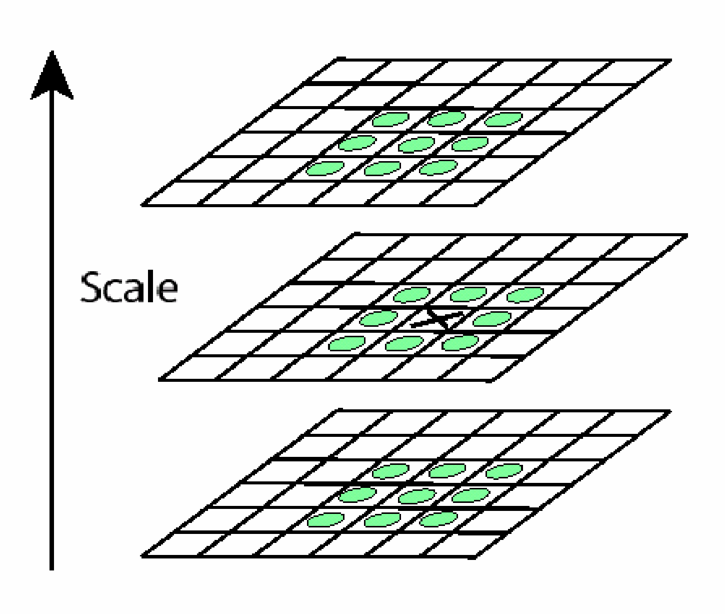
Determining Scales per Octave

- More scales per octave seems like should perform better, but get slower
  - More densely covering the scale space
- Lowe found that more than 3 scales per octave did not help
  - Measured by repeatability of features as image distorted
  - Also by ability to retrieve features using descriptors (yet to be defined)



Scale Space Extrema

- Non-maximum suppression both spatially and in scale
- Compare magnitude at given cell to all 26 neighbors
  - More positive or more negative
- On average not many comparisons per cell
- As with edges, NMS alone not enough



Localizing SIFT Keypoints

- Better results by interpolating than by taking center of cell as location
  - Fit quadratic to surrounding points
- Taylor expansion around point
  - Where D is difference of Gaussian

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{T}}\frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$$

- Offset of extremum as location
  - Use finite differences

$$\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1}\frac{\partial D}{\partial \mathbf{x}}$$

Selecting Good SIFT Keypoints

- Low contrast extrema discarded
  - Analogous to magnitude constraint in edge and corner detection
- Edge-like extrema also discarded
  - Using similar analysis to Harris corner detector
  - Eigenvalues α, β of Hessian proportional to principal curvature
  - Use trace and determinant to avoid computing square roots
  - Threshold (Lowe uses r=10)

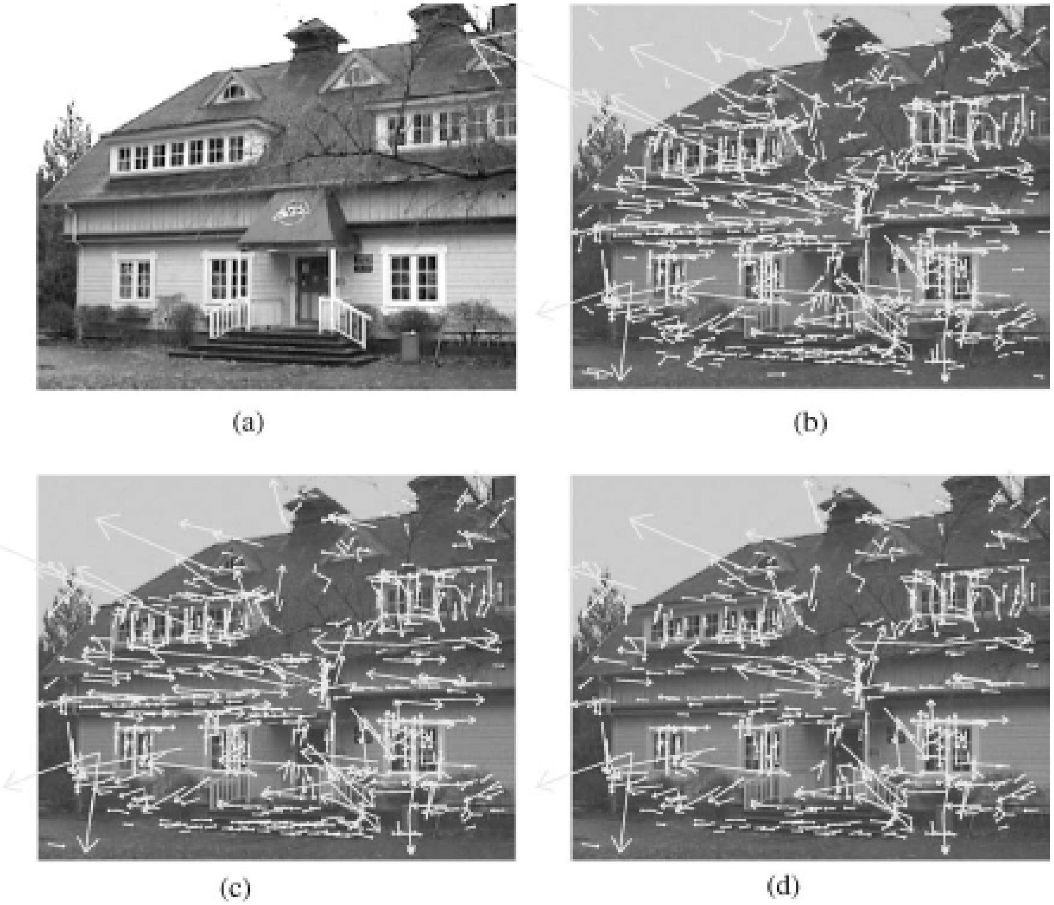
$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

$$\begin{aligned} \mathrm{Tr}(\mathbf{H}) &= D_{xx} + D_{yy} = \alpha + \beta, \\ \mathrm{Det}(\mathbf{H}) &= D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta. \end{aligned}$$

$$\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} < \frac{(r+1)^2}{r}$$

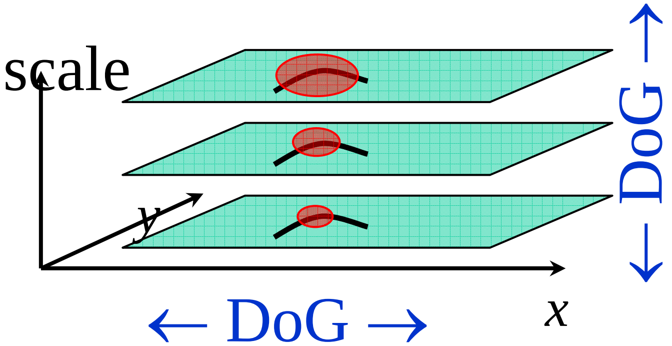
SIFT Feature Example

- Initial features (832)
  - Scale indicated by length of vector
- Low contrast removed (729)
- Low curvature removed (536)

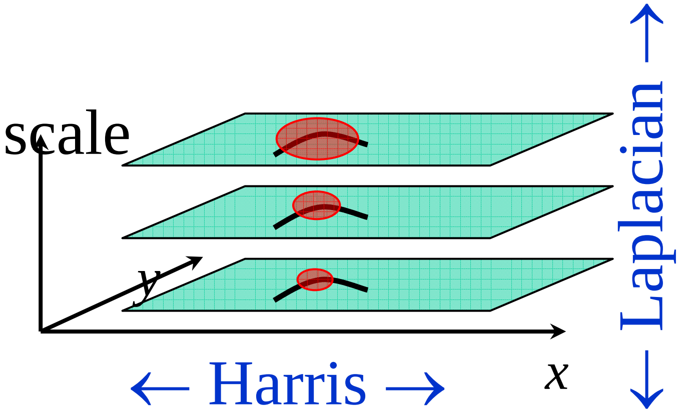


Scale Invariant Detectors

- SIFT (Lowe)
  Find local max of
  - Difference of Gaussians in space and scale



- Harris-Laplacian
  Find local max of
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale

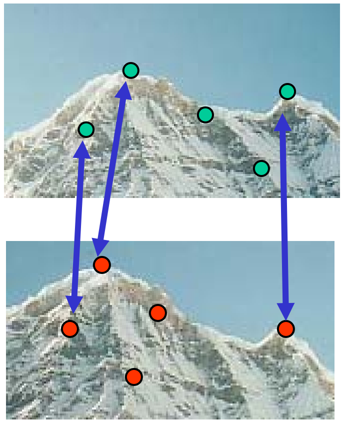


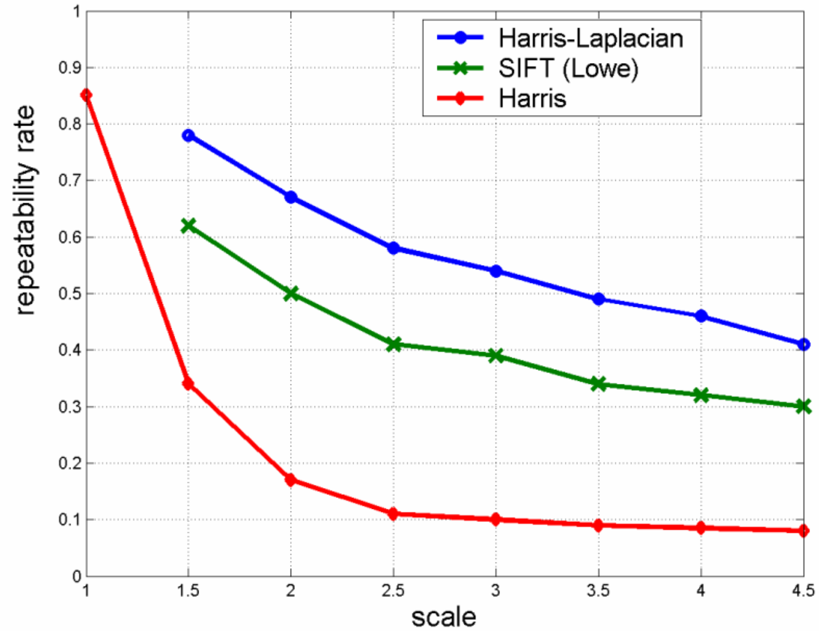
Scale Invariant Detectors

- Experimental evaluation of detectors w.r.t. scale change

Repeatability rate:

$$\frac{\text{\# correspondences}}{\text{\# possible correspondences}}$$





K.Mikolajczyk, C.Schmid. “Indexing Based on Scale Invariant Interest Points”. ICCV 2001

Scale Invariant Detection: Summary

- Given: two images of the same scene with a *scale difference* between them
- Goal: find *the same* interest points *independently* in each image
- Solution: search for *maxima* of suitable functions in *scale* and in *space* (over the image)

Methods:

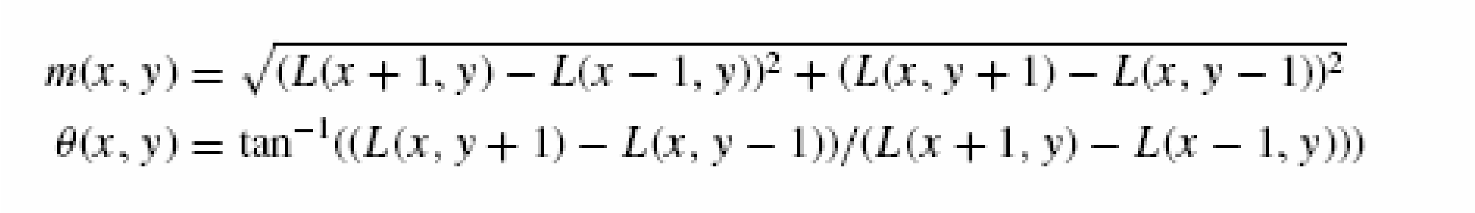
1. Harris-Laplacian [Mikolajczyk, Schmid]: maximize Laplacian over scale, Harris' measure of corner response over the image
2. SIFT [Lowe]: maximize Difference of Gaussians over scale and space

SIFT Orientation Invariance

- Determine orientation explicitly and normalize to canonical orientation
- Other alternative is detector that is itself invariant to orientation
  - But processing of image for such a detector removes more information
  - Recall discussion of image transformations
- Location and scale invariant detectors
  - In practice affine invariant because use extrema
  - Rotation or linear “insensitive” descriptors

SIFT Orientation Assignment

- Measure orientation and magnitude in closest scale image



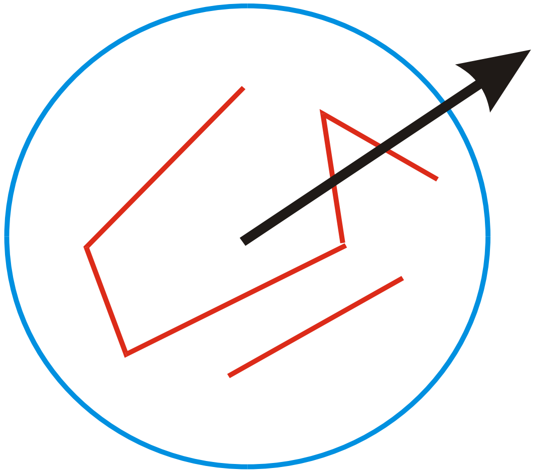
$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

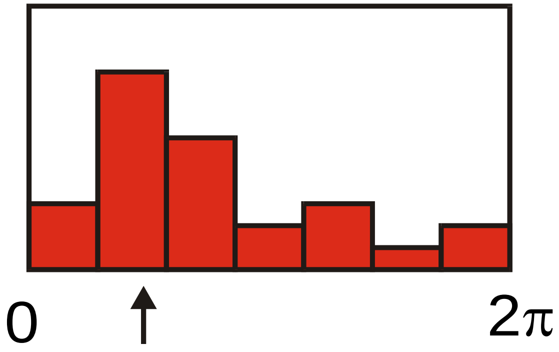
$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1))/(L(x+1, y) - L(x-1, y)))$$

- Form orientation histogram from region around keypoint
  - Using Gaussian weighting with sigma 1.5x scale
  - And weighting based on gradient magnitude
- Note such operations fast using box filtering and pre-computation

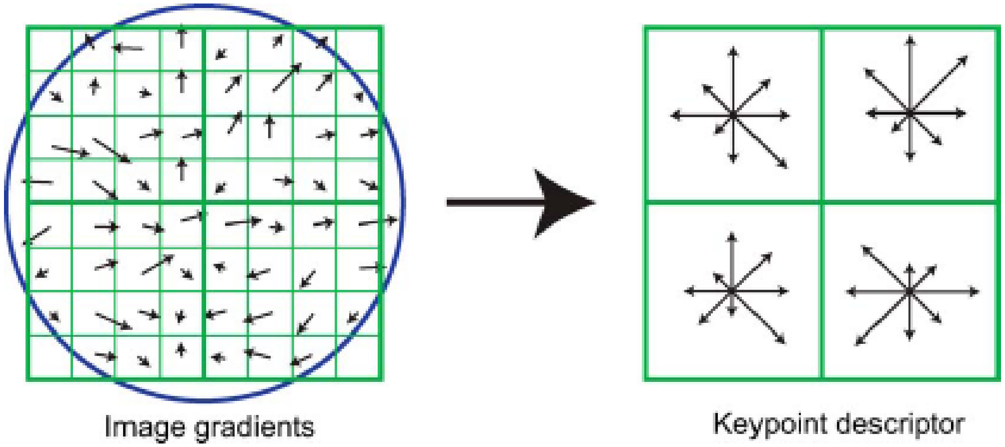
SIFT Orientation Assignment

- Orientation determined by largest peak
- Assign second orientation if second peak at least .8 height
  - About 15% of keypoints have two orientations
- Parabolic fit to three values around peak to interpolate orientation





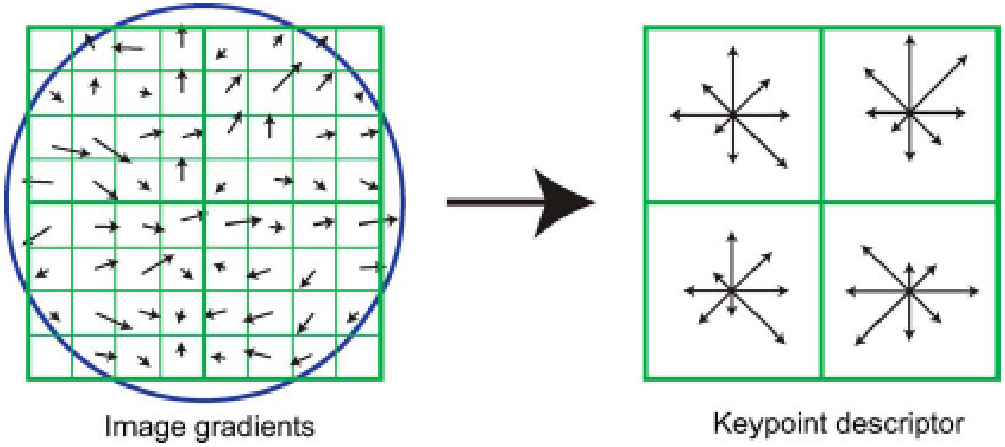
SIFT Descriptor



- Histogram gradient information over small local areas
  - Provide some measure of invariance to small changes in position
- Weighted both by magnitude and using Gaussian from center

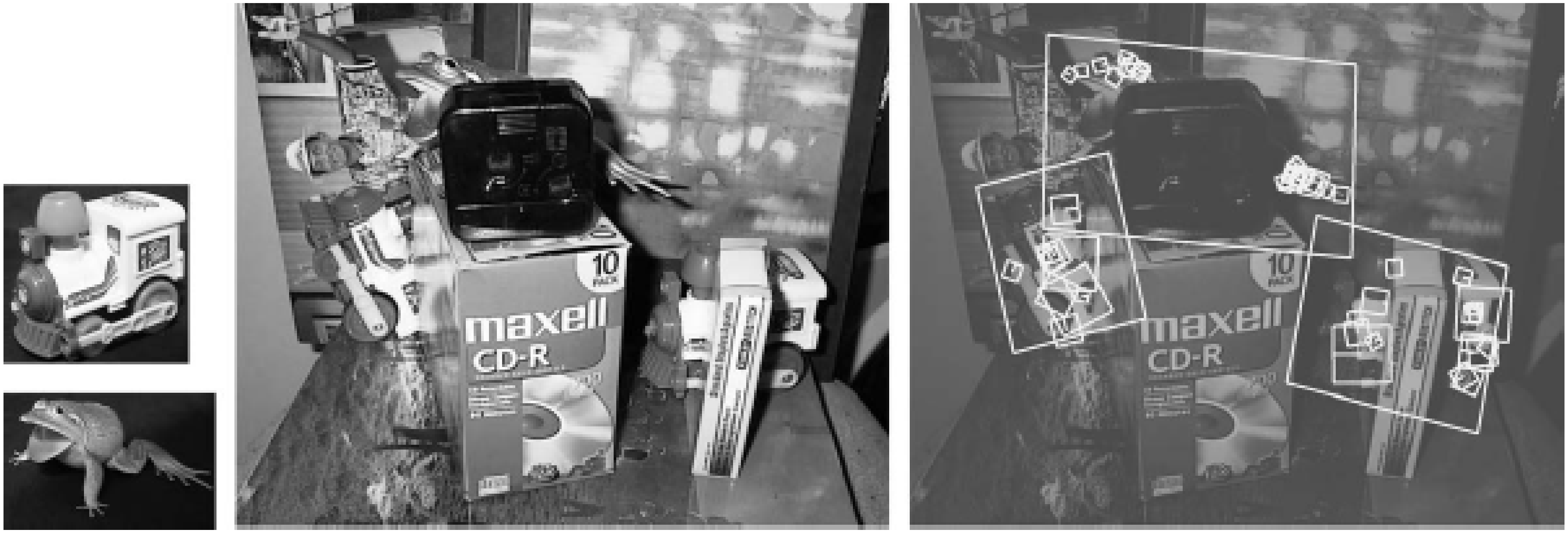
SIFT Descriptor

- 8 orientations and 4x4 array computed by histogramming over 16x16 image region
  - 128 dimensional feature vector
- Illustration shows 2x2 array and 8x8 region



SIFT Matching Example

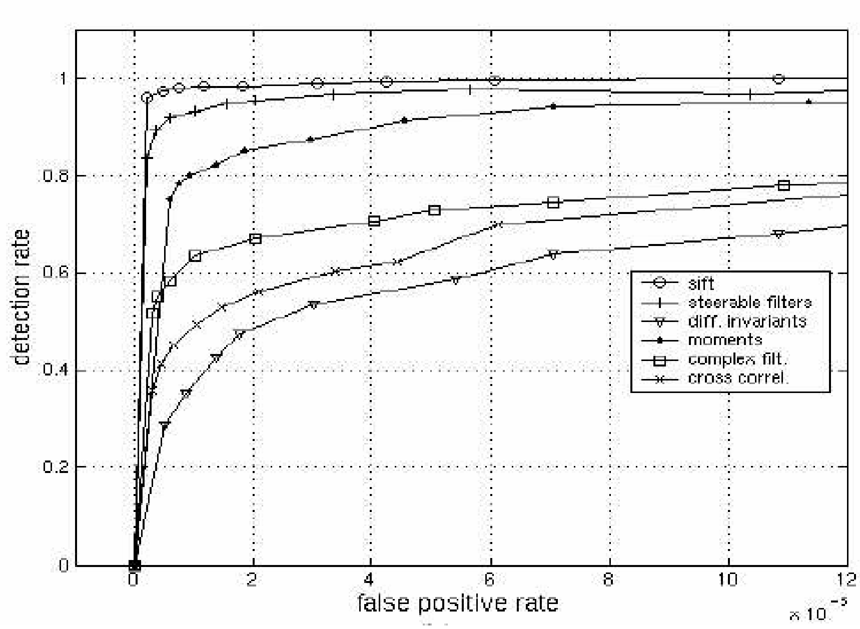
- Example objects in cluttered environment
  - Rectangle around detected objects based on model images at left



SIFT Matching Results

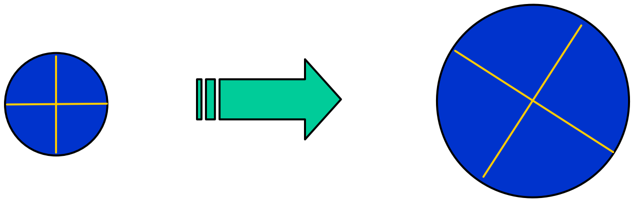
- Empirically found to have good performance under range of transformations
  - Rotation, scale, intensity change and small affine transformations

Scale = 2.5
Rotation = 45^0

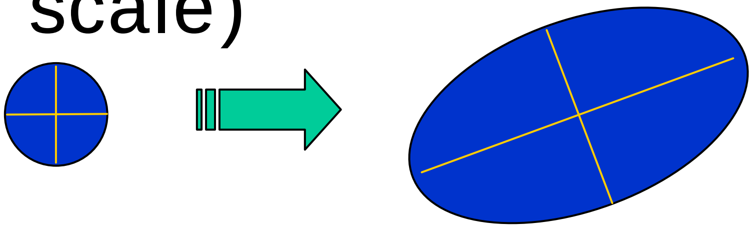


Affine invariant detection

- Above considered:
  Similarity transform (rotation + uniform scale)

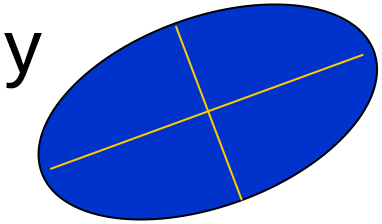


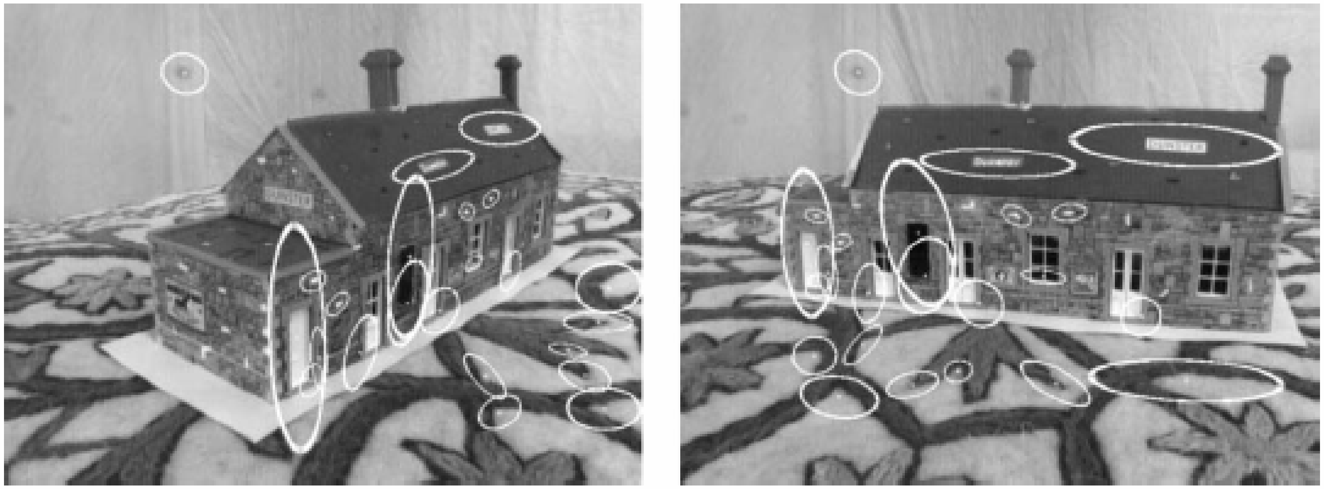
- Now go on to:
  Affine transform (rotation + non-uniform scale)



Affine invariant detection

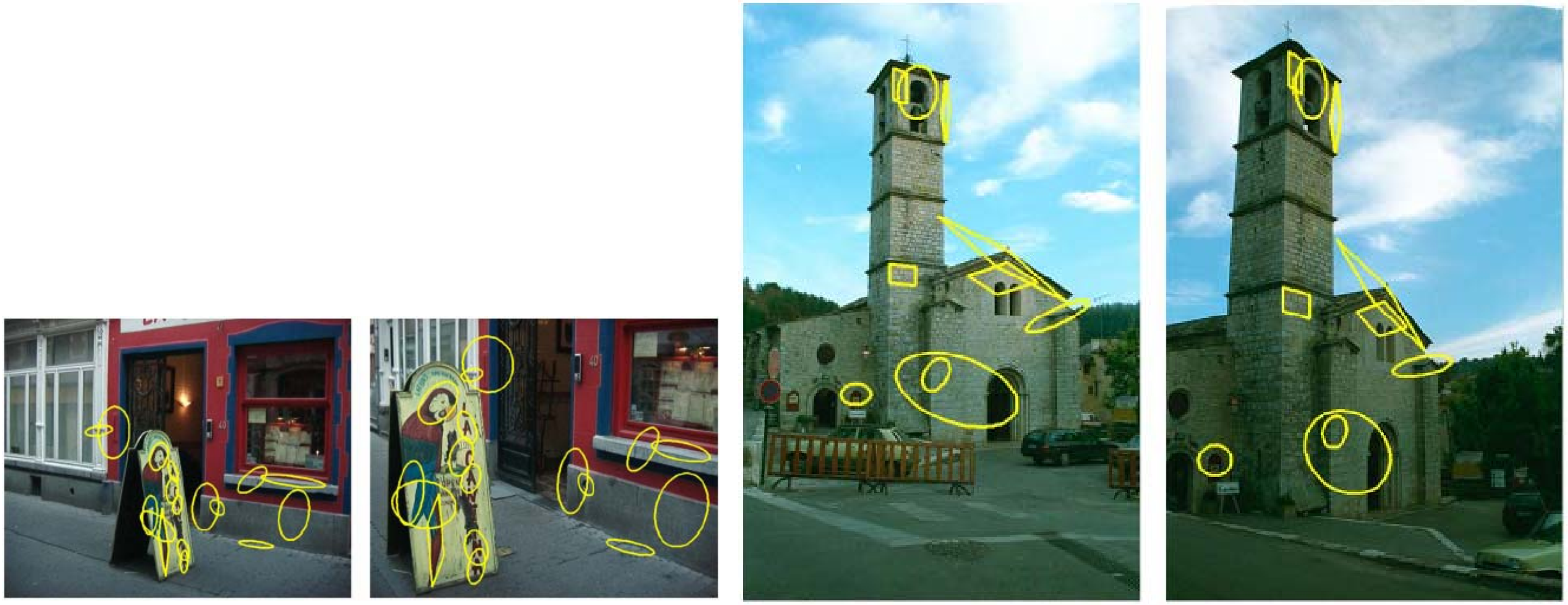
- Harris-Affine [Mikolajczyk & Schmid, IJCV04]:
- Use Harris *moment* matrix to select dominant directions and anisotropy





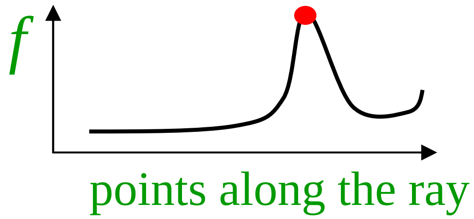
Affine invariant detection

- Matching Widely Separated Views Based on Affine Invariant Regions, T. TUYTELAARS and L. VAN GOOL, IJCV 2004



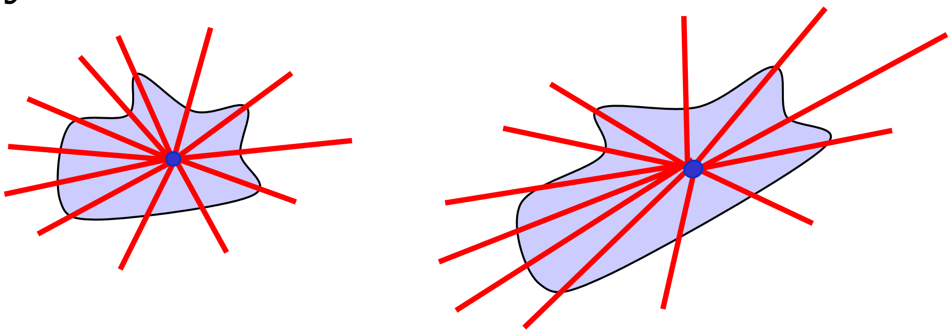
Affine invariant detection

- Take a local intensity extremum as initial point
- Go along every ray starting from this point and stop when extremum of function *f* is reached



$$f(t) = \frac{|I(t) - I_0|}{\frac{1}{t}\int_o^t |I(t) - I_0|\,dt}$$

- We will obtain approximately corresponding regions



Remark: search for scale in every direction

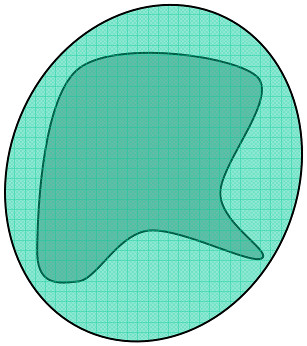
Affine invariant detection

- The regions found may not exactly correspond, so we approximate them with ellipses
- Geometric Moments:

$$m_{pq} = \int_{\mathbb{R}^2} x^p y^q f(x,y)\,dxdy$$

Fact: moments m_{pq} uniquely determine the function f

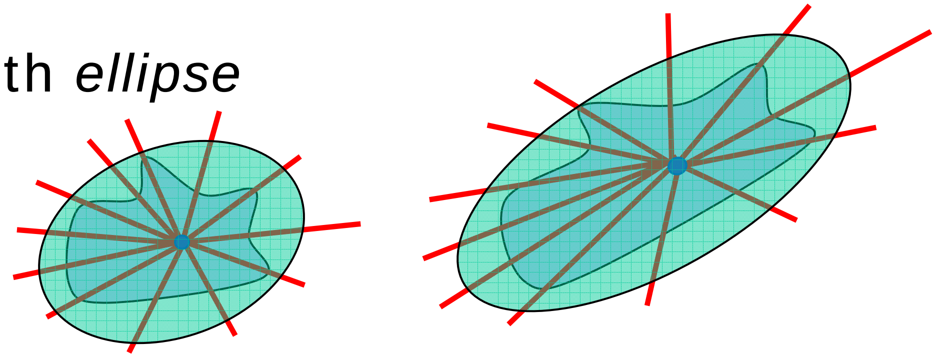
Taking f to be the characteristic function of a region (1 inside, 0 outside), moments of orders up to 2 allow to approximate the region by an ellipse



This ellipse will have the same moments of orders up to 2 as the original region

Affine invariant detection

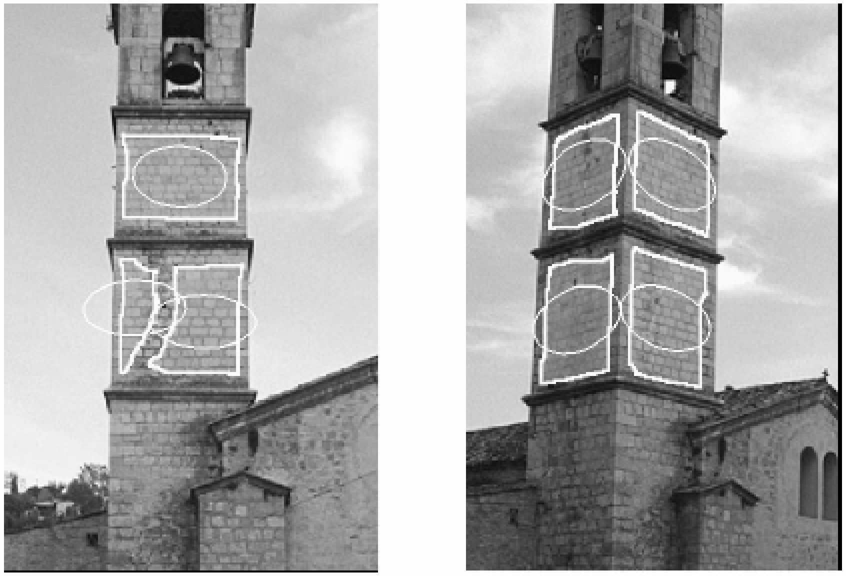
- Algorithm summary (detection of affine invariant region):
  - Start from a *local intensity extremum* point
  - Go in *every direction* until the point of extremum of some function *f*
  - Curve connecting the points is the region boundary
  - Compute *geometric moments* of orders up to 2 for this region
  - Replace the region with *ellipse*



Affine Invariant Texture Descriptor

- Segment the image into regions of different textures (by a non-invariant method)
- Compute matrix *M* (same as in Harris detector) over these regions

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$



- This matrix defines the ellipse

$$[x, y] M \begin{bmatrix} x \\ y \end{bmatrix} = 1$$

- Regions described by these ellipses are invariant under affine transformations
- Find affine normalized frame
- Compute rotation invariant descriptor

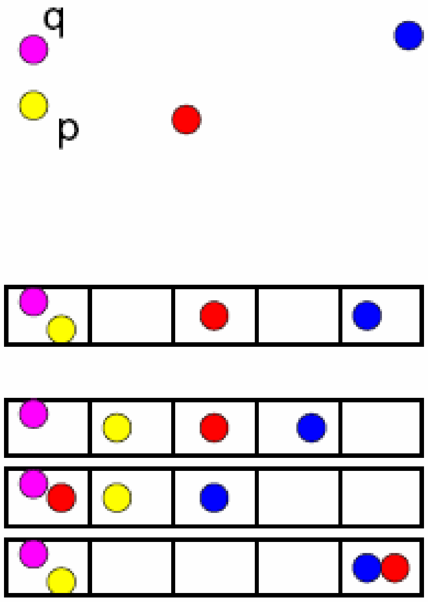
Feature matching

- Exhaustive search
  - for each feature in one image, look at *all* the other features in the other image(s)
- Hashing
  - compute a short descriptor from each feature vector, or hash longer descriptors (randomly)
- Nearest neighbor techniques
  - *k*-trees and their variants (Best Bin First)

Locality sensitive hashing

[Indyk-Motwani'98]

- Idea: construct hash functions $g: R^d \rightarrow U$ such that for any points p,q:
  - If $D(p,q) \leq r$, then $Pr[g(p)=g(q)]$ is ~~“high”~~ “not-so-small”
  - If $D(p,q) > cr$, then $Pr[g(p)=g(q)]$ is “small”
- Then we can solve the problem by hashing



Nearest neighbor techniques

- *k*-D tree and
- Best Bin First (BBF)

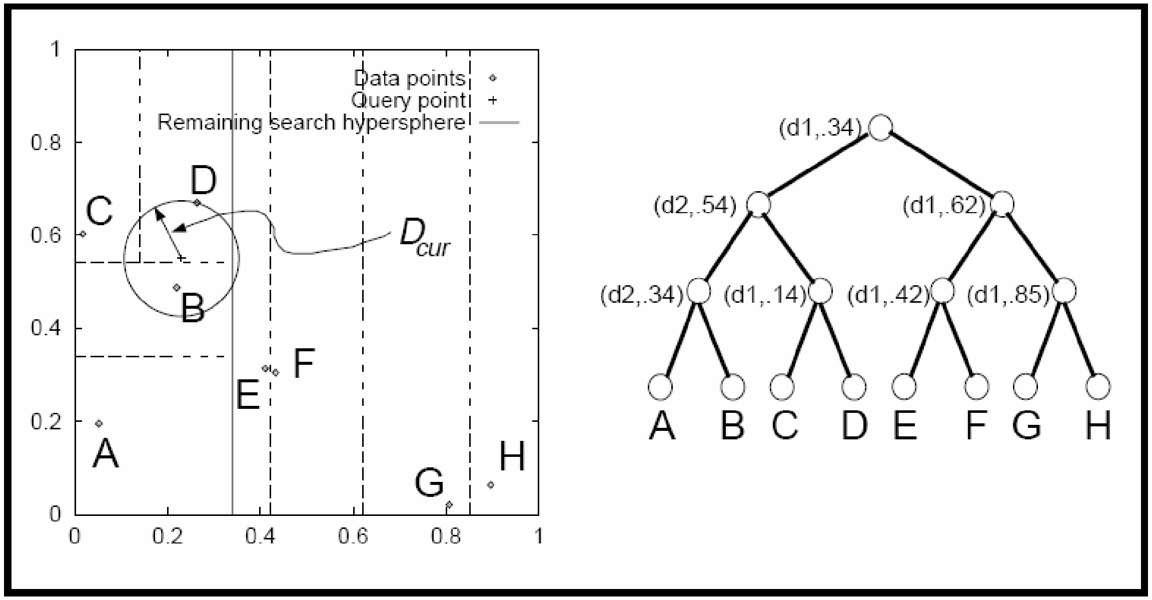


Figure 6: kd-tree with 8 data points labelled A-H, dimension of space k=2. On the right is the full tree, the leaf nodes containing the data points. Internal node information consists of the dimension of the cut plane and the value of the cut in that dimension. On the left is the 2D feature space carved into various sizes and shapes of bin, according to the distribution of the data points. The two representations are isomorphic. The situation shown on the left is after initial tree traversal to locate the bin for query point "+" (contains point D). In standard search, the closest nodes in the tree are examined first (starting at C). In BBF search, the closest bins to query point q are examined first (starting at B). The latter is more likely to maximize the overlap of (i) the hypersphere centered on q with radius D_{cur}, and (ii) the hyperrectangle of the bin to be searched. In this case, BBF search reduces the number of leaves to examine, since once point B is discovered, all other branches can be pruned.

Indexing Without Invariants in 3D Object Recognition, Beis and Lowe, PAMI'99