Smarter features for panoramas and beyond: SIFT

Fredo Durand
MIT EECS 6.8370/6.8371

Class evaluation survey

Smarter Features

Most slides from:
Matching with Invariant Features

Darya Frolova, Denis Simakov
The Weizmann Institute of Science

http://www.wisdom.weizmann.ac.il/~deniss/vision_spring04/files/InvariantFeatures.ppt

Matching with Features

- Problem 1:
  - Detect the *same* point *independently* in both images

counter-example:



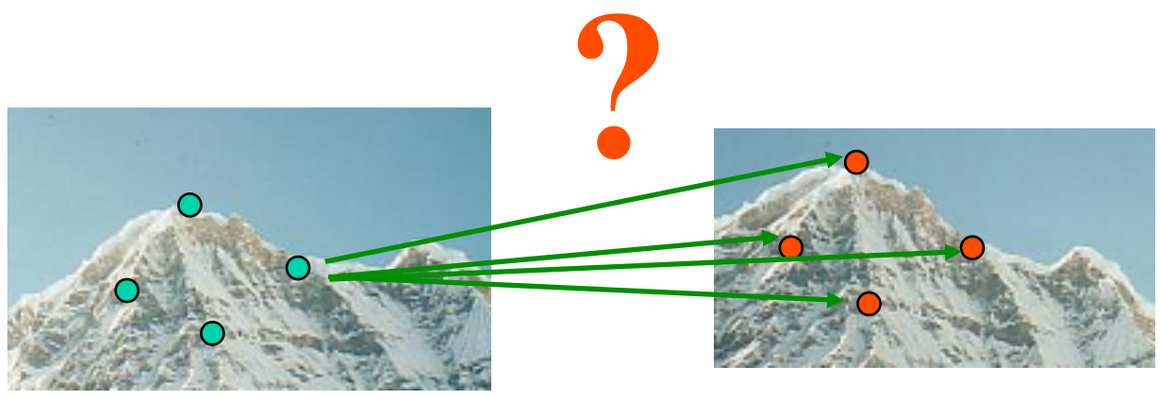


no chance to match!

We need a repeatable **detector**

Matching with Features

- Problem 2:
  - For each point correctly recognize the corresponding one



We need a reliable and distinctive **descriptor**

These feature point detectors and descriptors are perhaps the most important innovation in computer vision and graphics.

- Feature points are used also for:
  - Image alignment (homography, fundamental matrix)
  - 3D reconstruction
  - Motion tracking
  - Object recognition
  - Indexing and database retrieval
  - Robot navigation
  - … other

Possible changes between two images from same viewpoint

- Geometry
  - rotation along optical axis (2D rotation)
  - perspective distortion (wide angle)
  - scale (different zoom, or local scale due to perspective distortion)
  - radial distortion (lens aberration)
- Photometry (pixel values)
  - lighting changes, clouds moves
  - exposure not the same
  - vignetting

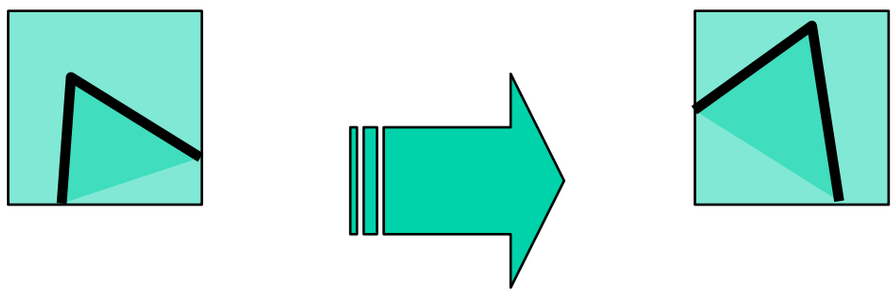
We want to:

detect *the same* interest points

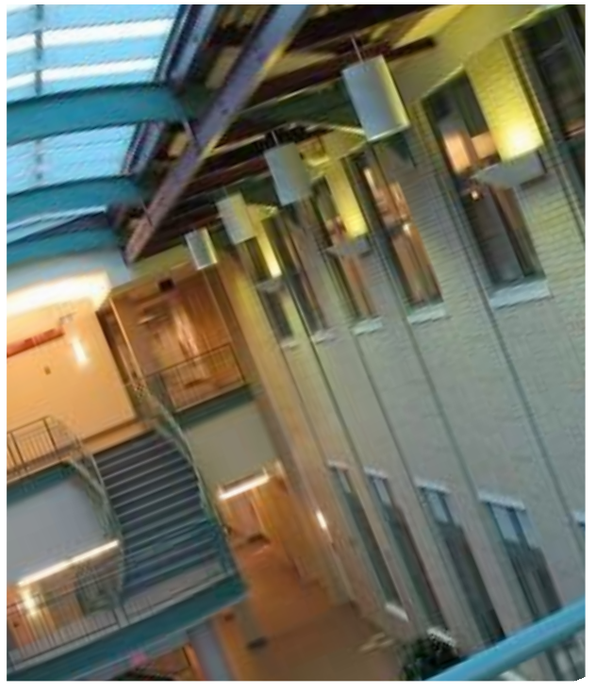
regardless of *image changes*

Harris Detector: Some Properties

- Rotation invariance?

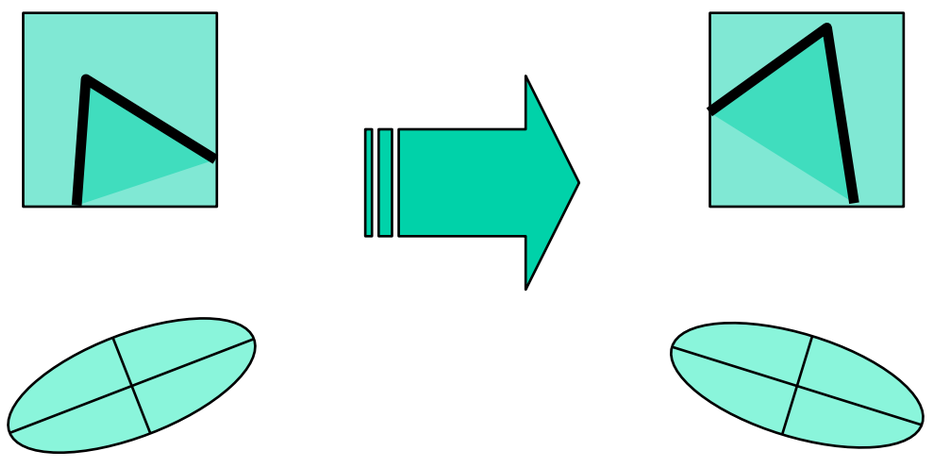






Harris Detector: Some Properties

- Rotation invariance



Ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner response R is invariant to image rotation

This is a new example where eigen analysis allows us to work in the canonical frame of the linear form

Harris Detector: Some Properties

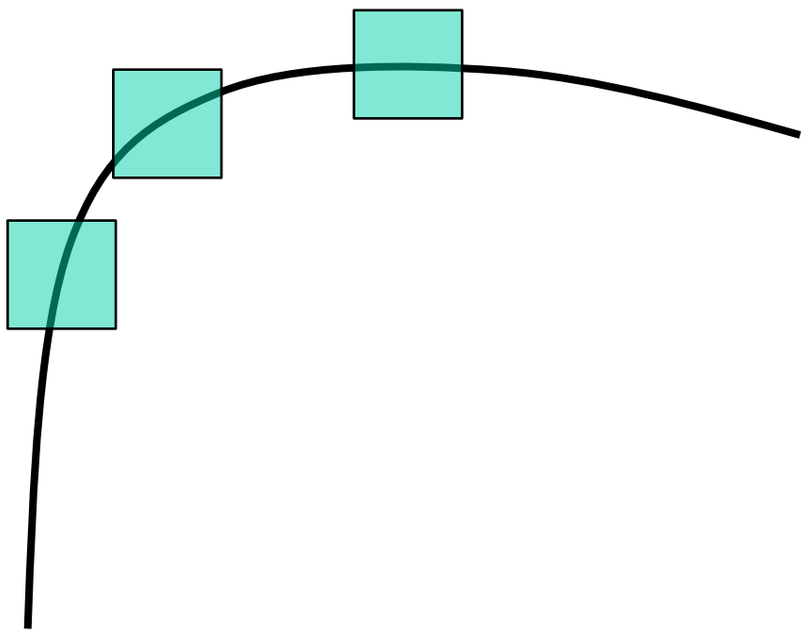
- Invariant to image scale?



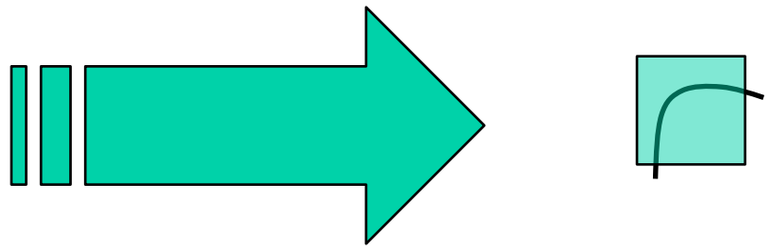


Harris Detector: Some Properties

- Not invariant to *image scale*!



All points will be classified as edges



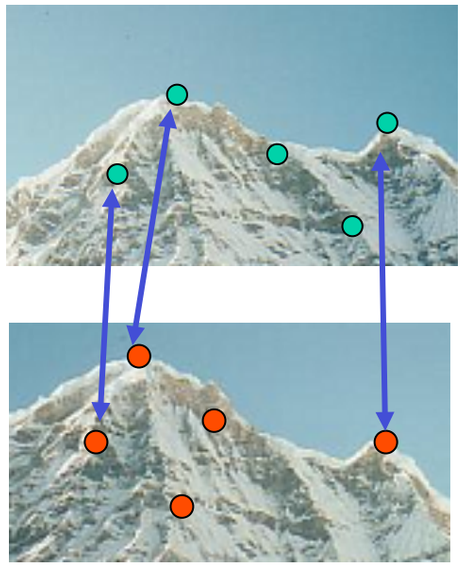
Corner !

Harris Detector: Some Properties

- Quality of Harris detector for different scale changes

Repeatability rate:

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



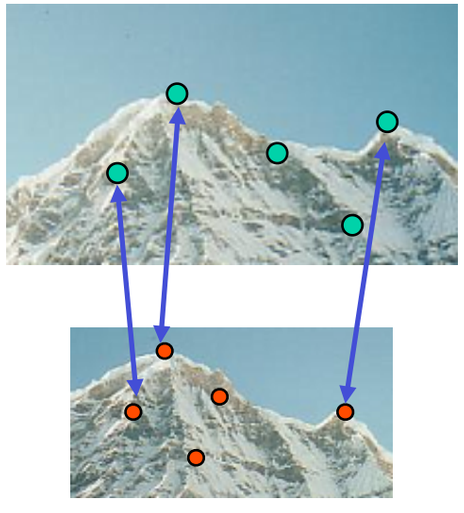
C.Schmid et.al. “Evaluation of Interest Point Detectors”. IJCV 2000

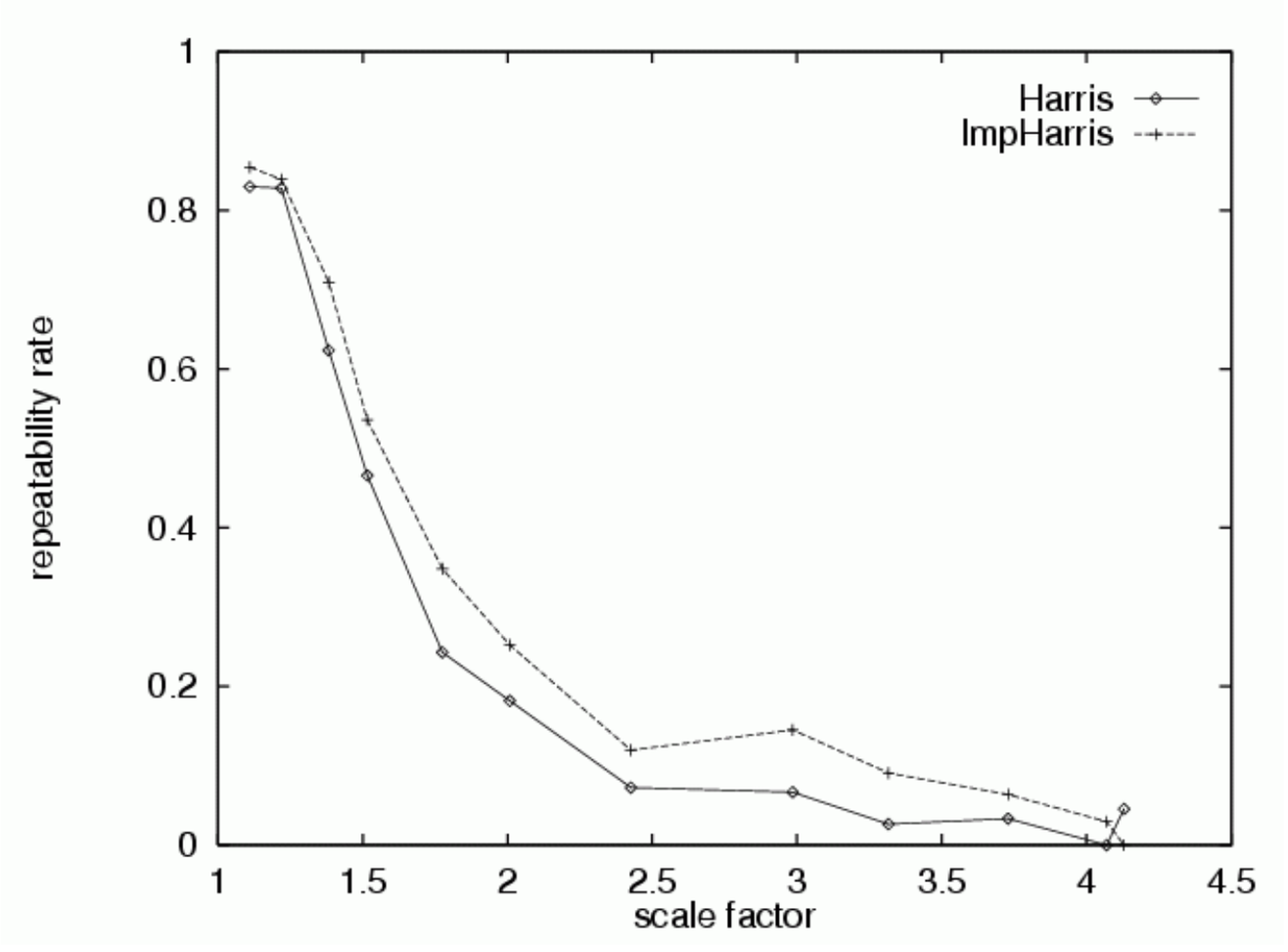
Harris Detector: Some Properties

- Quality of Harris detector for different scale changes

Repeatability rate:

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





C.Schmid et.al. “Evaluation of Interest Point Detectors”. IJCV 2000

Evaluation plots are from this paper



International Journal of Computer Vision 37(2), 151–172, 2000
© 2000 Kluwer Academic Publishers. Manufactured in The Netherlands.

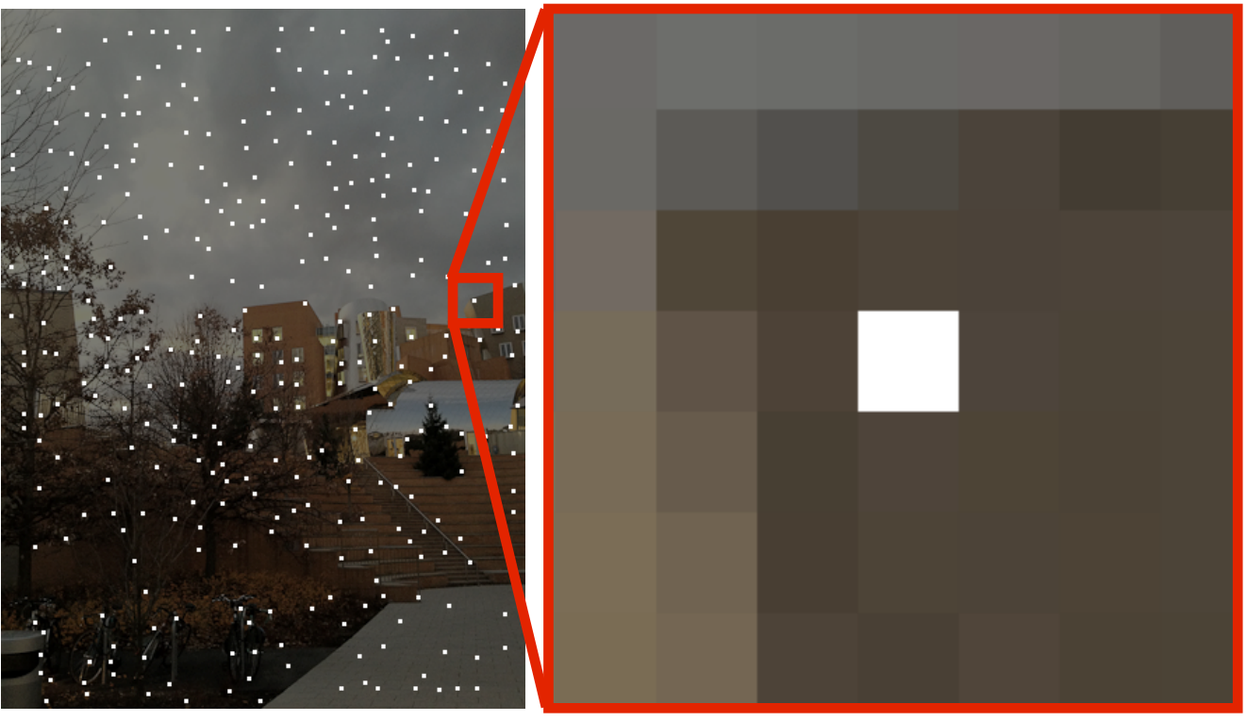
Evaluation of Interest Point Detectors

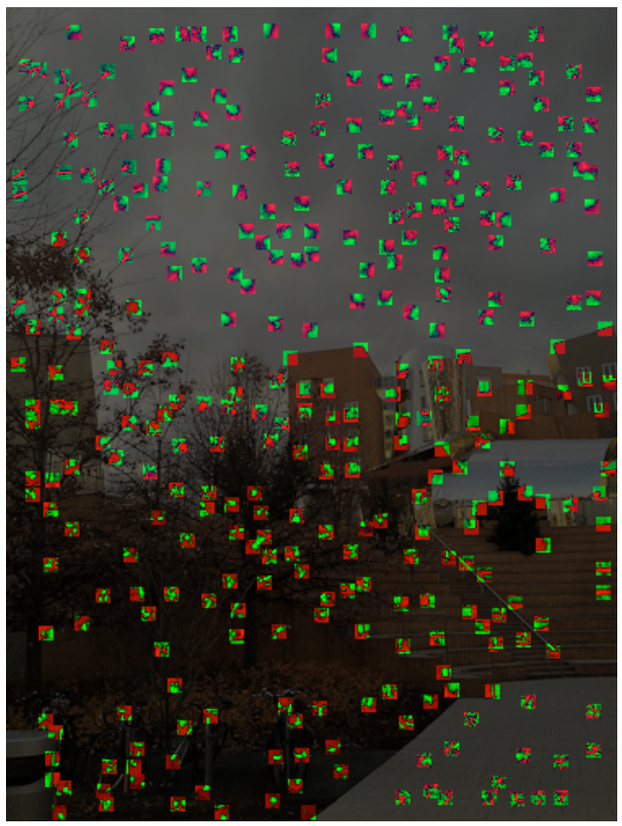
CORDELIA SCHMID, ROGER MOHR AND CHRISTIAN BAUCKHAGE
INRIA Rhône-Alpes, 655 av. de l'Europe, 38330 Montbonnot, France
Cordelia.Schmid@inrialpes.fr

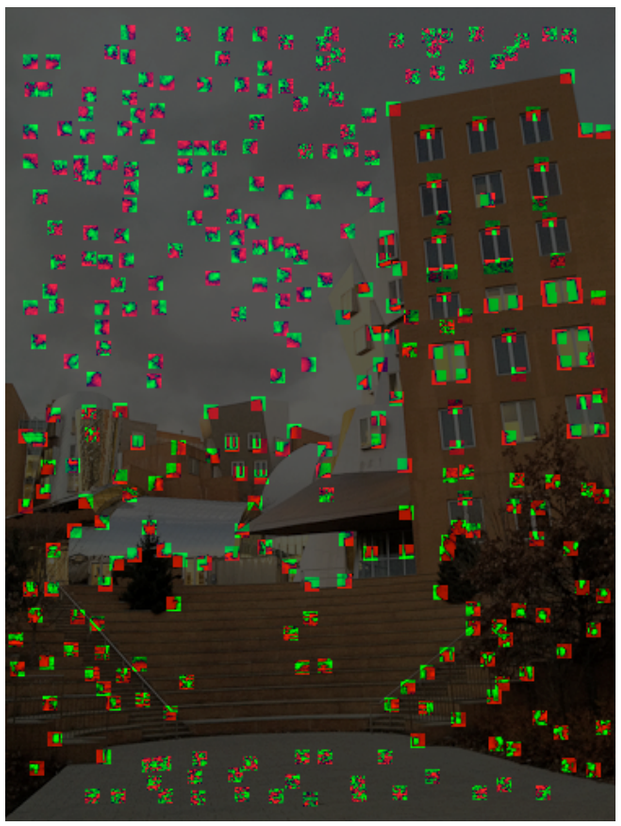
Abstract. Many different low-level feature detectors exist and it is widely agreed that the evaluation of detectors is important. In this paper we introduce two evaluation criteria for interest points: repeatability rate and information content. Repeatability rate evaluates the geometric stability under different transformations. Information content measures the distinctiveness of features. Different interest point detectors are compared using these two criteria. We determine which detector gives the best results and show that it satisfies the criteria well.

What about our descriptor

- Not invariance to scale or rotation







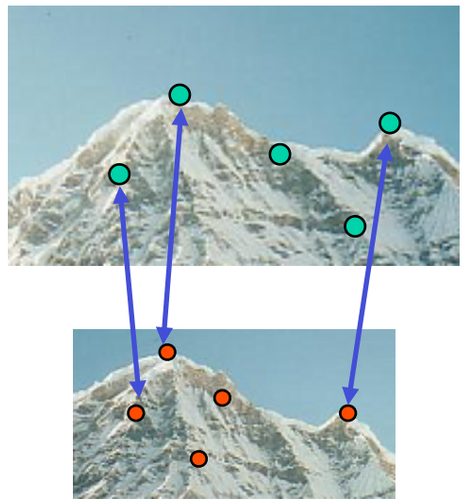
Goal

- **Compute stable features in space and scale**
- **magically compute some local characteristic scale and angle**
  - we want these characteristic scale and angle to be the same if we run it on a scaled or rotated version of the image
- **extract descriptors from window given by x,y, scale and angle**
  - Ideally, the descriptor will be the same if we run it on a scaled or rotated version of the image

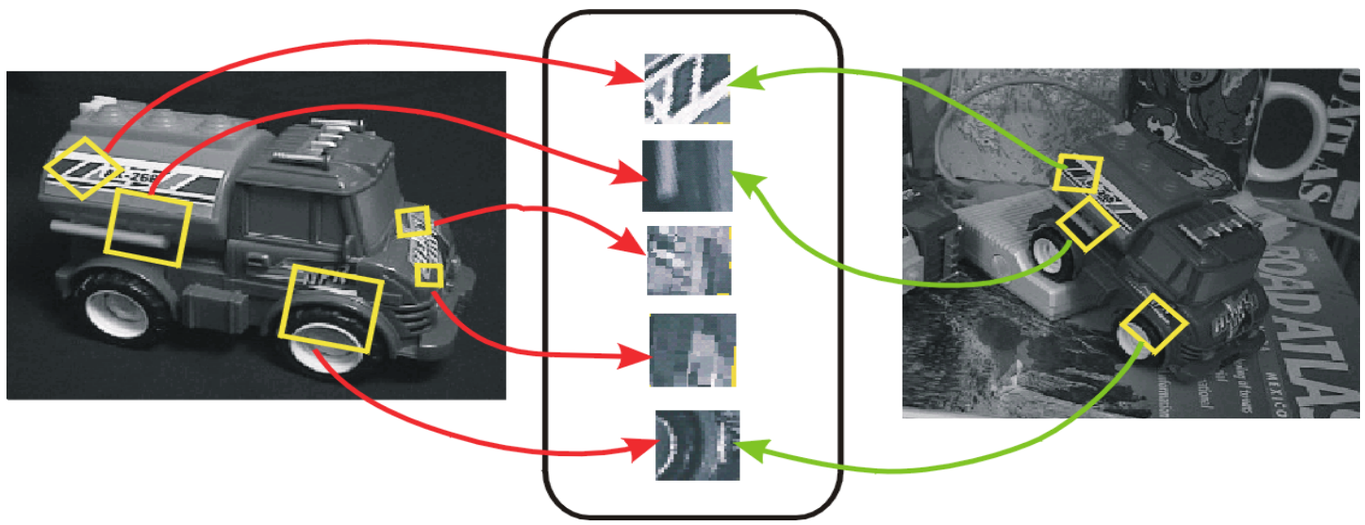
Invariance to rotation

Goals for handling scale/rotation

- Ensure invariance
  - Same scene points must be detected in all images

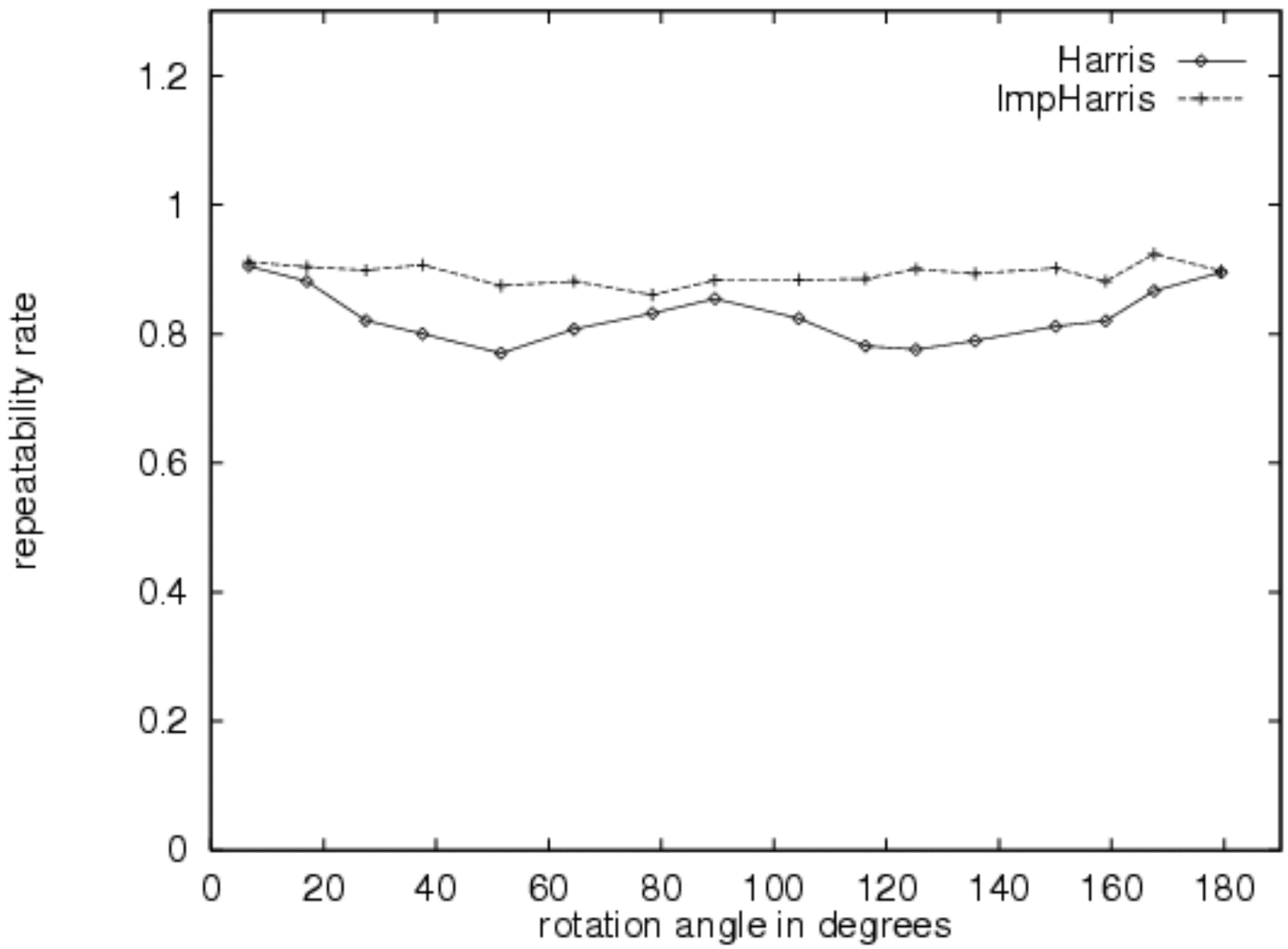


- Prepare terrain for descriptor
  - Descriptors are based on a local neighborhood
  - Find canonical scale and angle for neighborhood (independently in all images)



Rotation Invariant Detection

- Harris Corner Detector



This gives us rotation invariant *detection*, but we'll need to do more to ensure a rotation invariant *descriptor*...

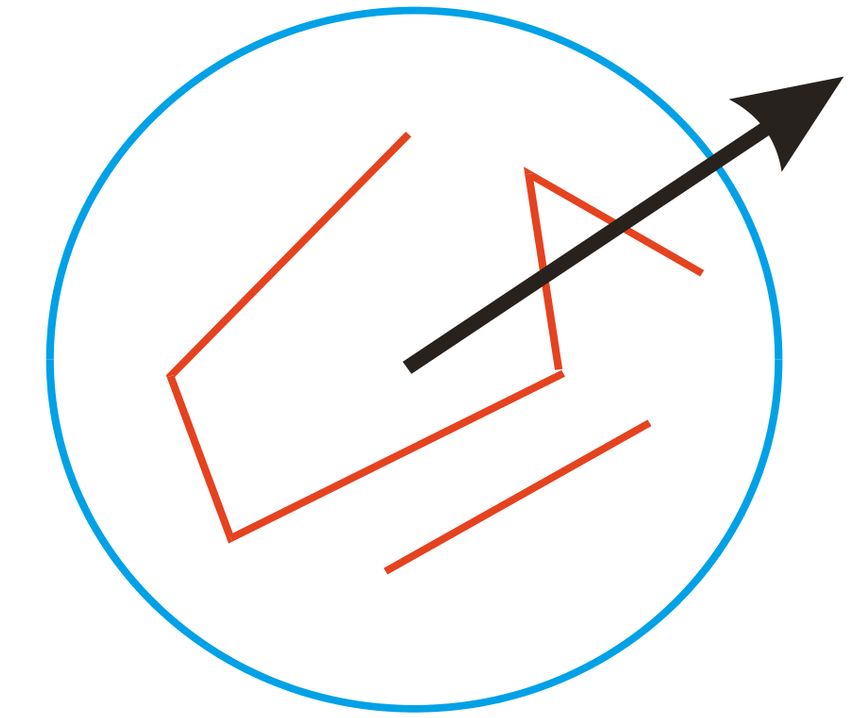
C.Schmid et.al. "Evaluation of Interest Point Detectors". IJCV 2000

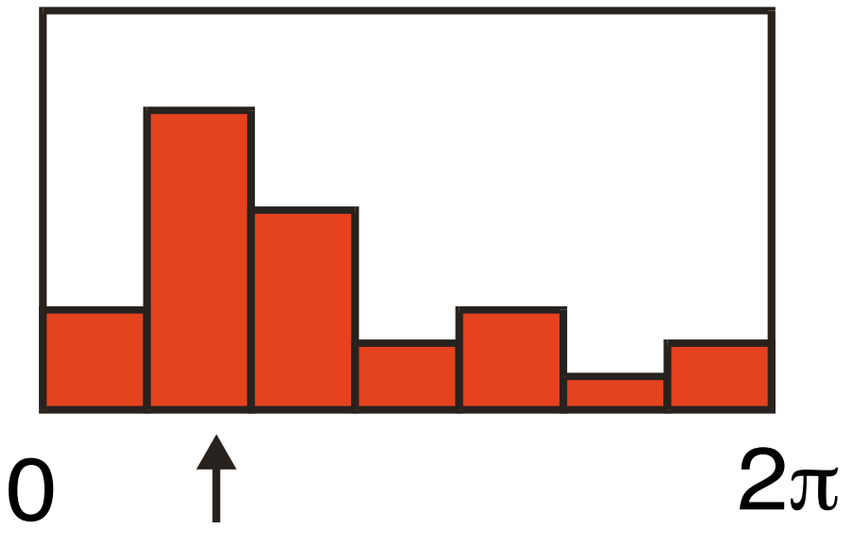
Select canonical orientation for descriptor

- One option: use maximum eigenvector in Harris

Alternative

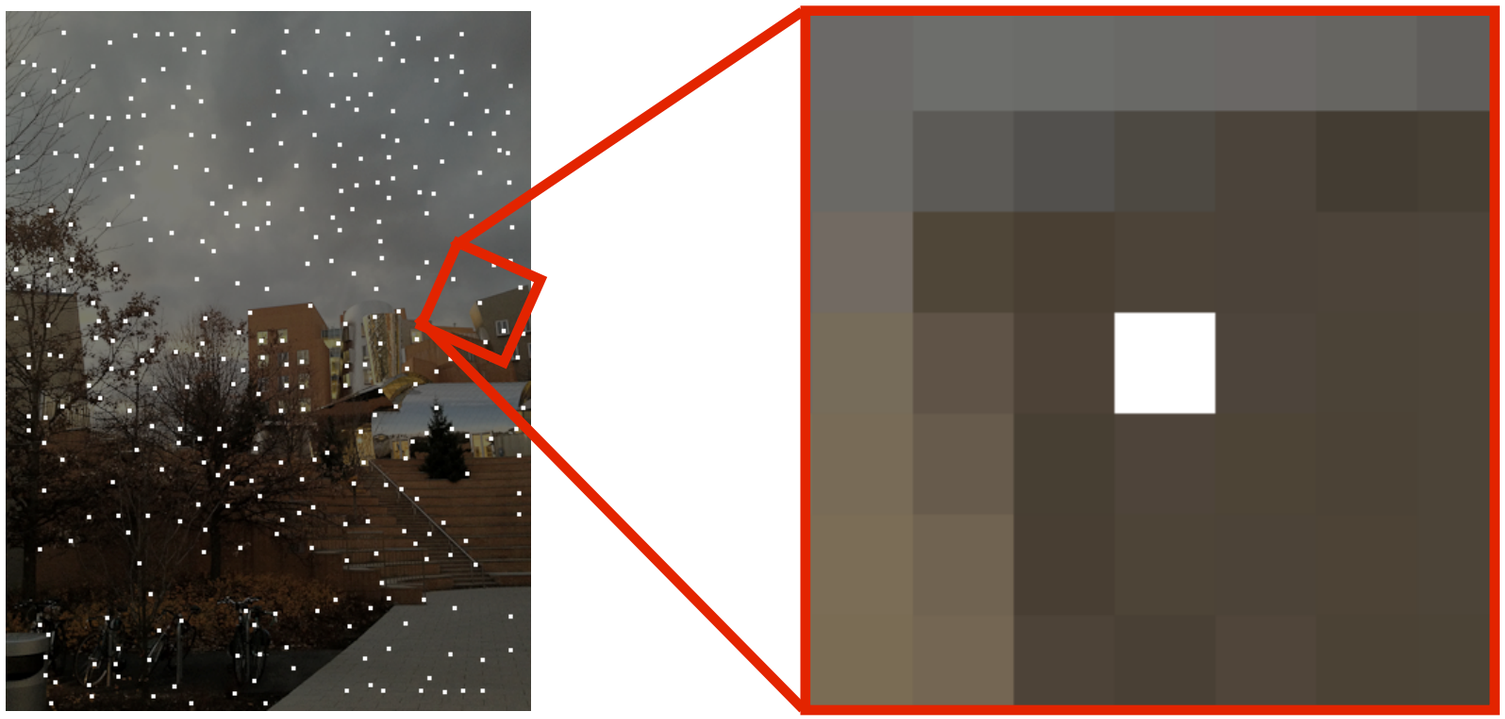
- Create histogram of local gradient directions computed at selected scale
- Assign canonical orientation at peak of smoothed histogram
- Each key specifies stable 2D coordinates (x, y, scale, orientation)





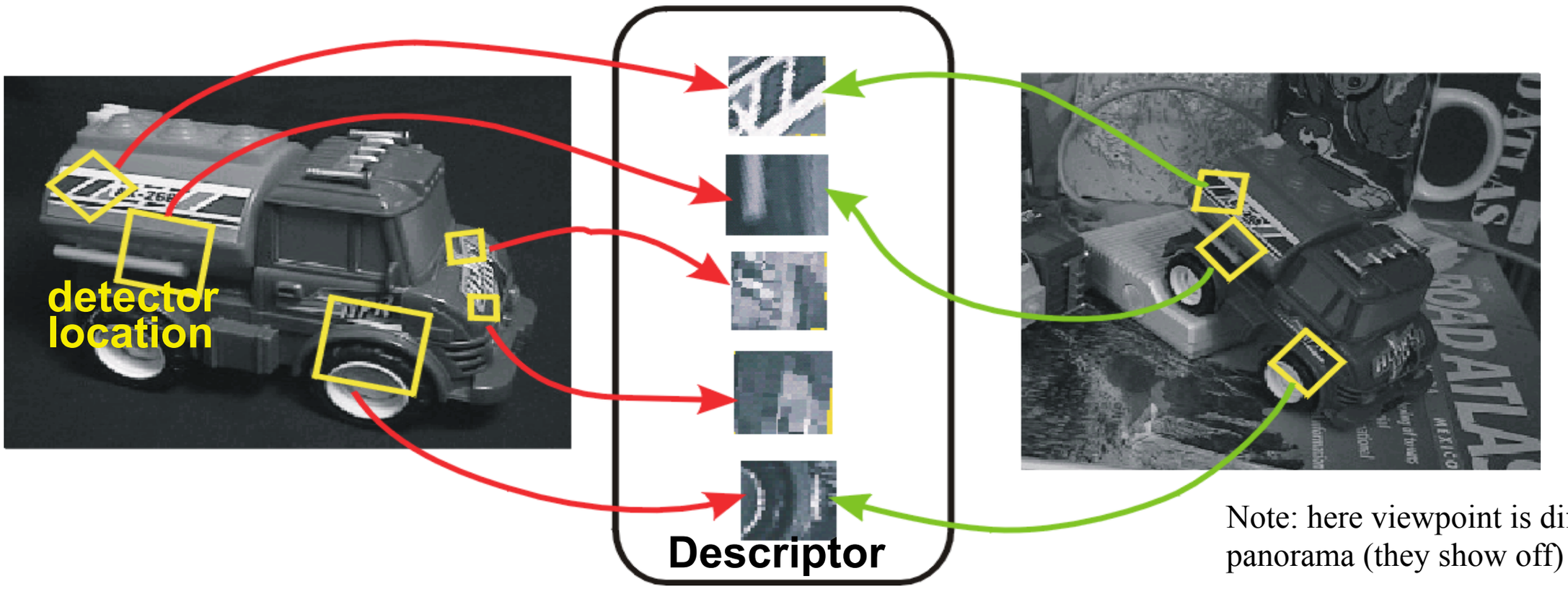
Using canonical orientation

- For our simple descriptor:
  Use pixel values from image rotated according to canonical orientation



In general

- **Descriptor**: encode local neighboring window
  - Scale & rotation of window are the same in both image (but computed independently)



Note: here viewpoint is different, not panorama (they show off)

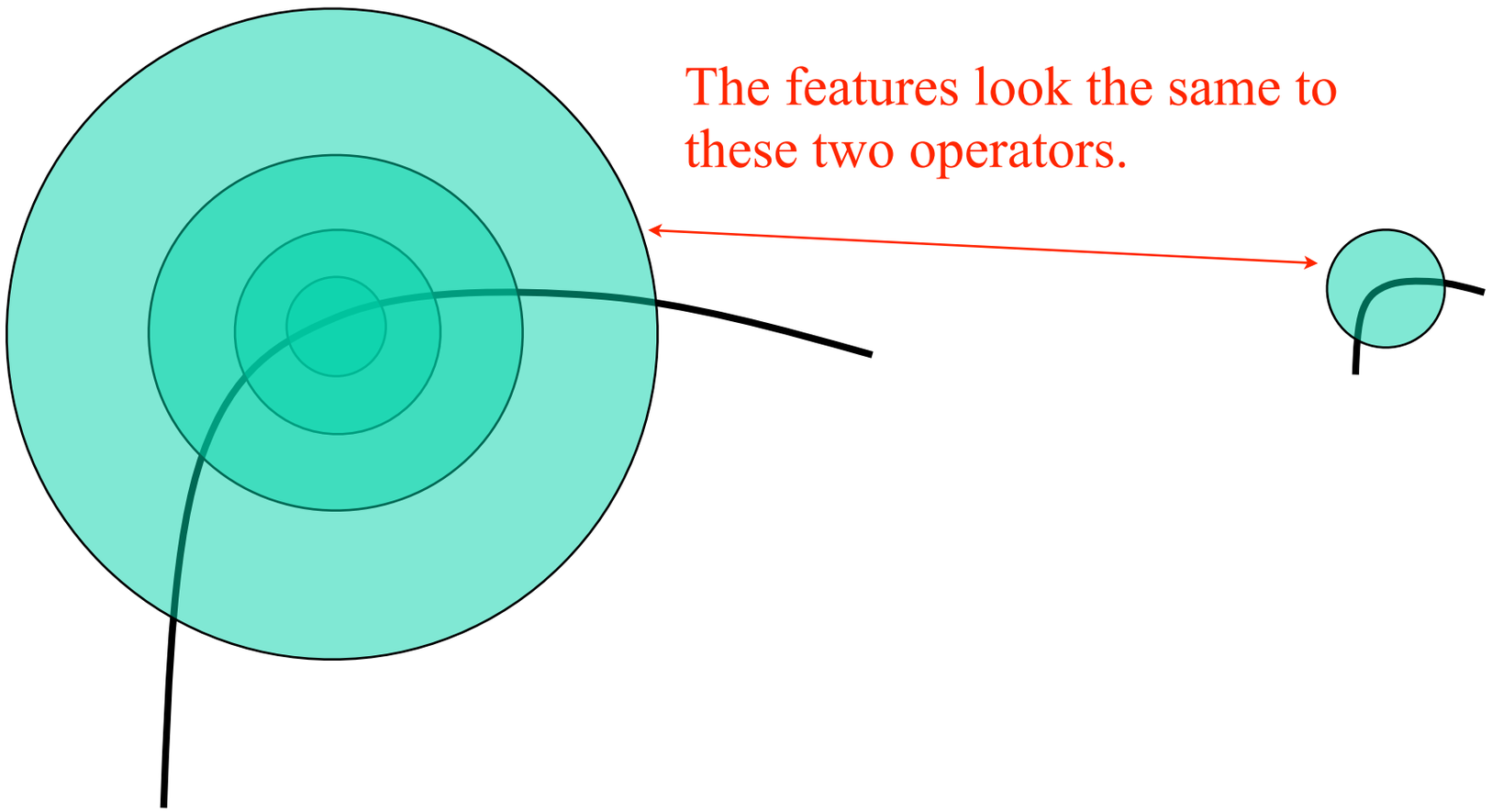
Invariance to scale

Warning: Two notions of scale

- Scale factor between two images
- Size of window that's good for a feature

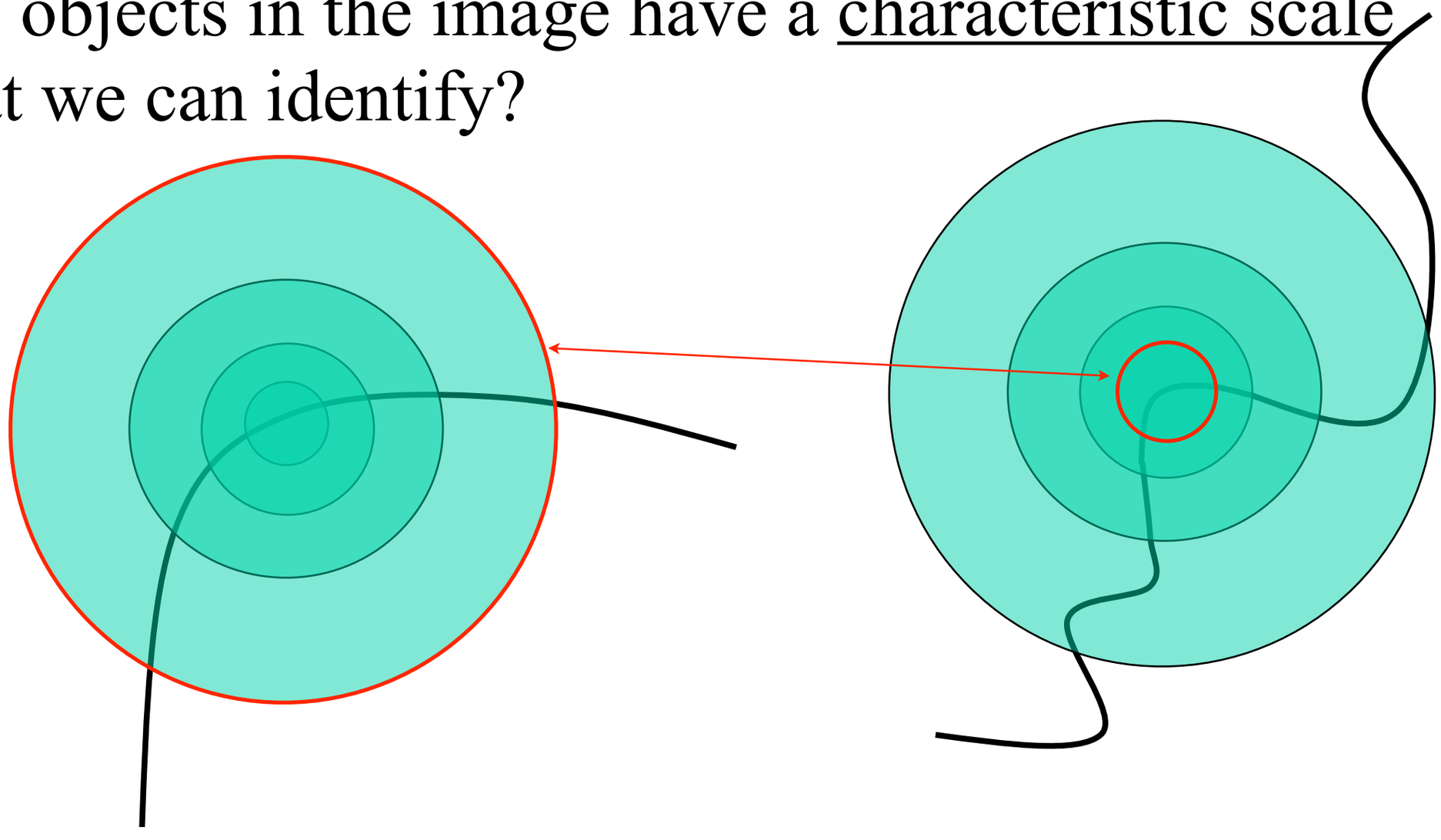
Scale Invariant Detection

- Consider regions (e.g. circles) of different sizes around a point
- Regions of corresponding sizes will look the same in both images



Scale Invariant Detection

- The problem: how do we choose corresponding circles ***independently*** in each image?
- Do objects in the image have a characteristic scale that we can identify?

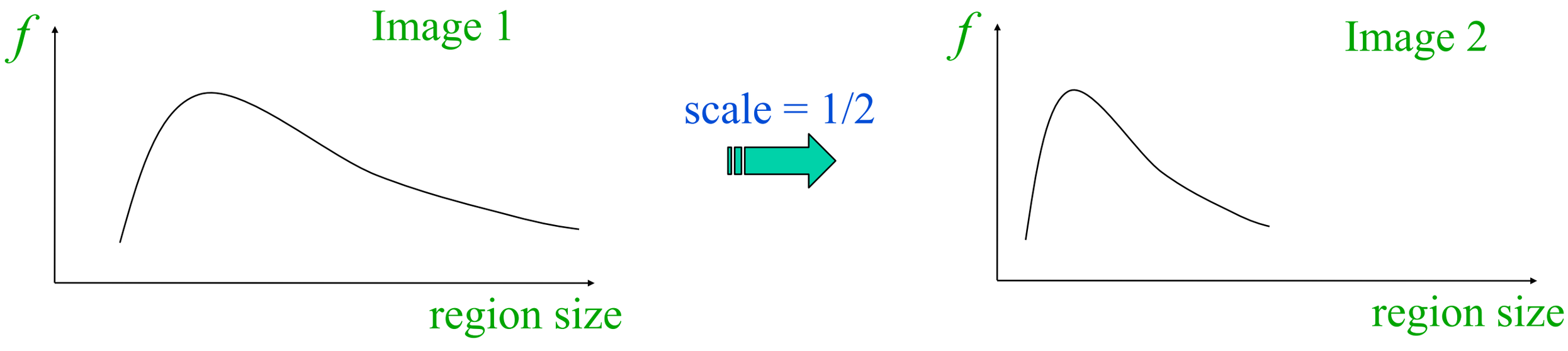


Scale Invariant Detection

- Solution:
  - Design a function on the region (circle), which is "scale invariant" (the same response for corresponding, scaled regions, even if they are at different scales)

Example: average intensity. For corresponding regions (even of different sizes) it will be the same.

  - For a point in one image, we can consider it as a function of region size (circle radius)



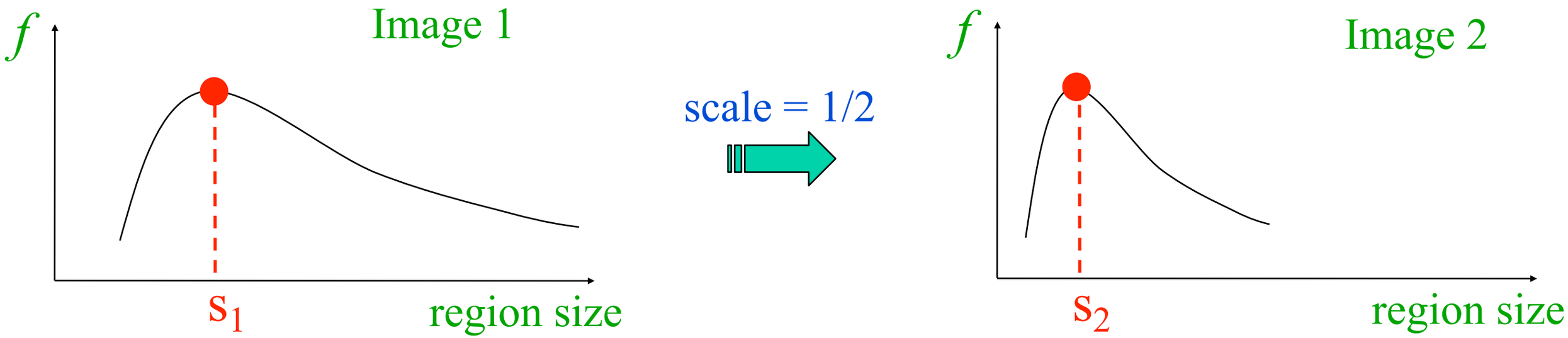
Scale Invariant Detection

- Common approach:

  Take a **local maximum** (wrt radius) of this function

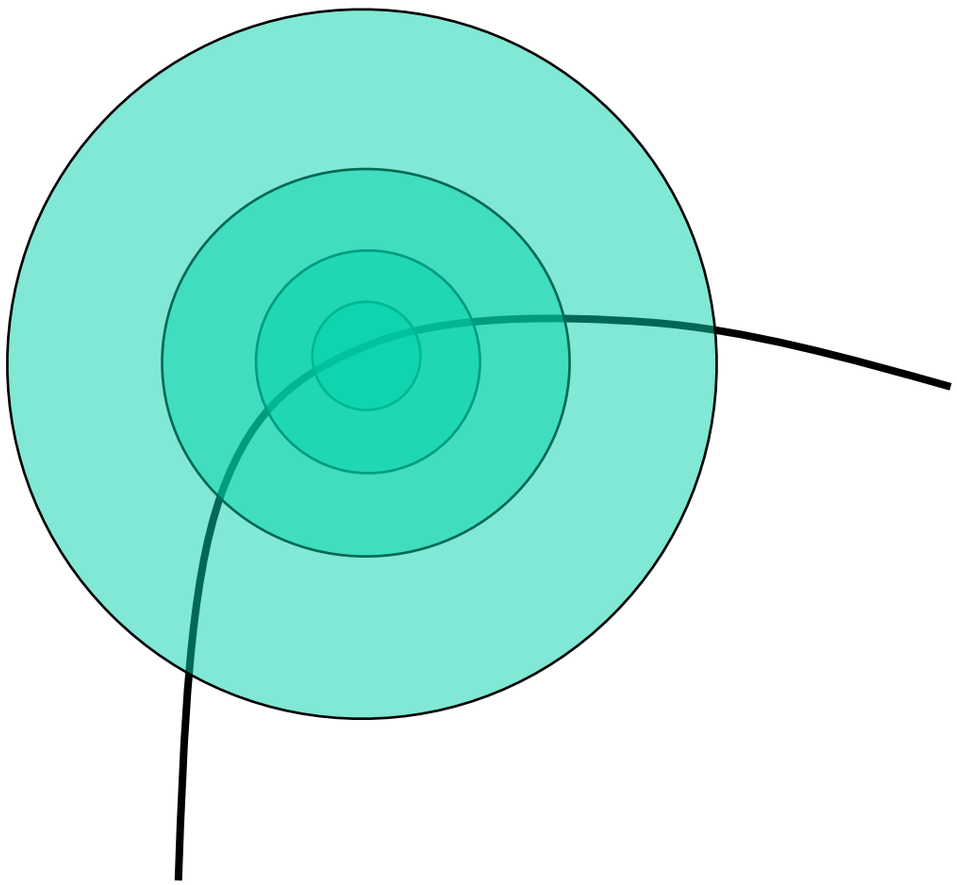
  Observation: region size, for which the maximum is achieved, should be *invariant* to image scale, and only sensitive to the underlying image data.

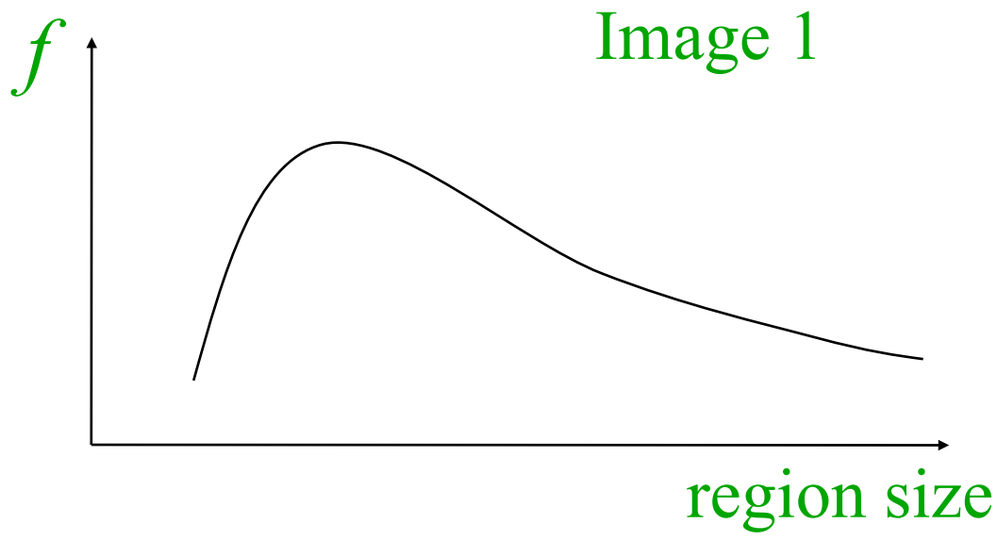
Important: this scale invariant region size is found in each image independently!



Recap scale invariance basic idea

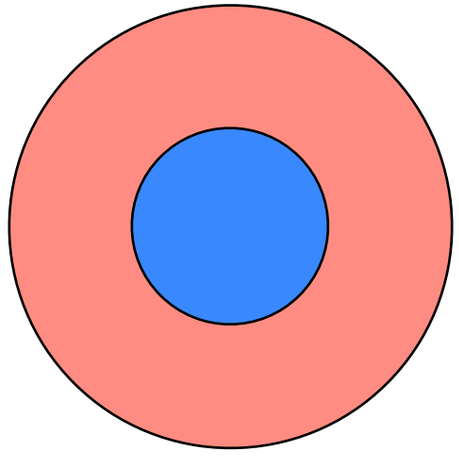
- Compute scale-invariant function over window
- Extract extrema along scale





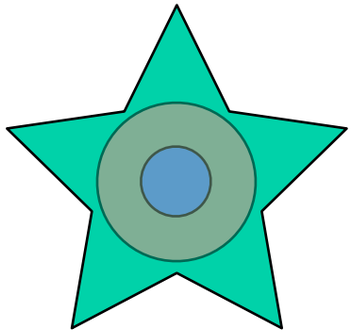
Scale-invariant function

- Average intensity is boring and not peaky
  - maximum would not be robust
- Instead, compute notion of contrast
  - Difference between the center and surround of the window

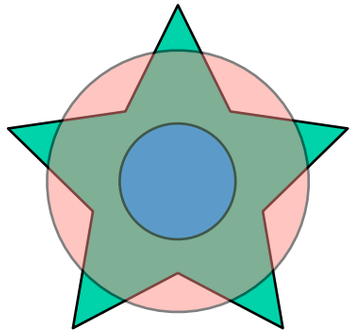


Center-surround across scale

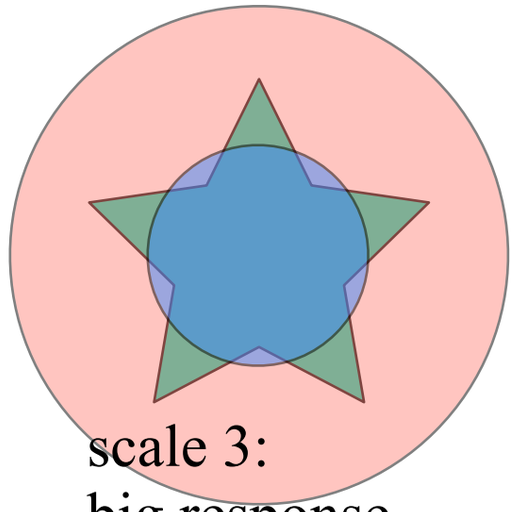
- We want to analyze a “star” image



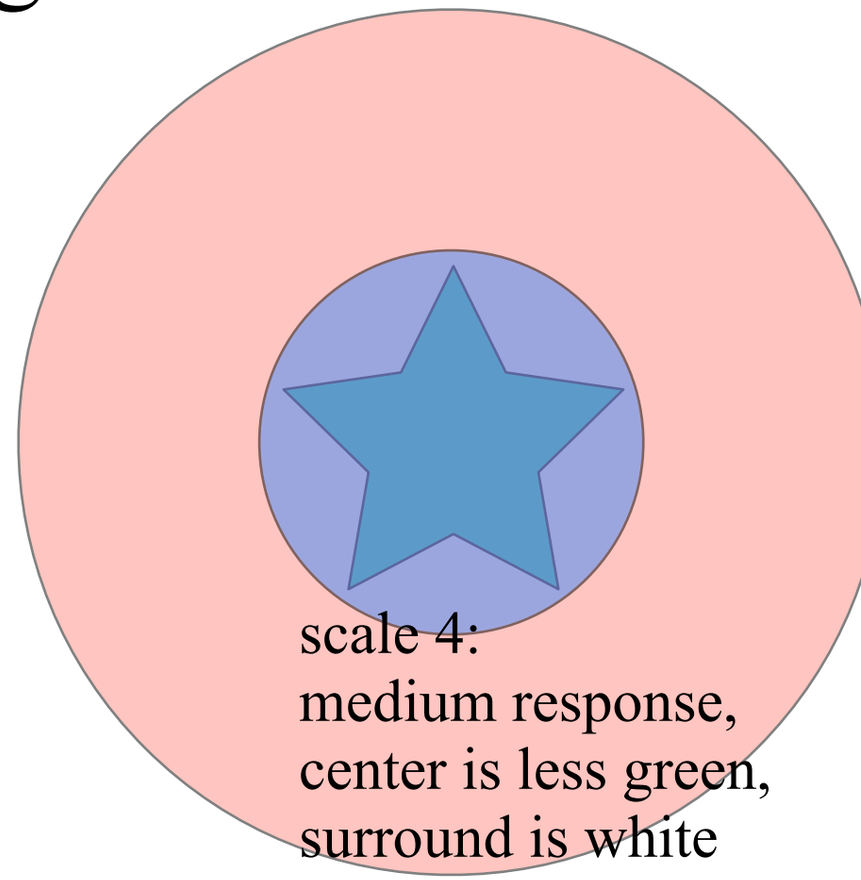
small scale:
zero response,
center has same
color as surround



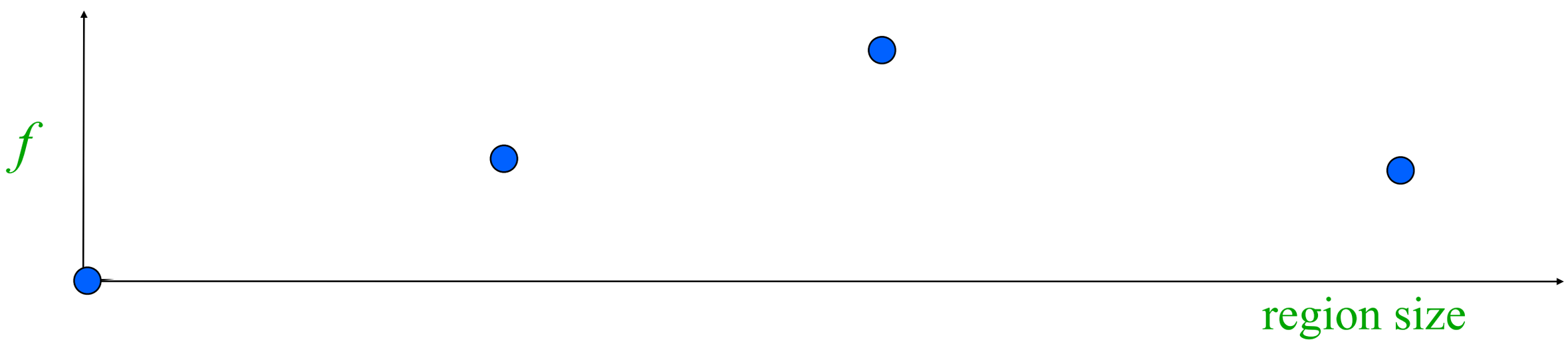
scale 2:
small response,
surround has
some white



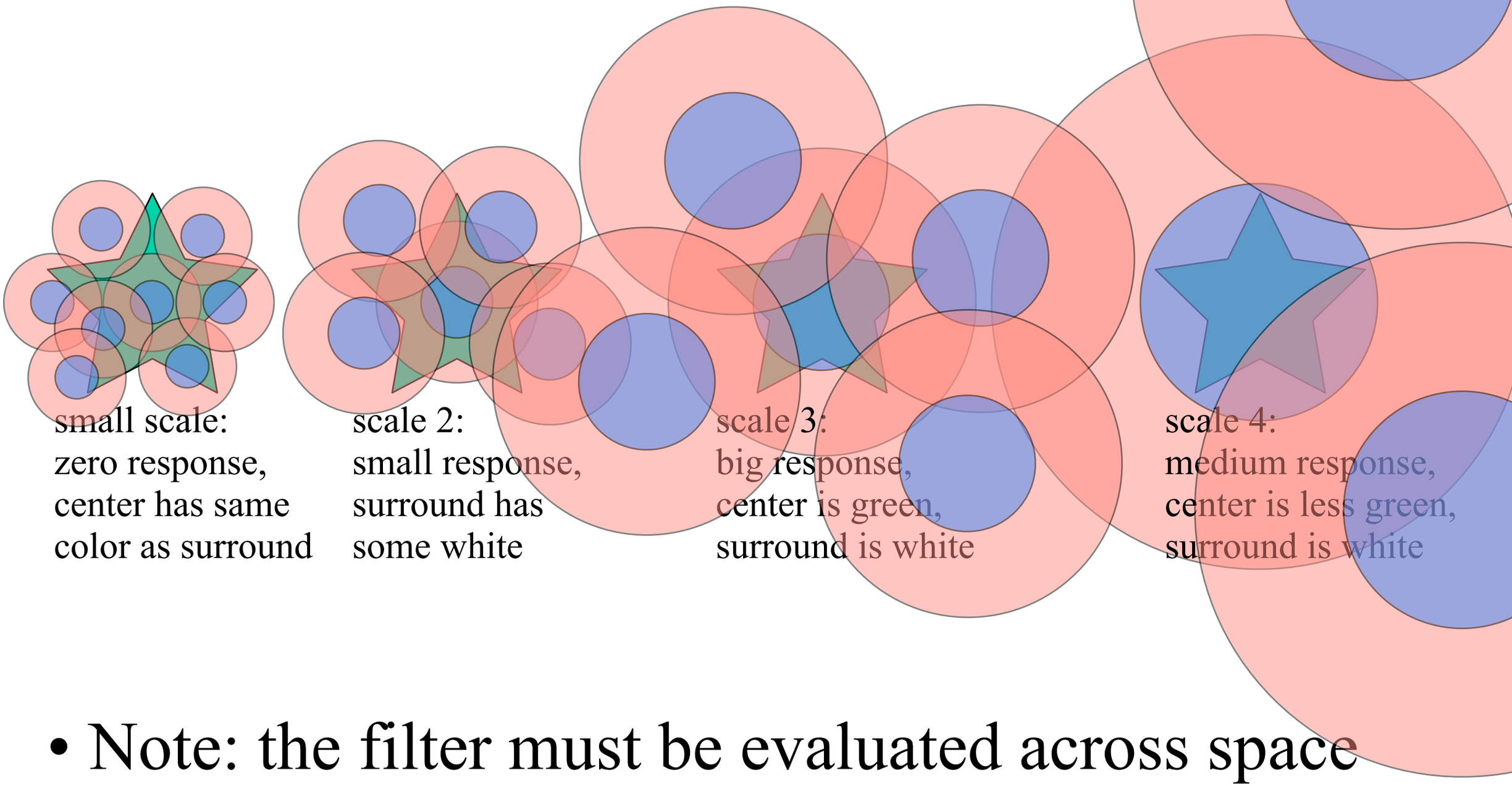
scale 3:
big response,
center is green,
surround is white



scale 4:
medium response,
center is less green,
surround is white



Center-surround



small scale:
zero response,
center has same
color as surround

scale 2:
small response,
surround has
some white

scale 3:
big response,
center is green,
surround is white

scale 4:
medium response,
center is less green,
surround is white

- Note: the filter must be evaluated across space as well: Extrema over space & scale

Center-surround in practice

- Functions for determining scale

$$f = \text{Kernel} * \text{Image}$$

where * means convolution

Kernels:

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

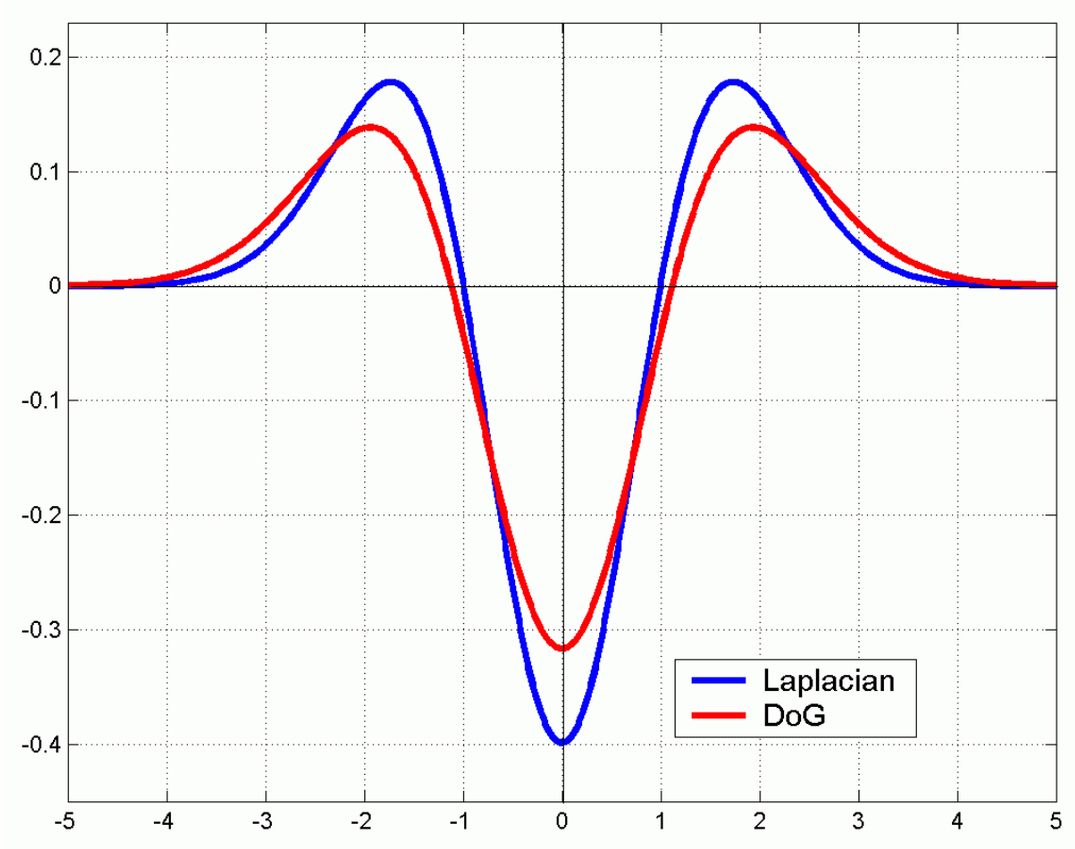
(Laplacian: 2nd derivative of Gaussian)

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

(Difference of Gaussians)

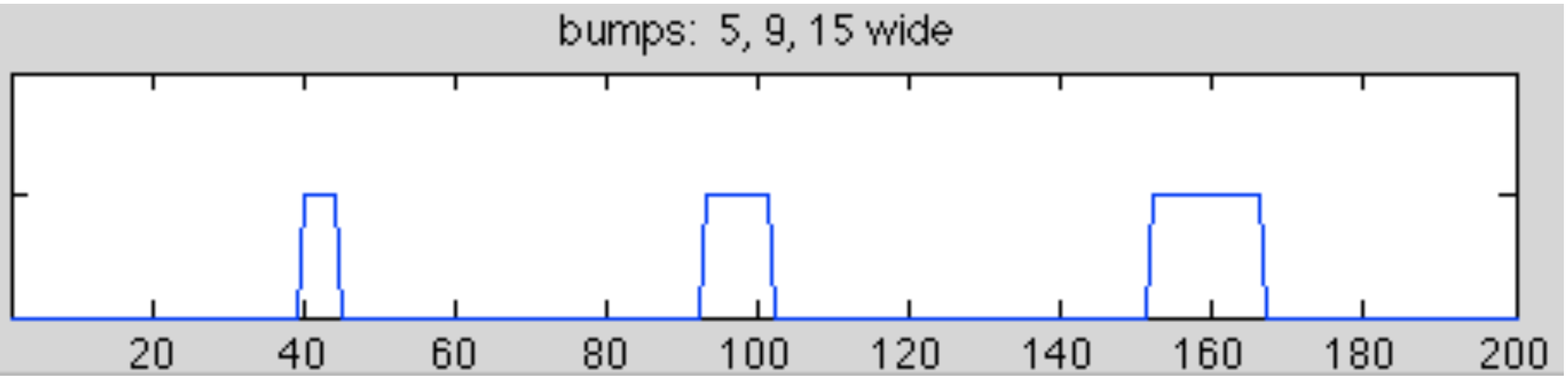
where Gaussian

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



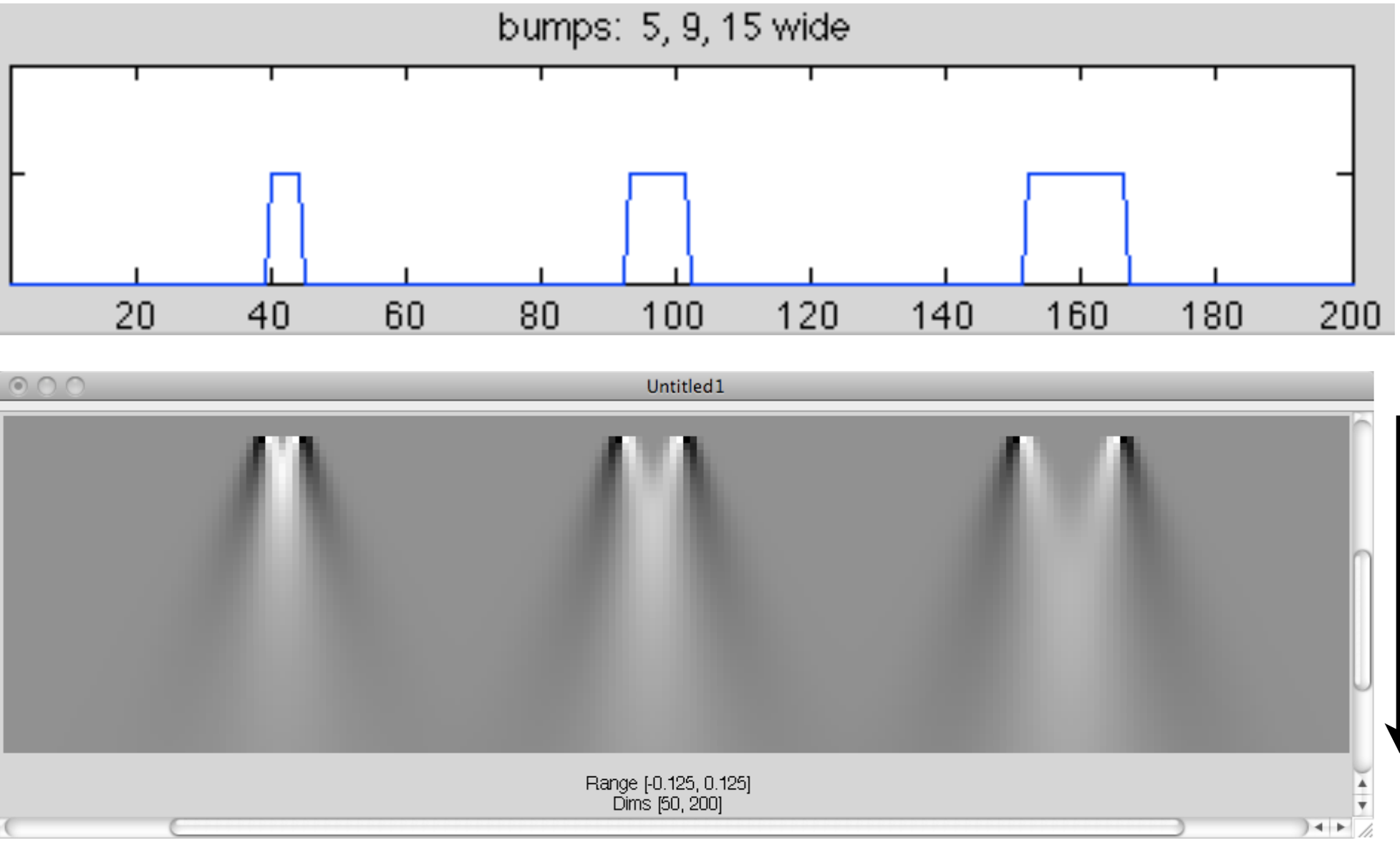
Note: both kernels are invariant to *scale* and *rotation*

Three 1D bumps of different sizes



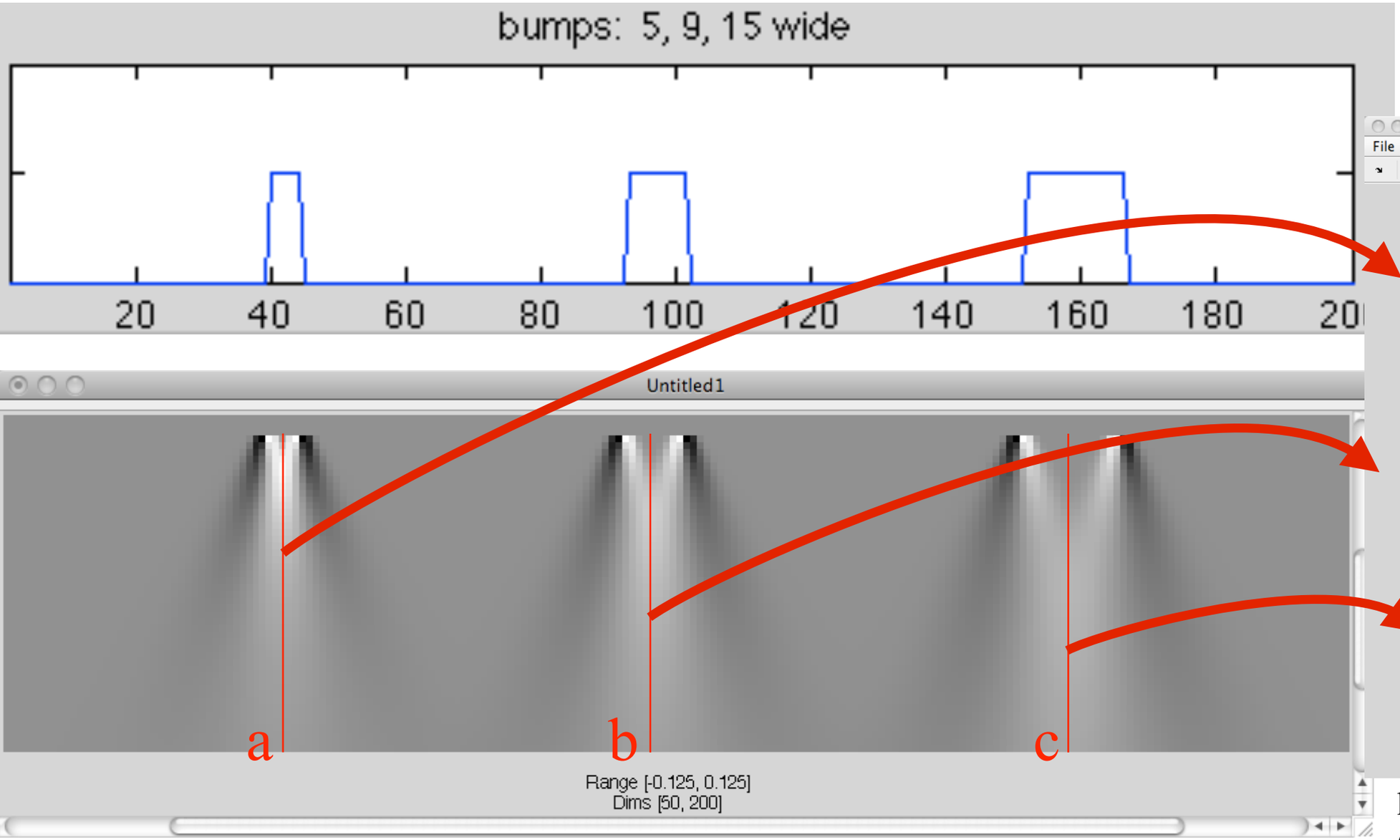
Let’s see how our difference of Gaussian works ideally, it would find a scale proportional to each bump’s size

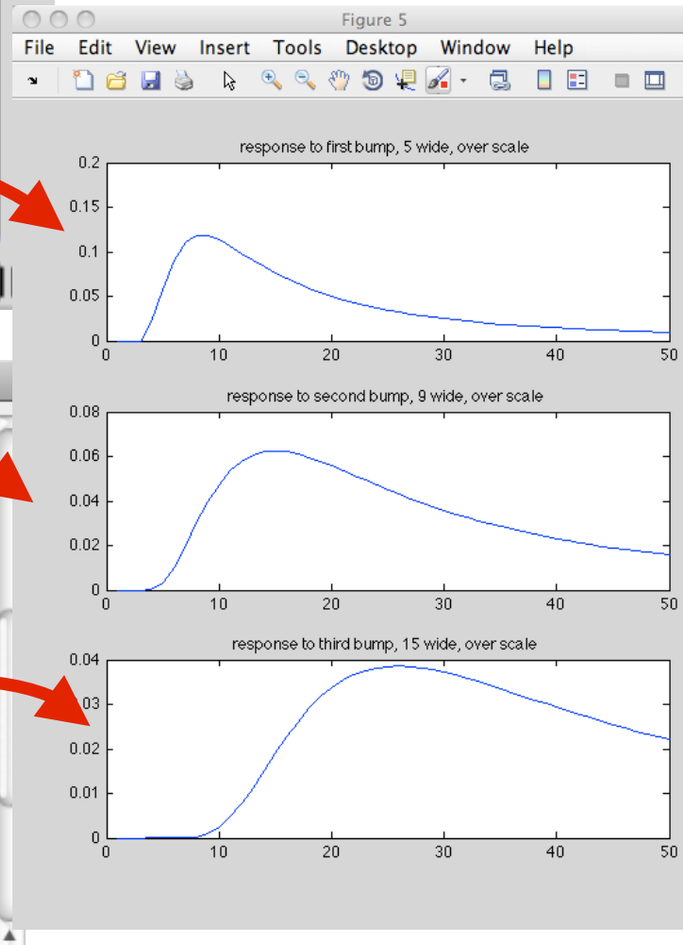
Bumps, filtered by difference-of-Gaussian



Bumps, filtered by difference-of-Gaussian

Consider response across scale at center of bumps



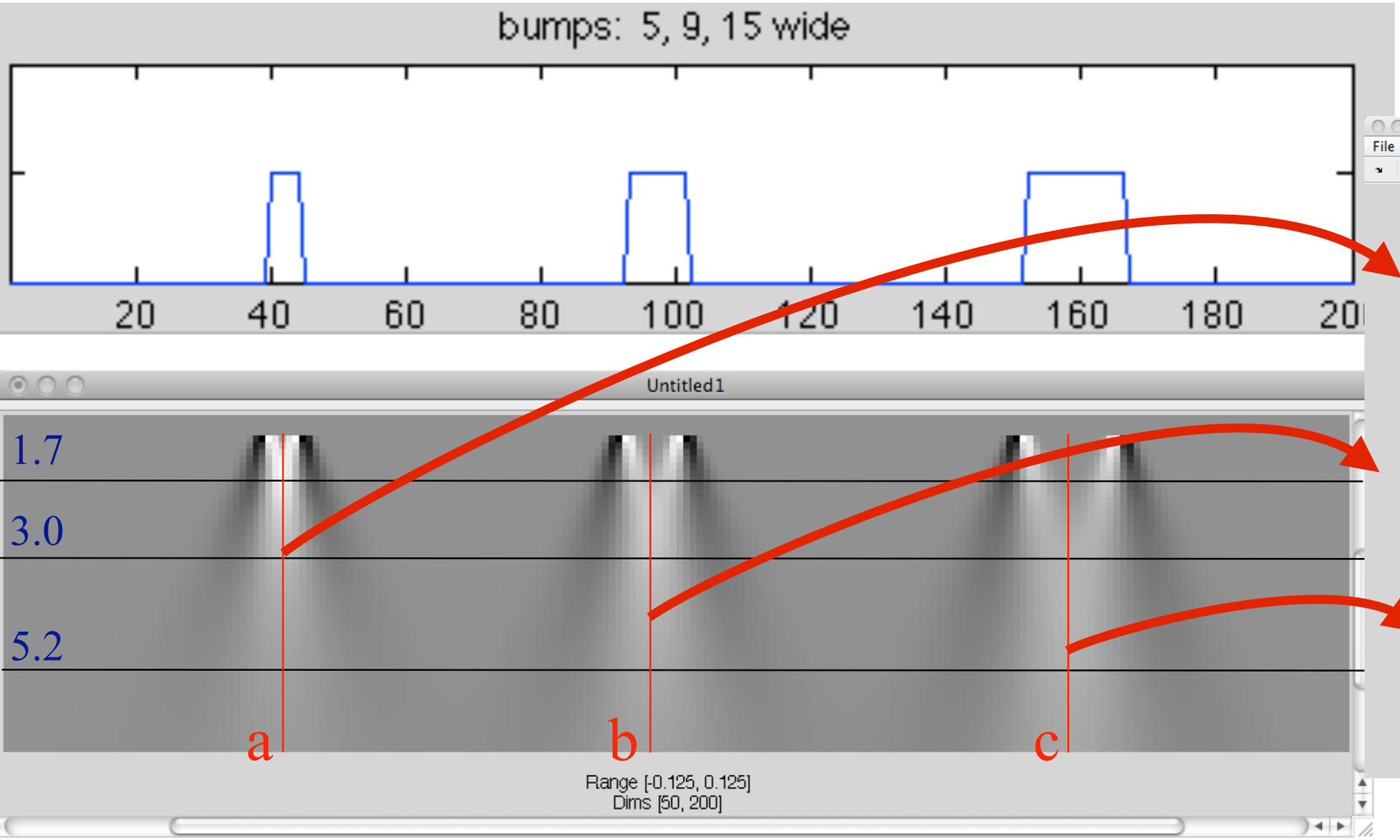


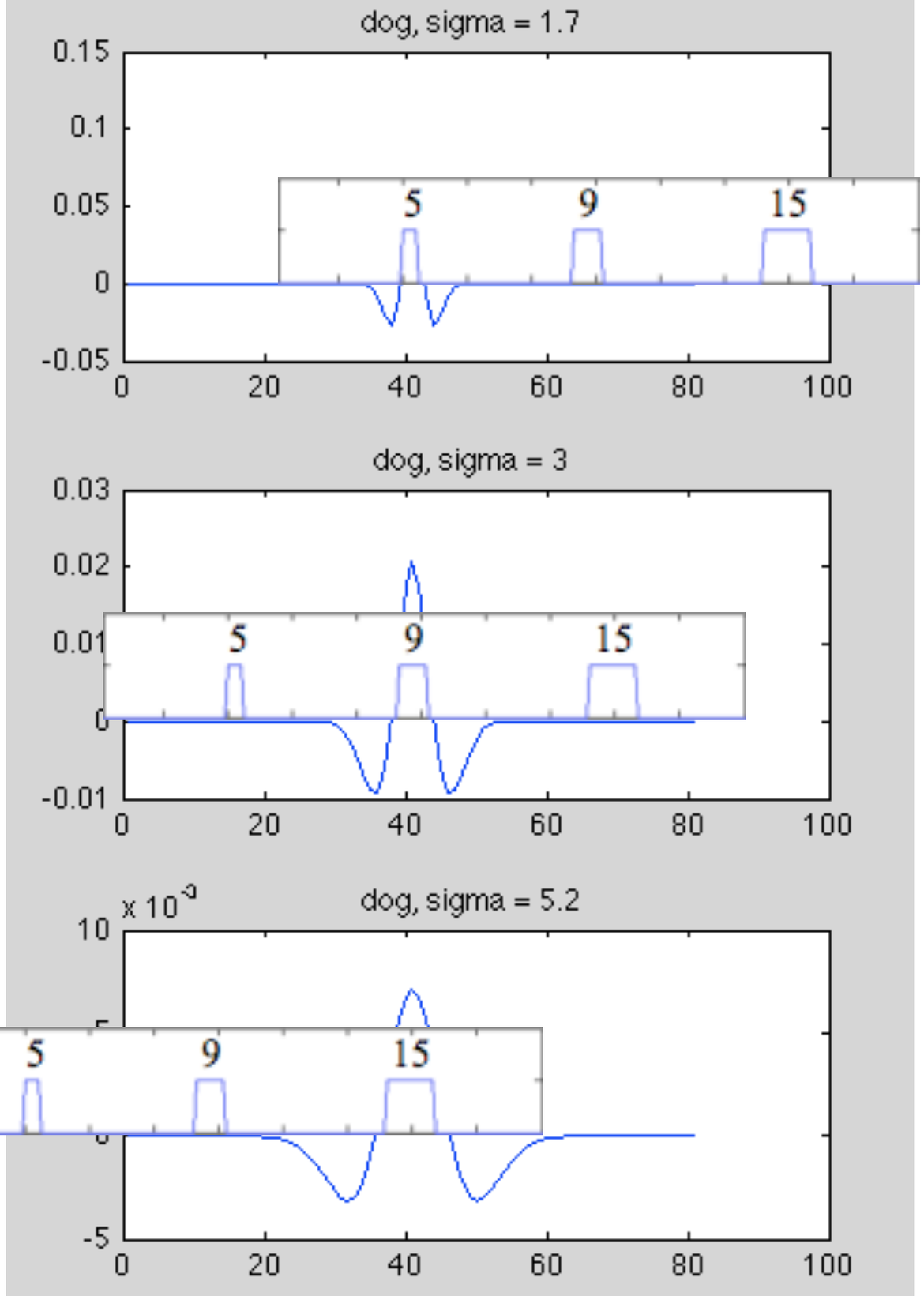
horizontal axis is now scale
(but numbers are arbitrary :-(

Bumps, filtered by difference-of-Gaussian

Scales of peak responses are proportional to bump width

[1.7, 3, 5.2] ./ [5, 9, 15] = 0.3400 0.3333 0.3467



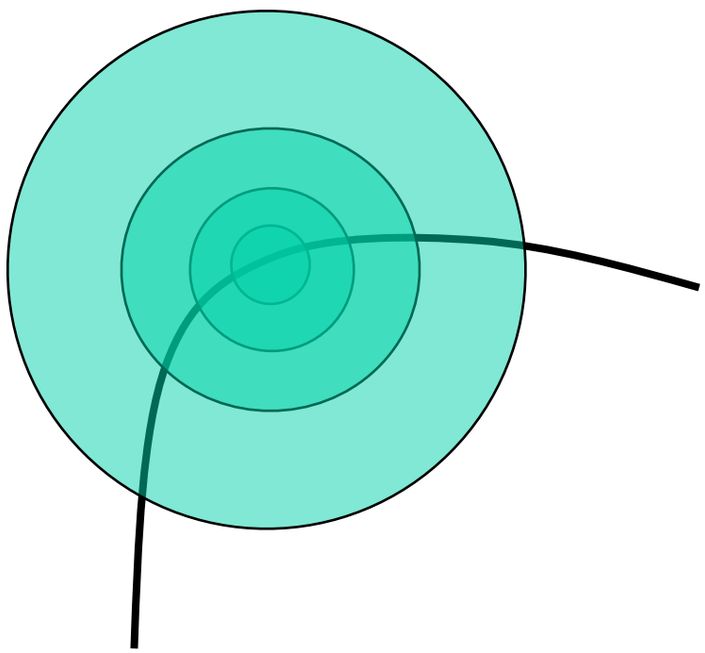


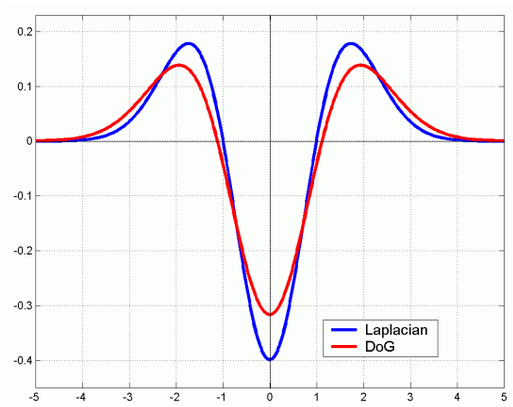
Visualize difference of Gaussian that achieves the extremum together with bumps

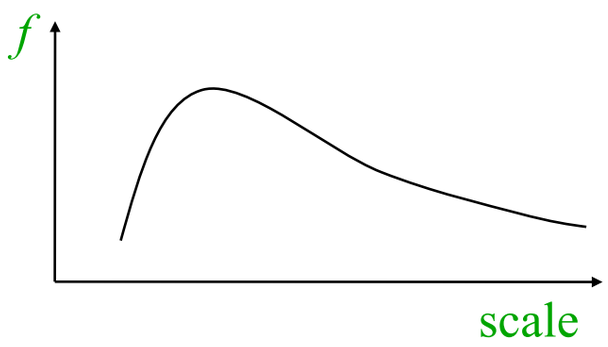
Note how the positive bump exactly matches the bump

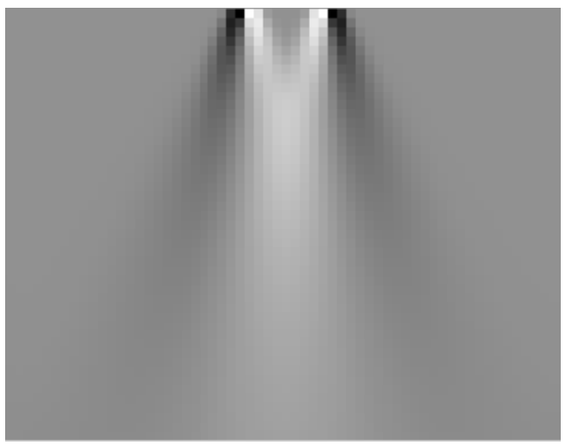
Recap scale invariance

- Compute scale-invariant function over window
  - center surround, difference of Gaussian
- Extract extrema along space & scale







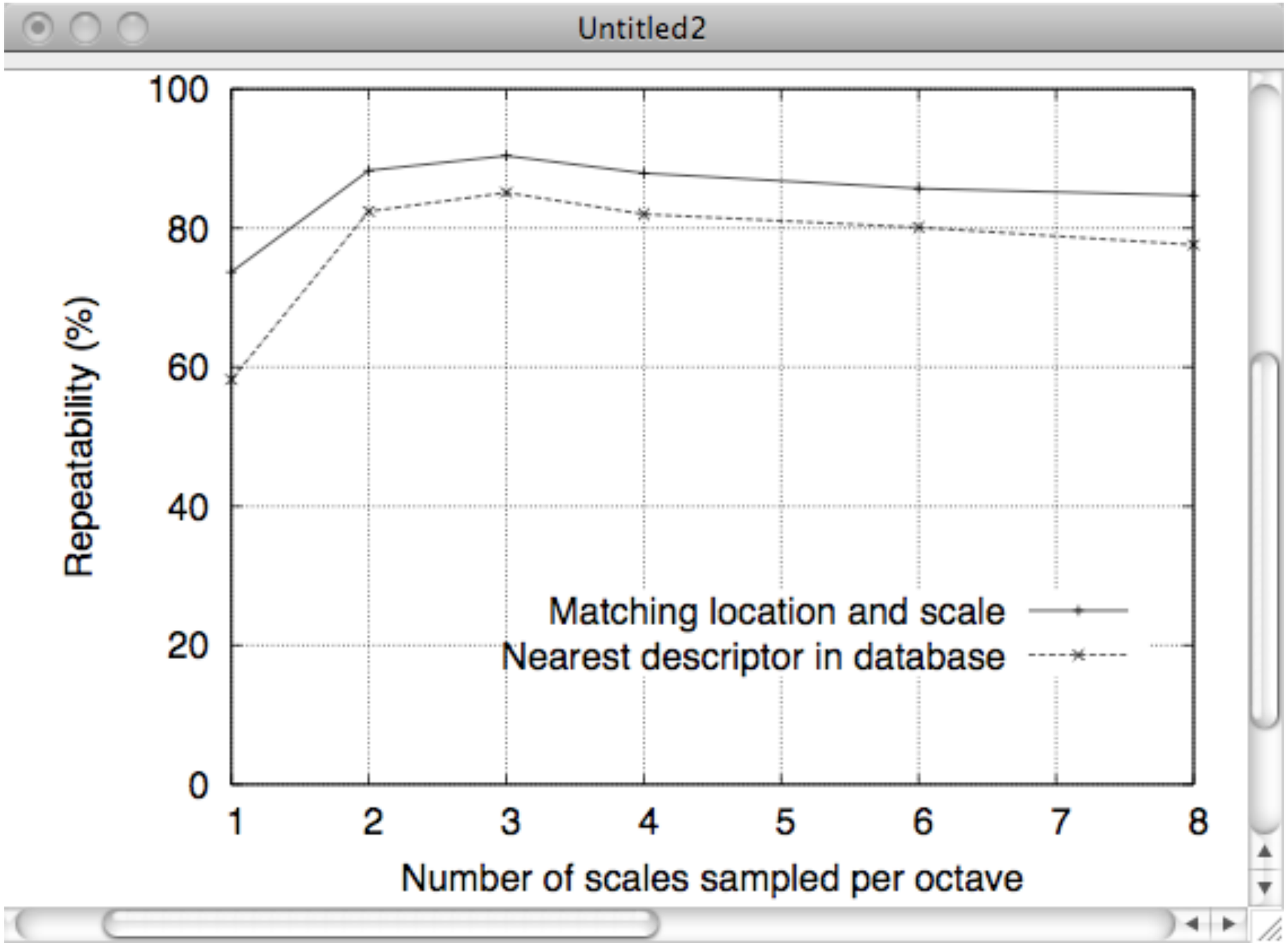


Scale discretization

- We need to find extrema over scale
- Brute force search: compute image at a discrete set of scales and apply center-surround
  - Trick: re-use a given Gaussian for center and surround of two adjacent scales
- Question: how finely do we subdivide scale?

Repeatability vs number of scales sampled per octave

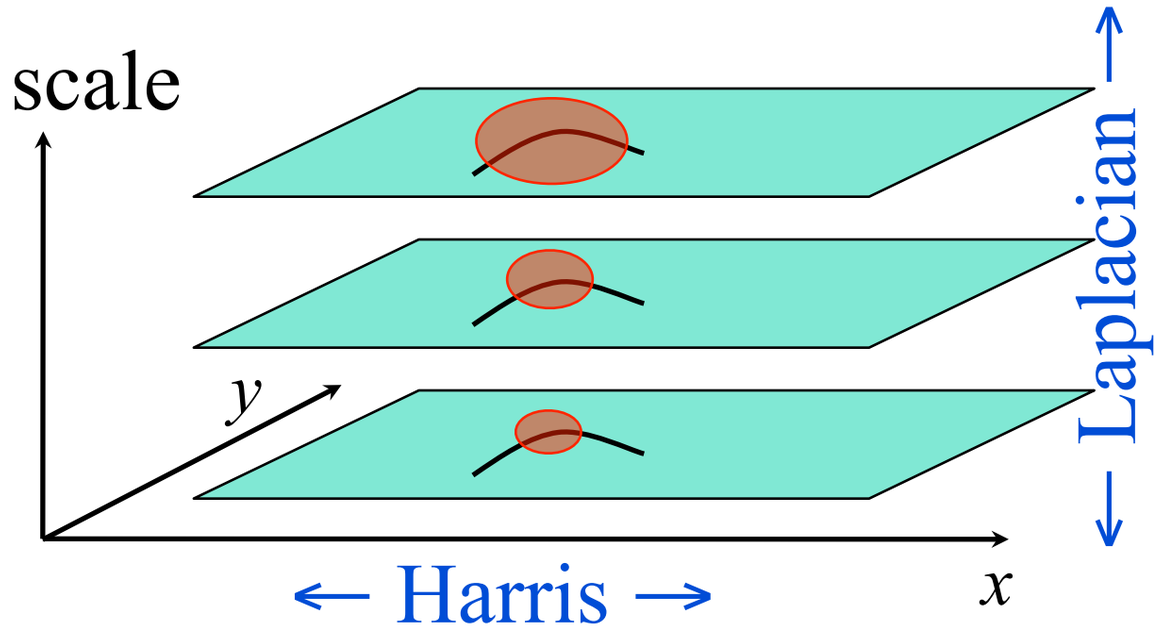
Octave: factor of 2 in image scale



David G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, 60, 2 (2004), pp. 91-110

Scale Invariant Detectors

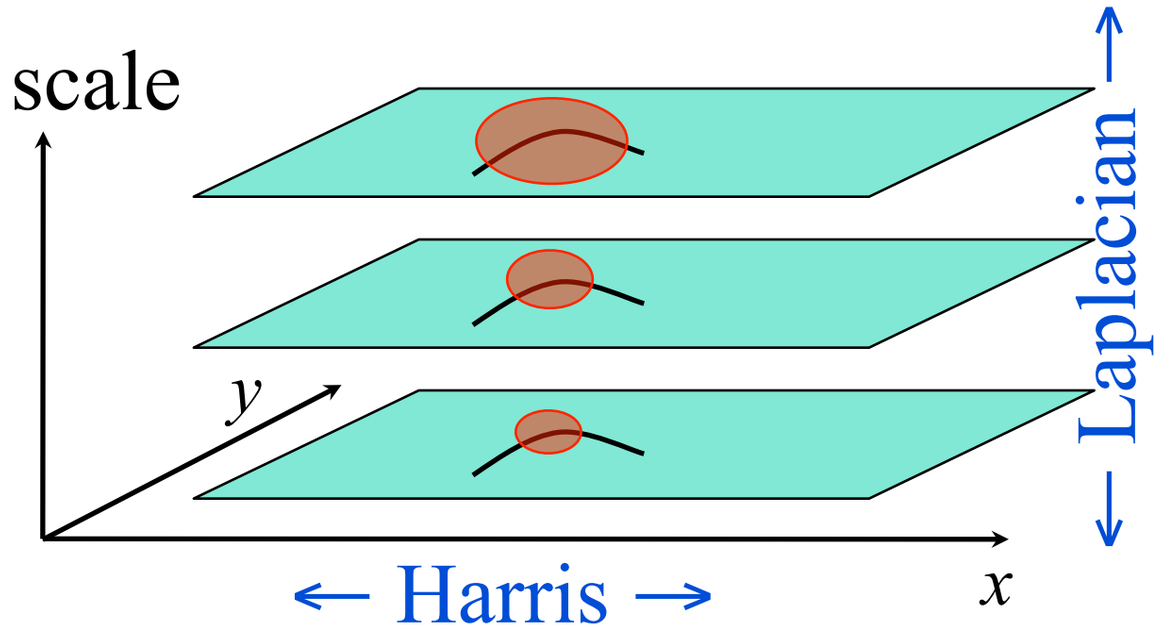
- Harris-Laplacian[1]
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



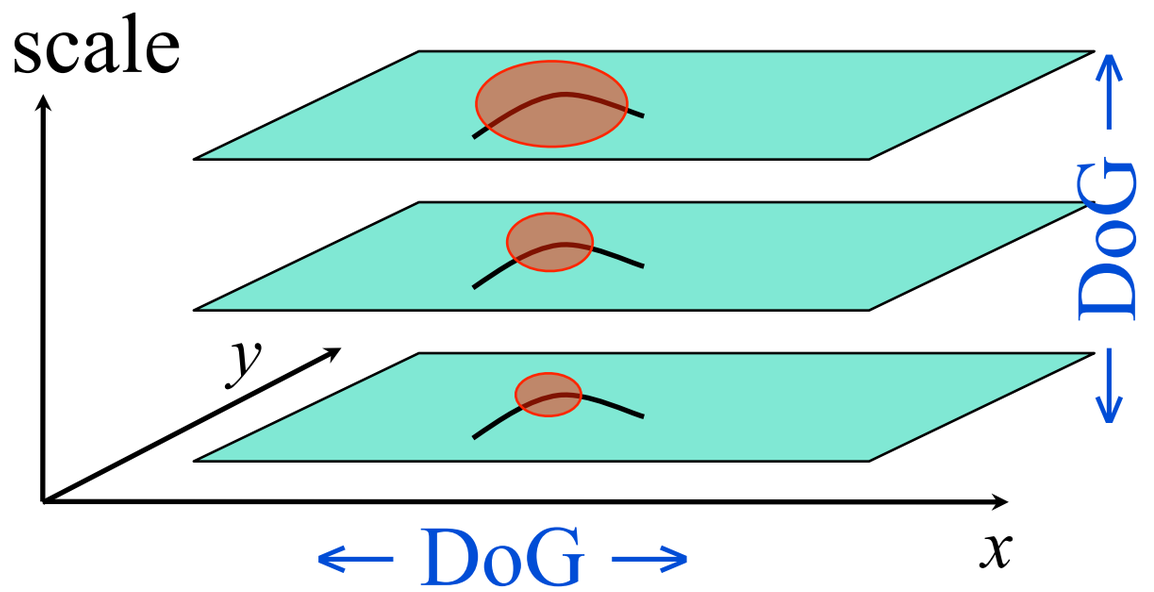
[1] K.Mikolajczyk, C.Schmid. “Indexing Based on Scale Invariant Interest Points”. ICCV 2001

Scale Invariant Detectors Zoo

- Harris-Laplacian[1]
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



- SIFT (Lowe)[2]
  Find local maximum of:
  - Difference of Gaussians in space and scale



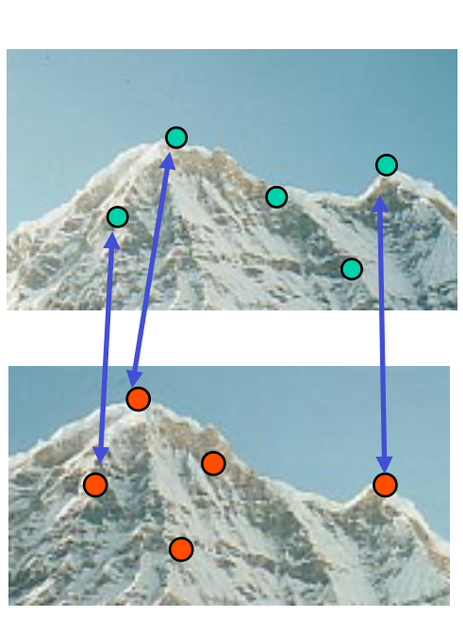
[1] K.Mikolajczyk, C.Schmid. “Indexing Based on Scale Invariant Interest Points”. ICCV 2001
[2] D.Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. Accepted to IJCV 2004

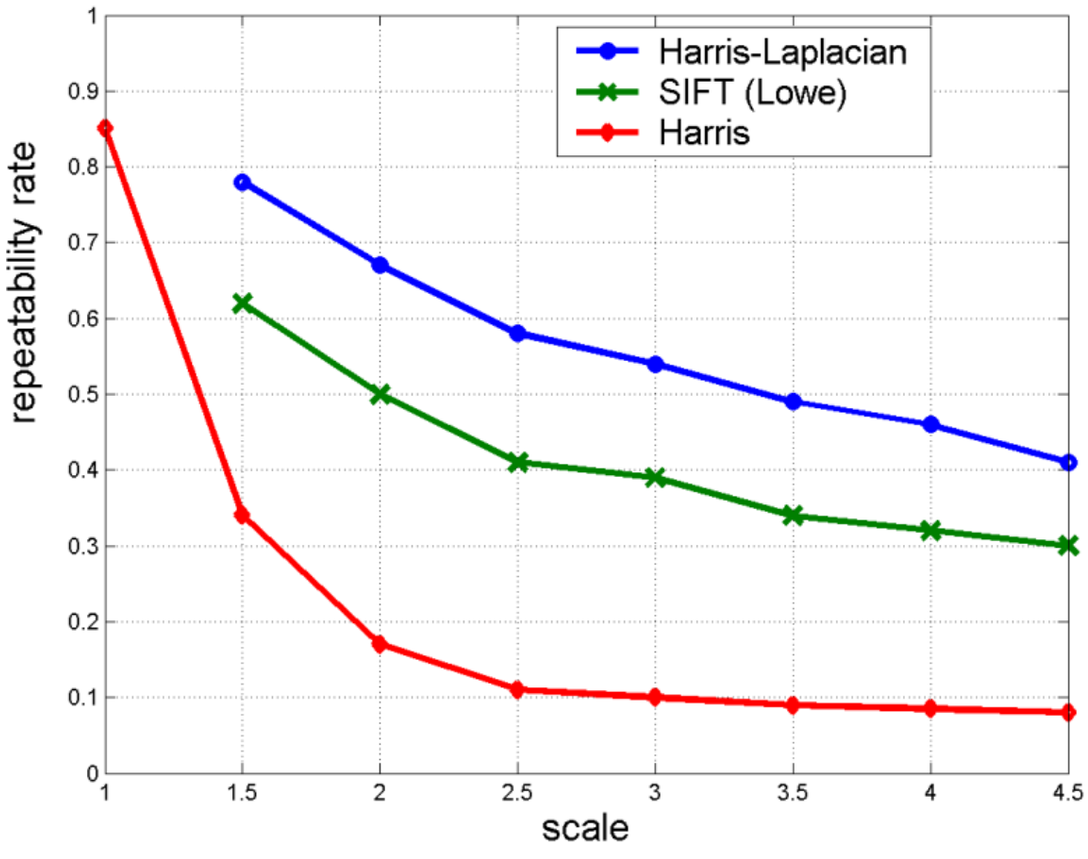
Scale Invariant Detectors

- 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

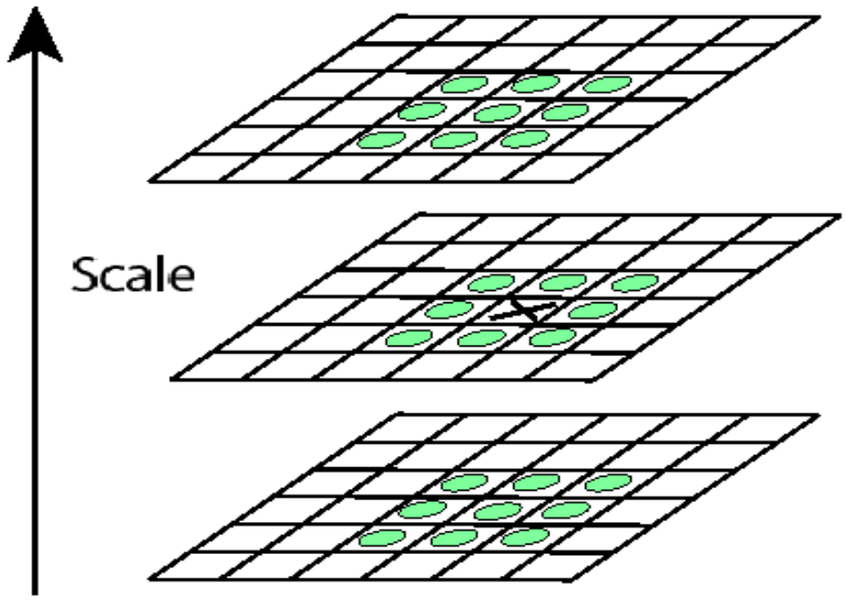
Advanced Key point localization

- Detect maxima and minima of difference-of-Gaussian in scale space
- Fit a quadratic to surrounding values for sub-pixel and sub-scale interpolation (Brown & Lowe, 2002)
- Taylor expansion around point:

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

- Offset of extremum (use finite differences for derivatives):

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



Scale and Rotation Invariant Detection: Summary

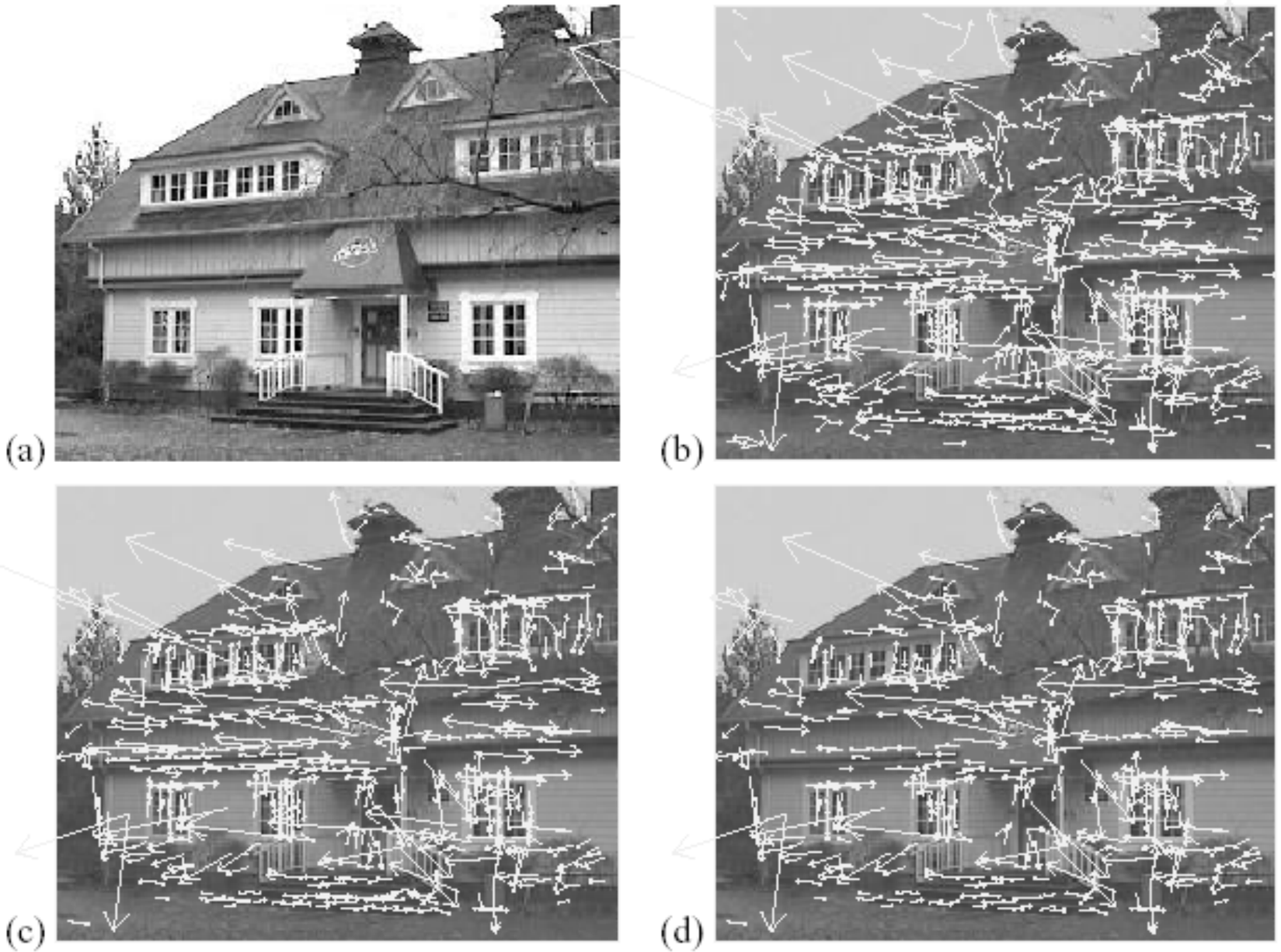
- Given: two images of the same scene with a large *scale difference and/or rotation* between them
- Goal: find *the same* interest points *independently* in each image
- Solution: search for *maxima* of suitable functions in *scale* and in *space* (over the image). Also, find characteristic *orientation*.

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

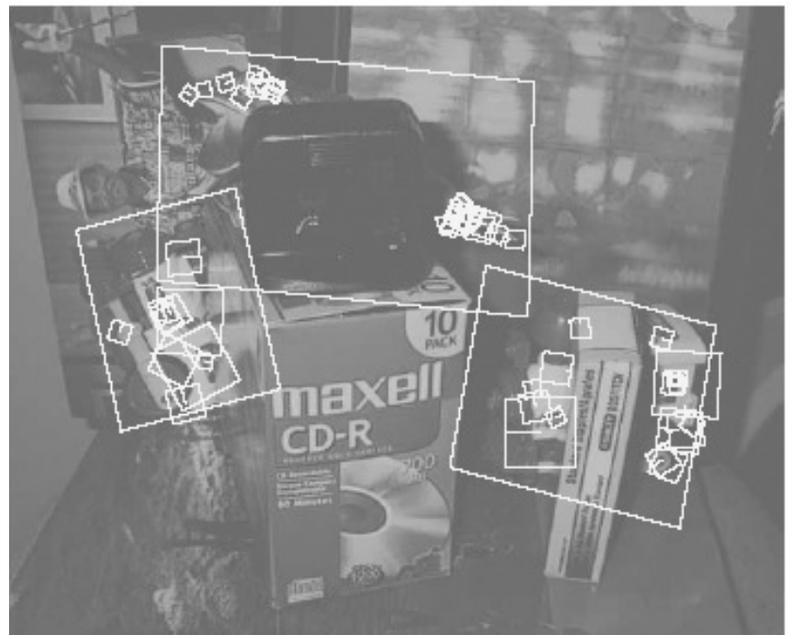
Example of keypoint detection

Threshold on value at DOG peak and on ratio of principle curvatures (Harris approach)



(a) 233x189 image
(b) 832 DOG extrema
(c) 729 left after peak value threshold
(d) 536 left after testing ratio of principle curvatures

Recap



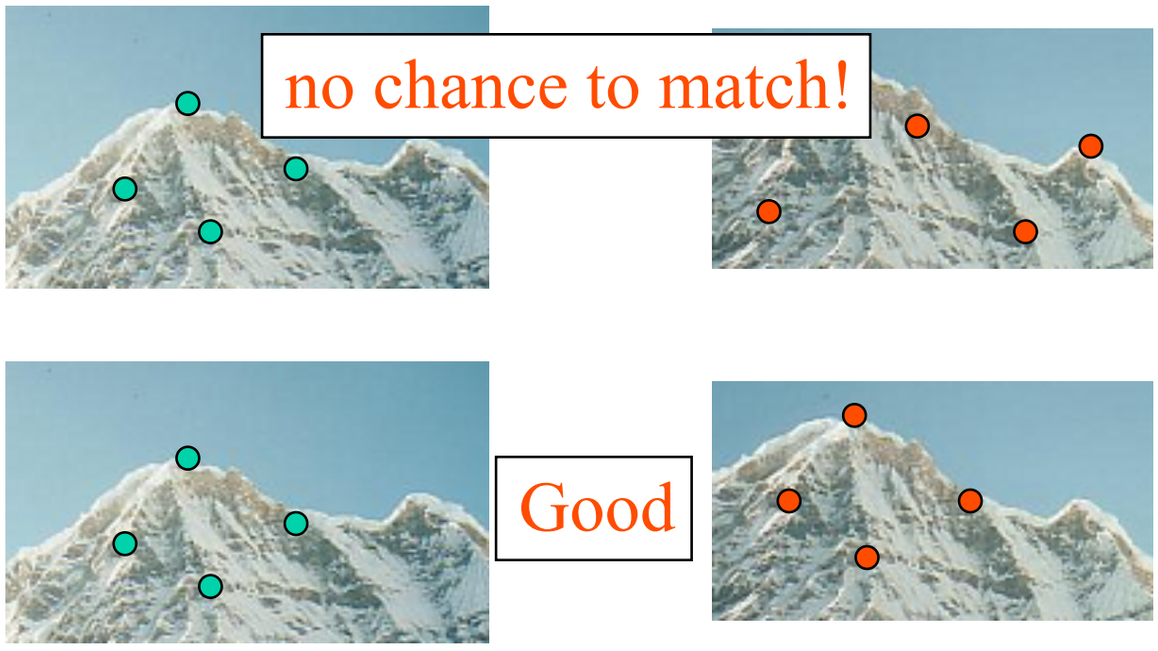
- We have scale & rotation invariant distinctive feature
- We have a characteristic scale
- We have a characteristic orientation

- Now we need to find a **descriptor** of this point
  - For matching with other images

SIFT descriptor

Recall: Matching with Features

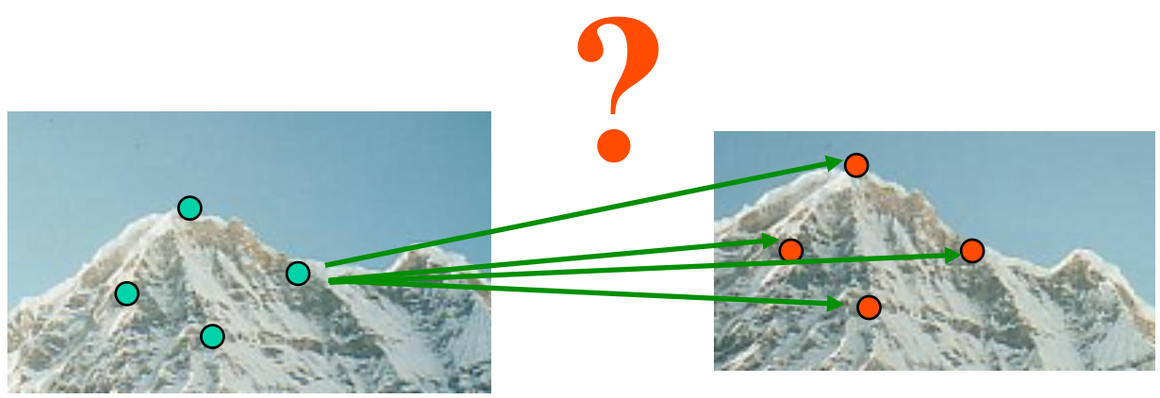
- Problem 1:
  - Detect the *same* point *independently* in both images



We need a repeatable detector:
DONE

Recall: Matching with Features

- Problem 2:
  - For each point correctly recognize the corresponding one



We need a reliable and distinctive descriptor

CVPR 2003 Tutorial

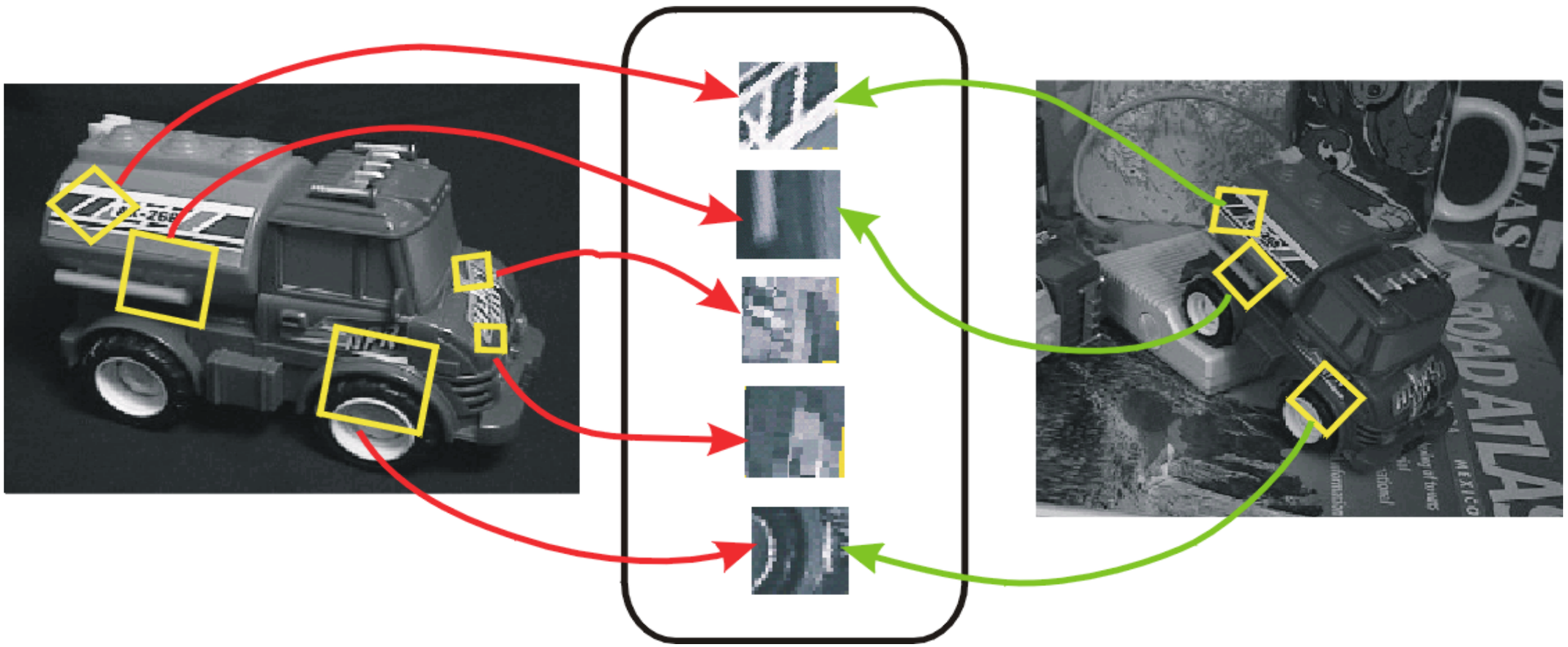
Recognition and Matching Based on Local Invariant Features

David Lowe
Computer Science Department
University of British Columbia

http://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

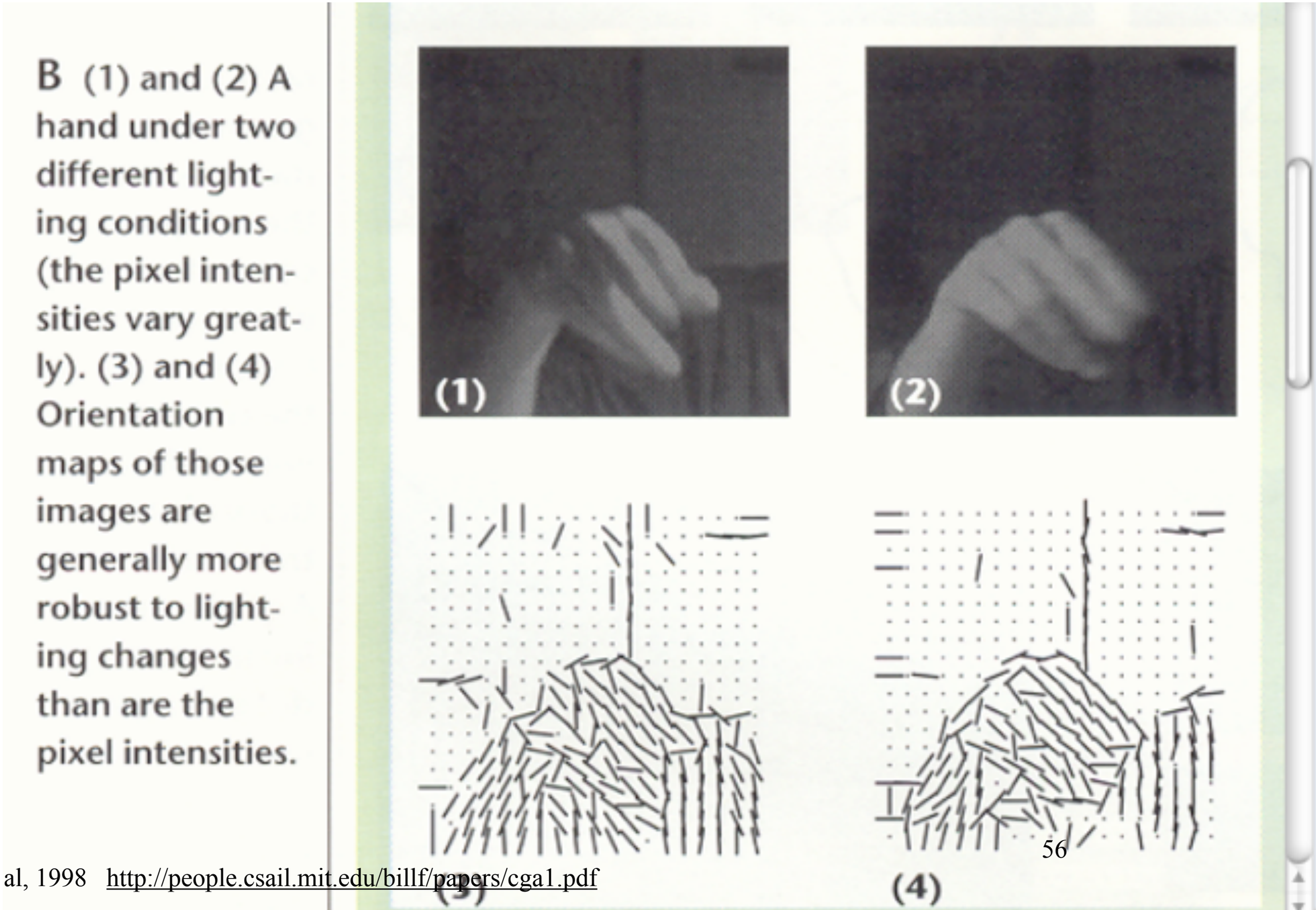
Invariant Local Features

- Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters



SIFT Features

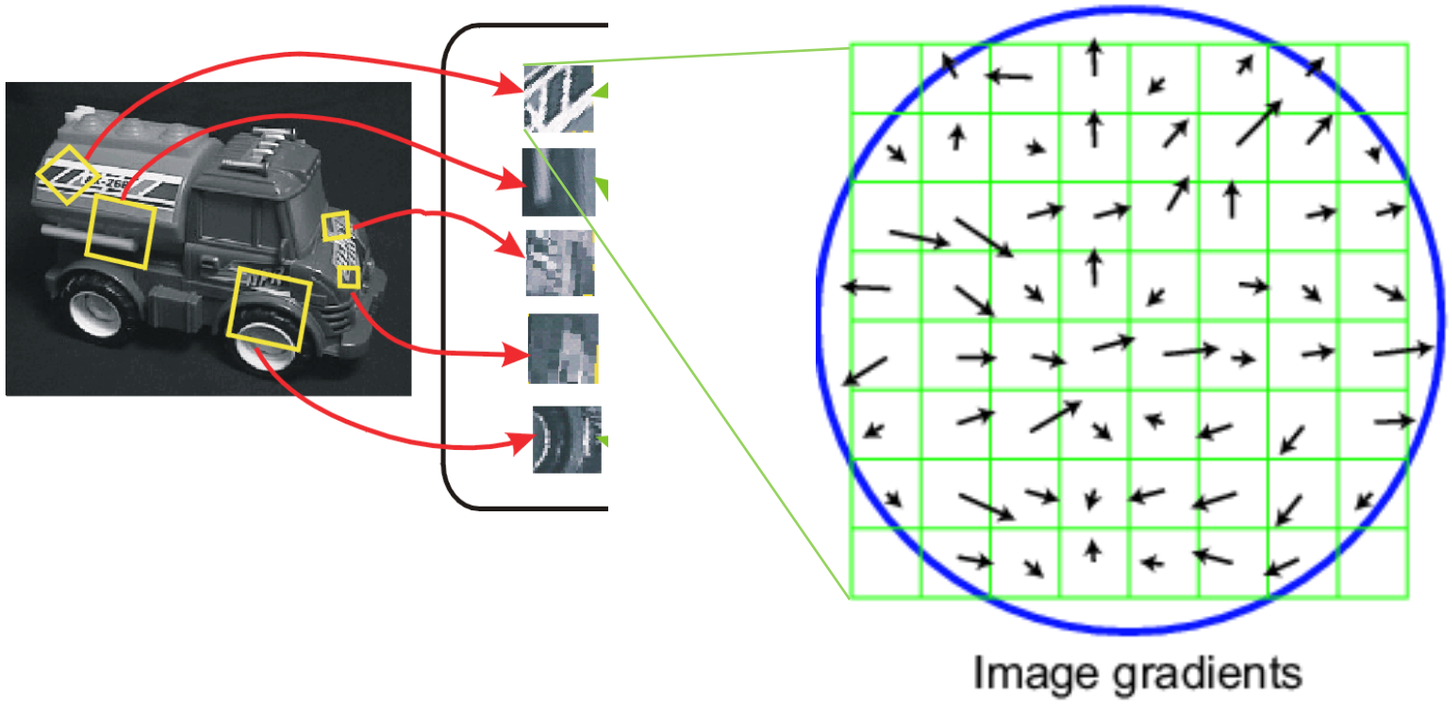
Motivation for using gradients: gradient orientations somewhat invariant to lighting variations



Freeman et al, 1998 http://people.csail.mit.edu/billf/papers/cga1.pdf

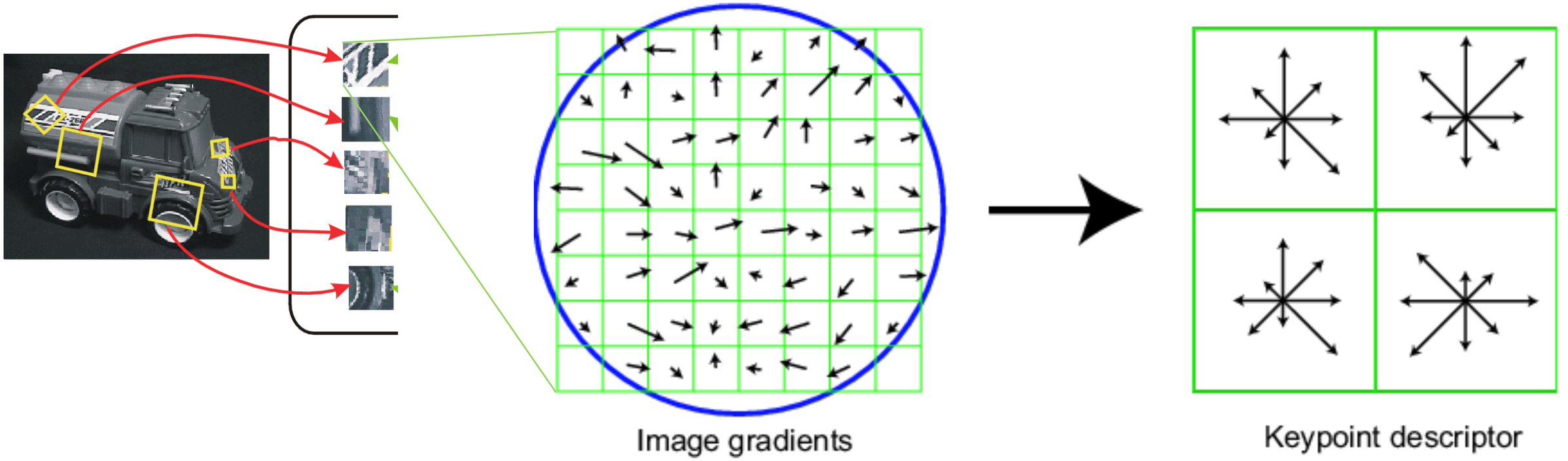
SIFT vector formation

- Computed on rotated and scaled version of window according to computed orientation & scale
  - –resample a 16x16 version of the window
- Based on gradients weighted by a Gaussian of variance half the window (for smooth falloff)



SIFT vector formation

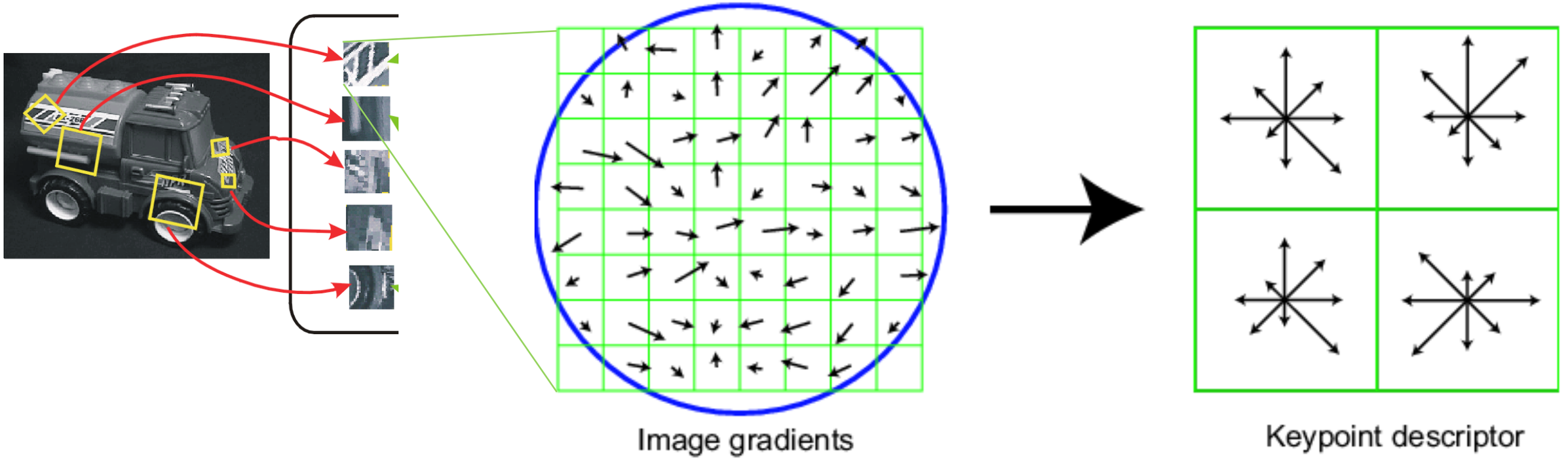
- 4x4 array of gradient orientation histograms
  - –not really histogram, weighted by magnitude
- 8 orientations x 4x4 array = 128 dimensions
- Motivation: some sensitivity to spatial layout, but not too much.



showing only 2x2 here but is 4x4

Reduce effect of illumination

- 128-dim vector normalized to 1
- Threshold gradient magnitudes to avoid excessive influence of high gradients
  - after normalization, clamp gradients >0.2
  - renormalize



Sensitivity to number of histogram orientations

ijcv04-1.pdf (page 17 of 28)
Previous Next Zoom Move Text Select Sidebar Search

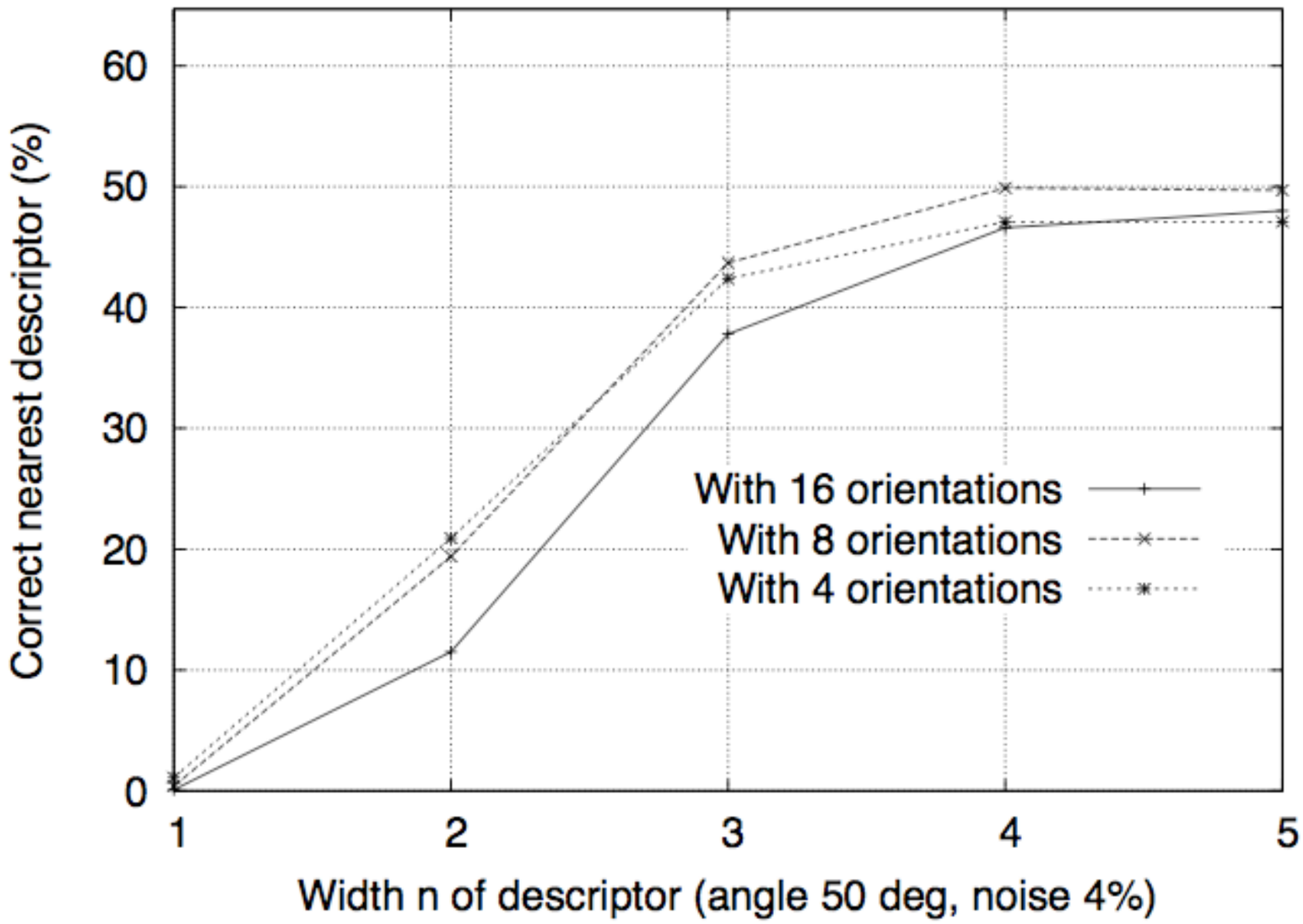
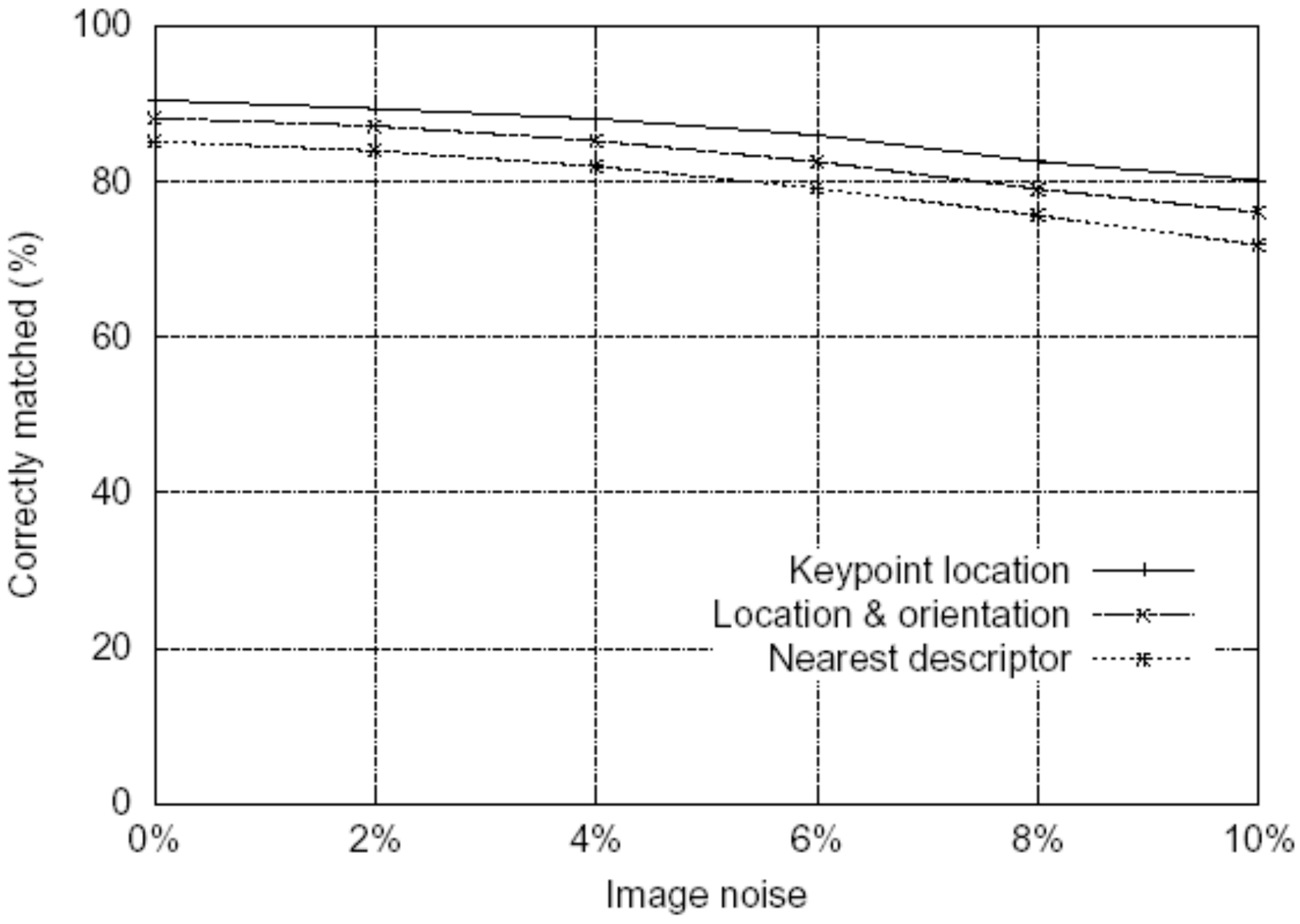


Figure 8: This graph shows the percent of keypoints giving the correct match to a database of 40,000 keypoints as a function of width of the $n \times n$ keypoint descriptor and the number of orientations in each histogram. The graph is computed for images with affine viewpoint change of 50 degrees and addition of 4% noise.

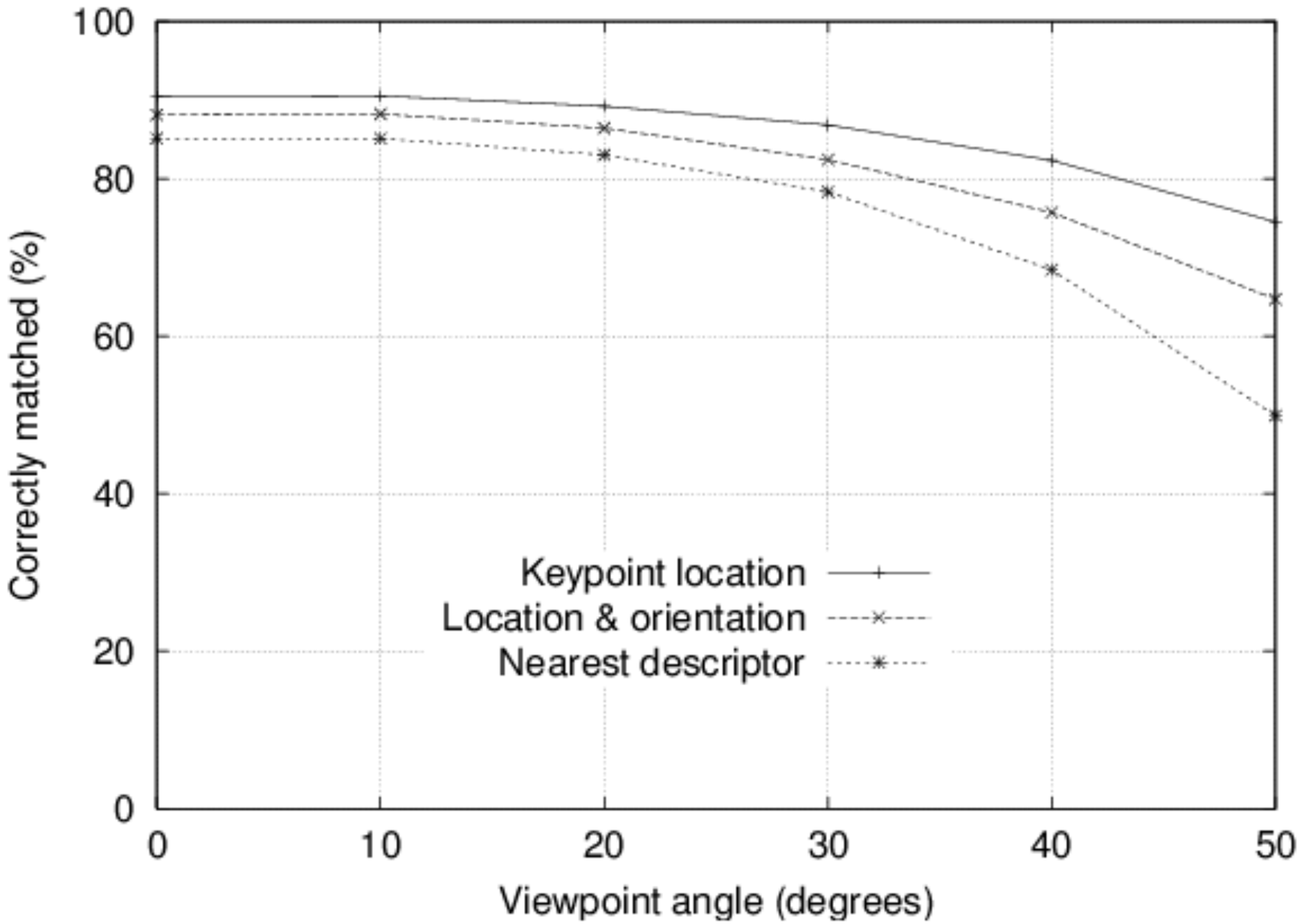
Feature stability to noise

- Match features after random change in image scale & orientation, with differing levels of image noise
- Find nearest neighbor in database of 30,000 features



Feature stability to affine change

- Match features after random change in image scale & orientation, with 2% image noise, and affine distortion
- Find nearest neighbor in database of 30,000 features



Distinctiveness of features

- Vary size of database of features, with 30 degree affine change, 2% image noise
- Measure % correct for single nearest neighbor match

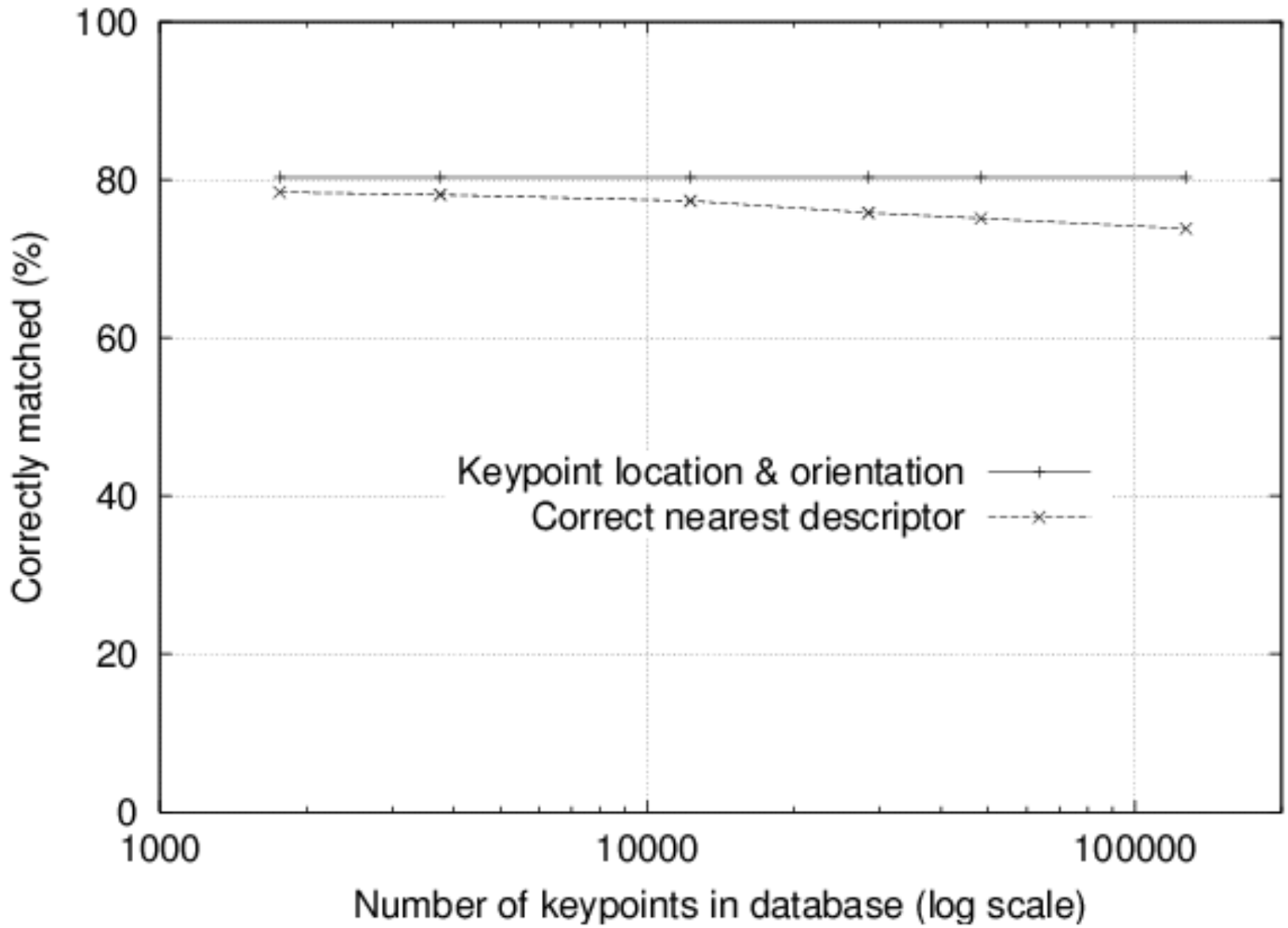




Figure 12: The training images for two objects are shown on the left. These can be recognized in a cluttered image with extensive occlusion, shown in the middle. The results of recognition are shown on the right. A parallelogram is drawn around each recognized object showing the boundaries of the original training image under the affi ne transformation solved for during recognition. Smaller squares indicate the keypoints that were used for recognition.

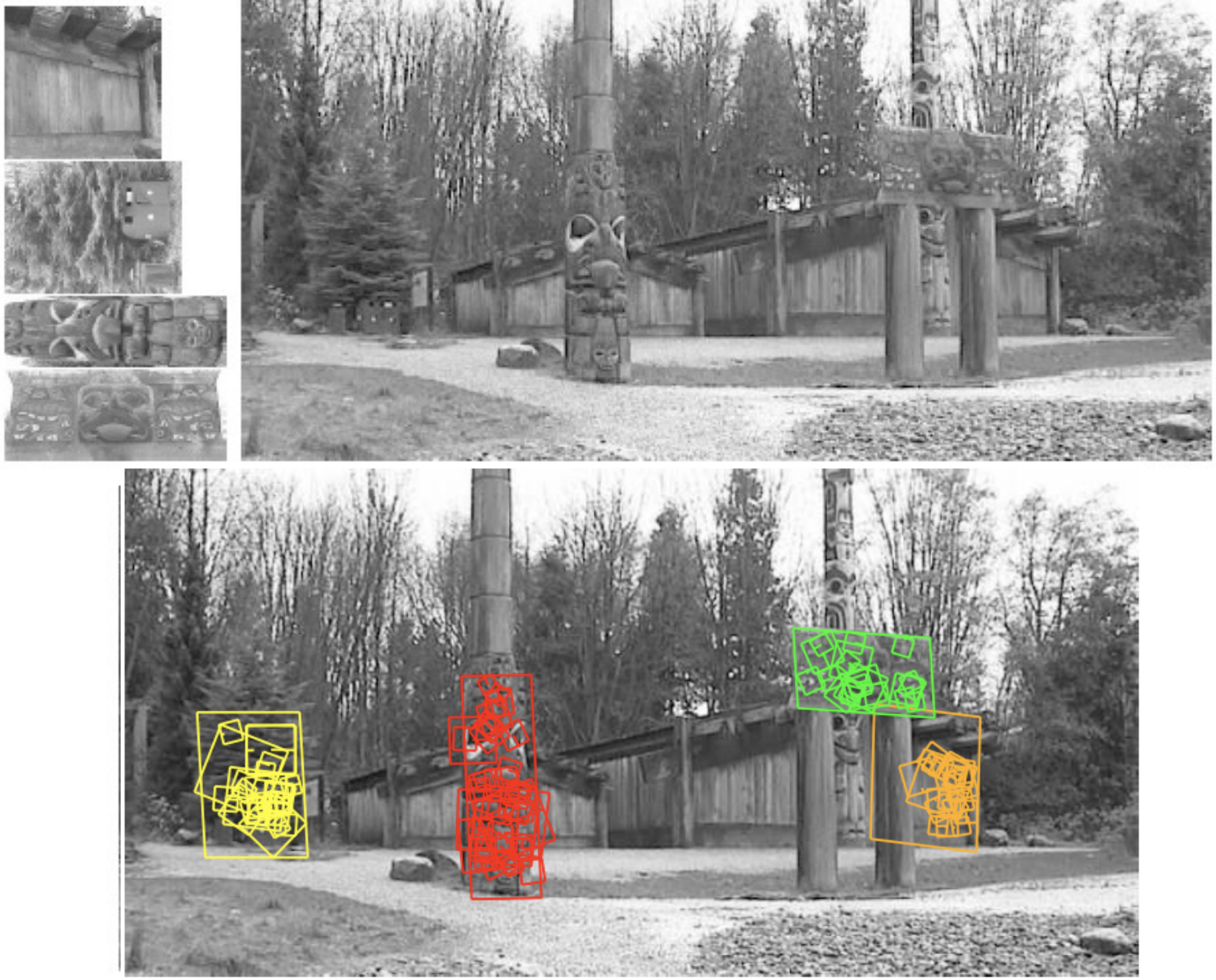


Figure 13: This example shows location recognition within a complex scene. The training images for locations are shown at the upper left and the 640x315 pixel test image taken from a different viewpoint is on the upper right. The recognized regions are shown on the lower image, with keypoints shown as squares and an outer parallelogram showing the boundaries of the training images under the affi ne transform used for recognition.

A good SIFT features tutorial

http://www.cs.toronto.edu/~jepson/csc2503/tutSIFT04.pdf

By Estrada, Jepson, and Fleet.

The original paper:

http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.2.8899

Recap

- Stable (repeatable) feature points can be detected regardless of image changes
  - Scale: search for correct scale as *maximum* of appropriate function
- Invariant and distinctive descriptors can be computed

Autopano demo

- Uses sift

3D version:

Photo tourism and Photosynth

Photo Tourism:
Exploring Photo Collections in 3D

Noah Snavely

Steven M. Seitz

University of Washington

Richard Szeliski

Microsoft Research

© 2006 Noah Snavely



15,464

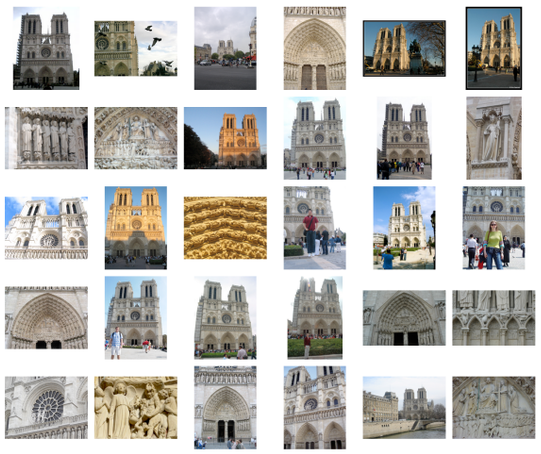


37,383



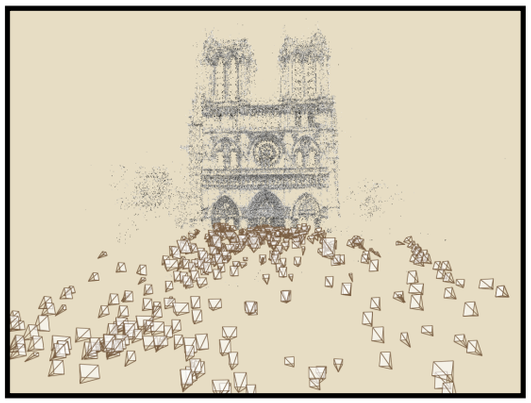
76,389

Photo Tourism overview



Input photographs

Scene reconstruction



Relative camera positions and orientations

Point cloud

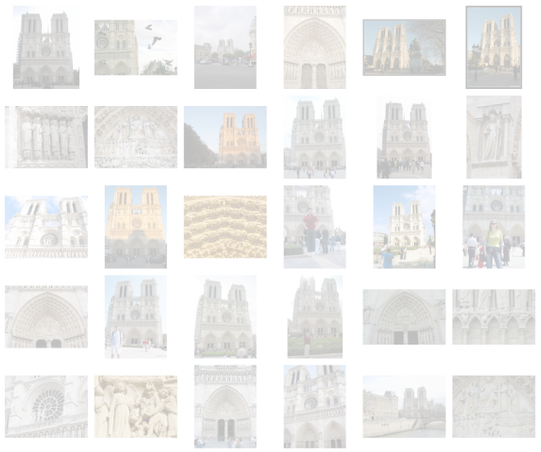
Sparse correspondence



Photo Explorer

© 2006 Noah Snavely

Photo Tourism overview



Input photographs

Scene reconstruction

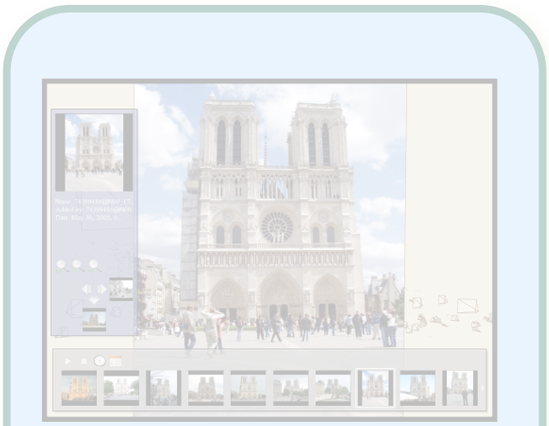
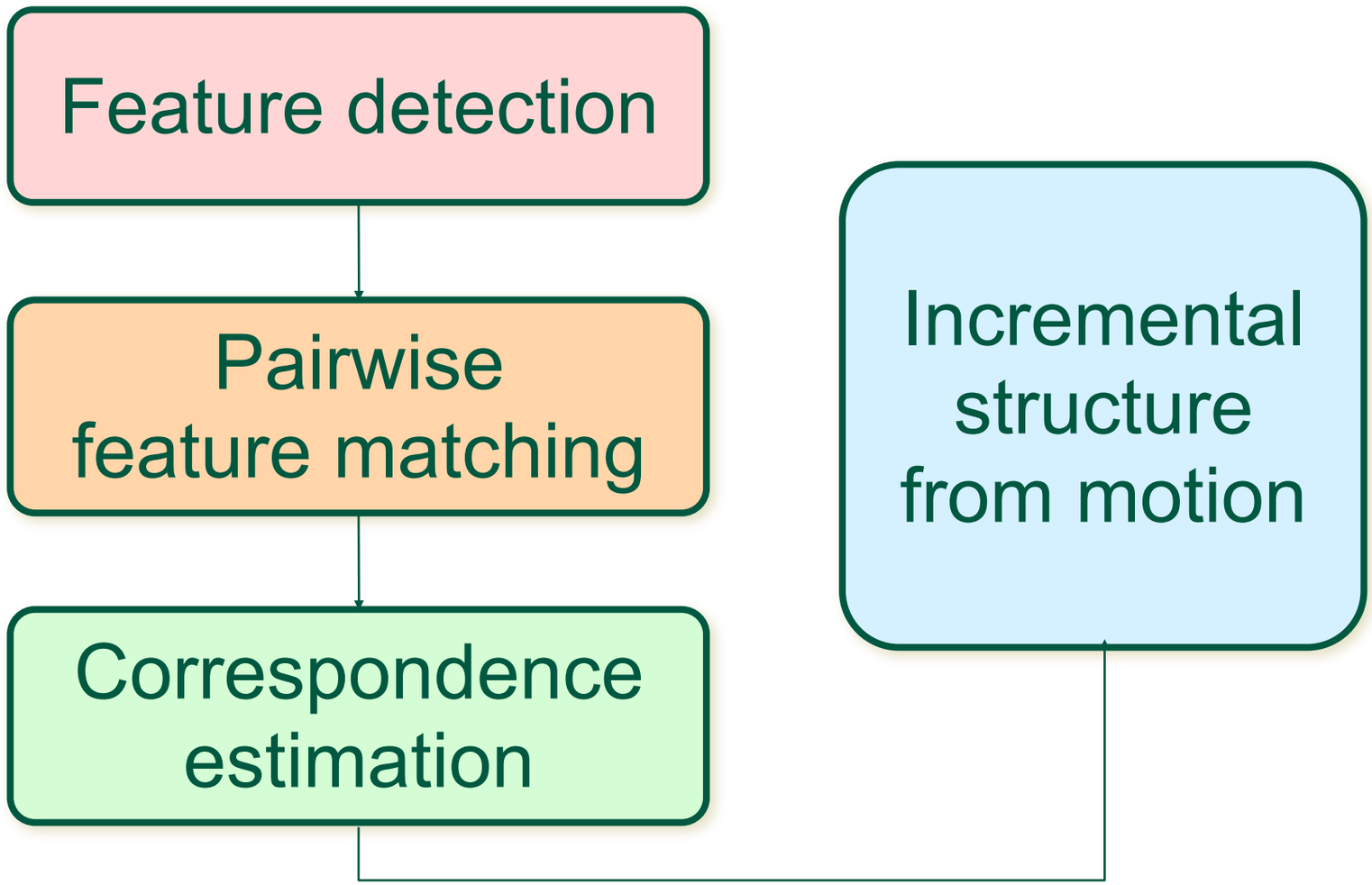


Photo Explorer

© 2006 Noah Snavely

Scene reconstruction

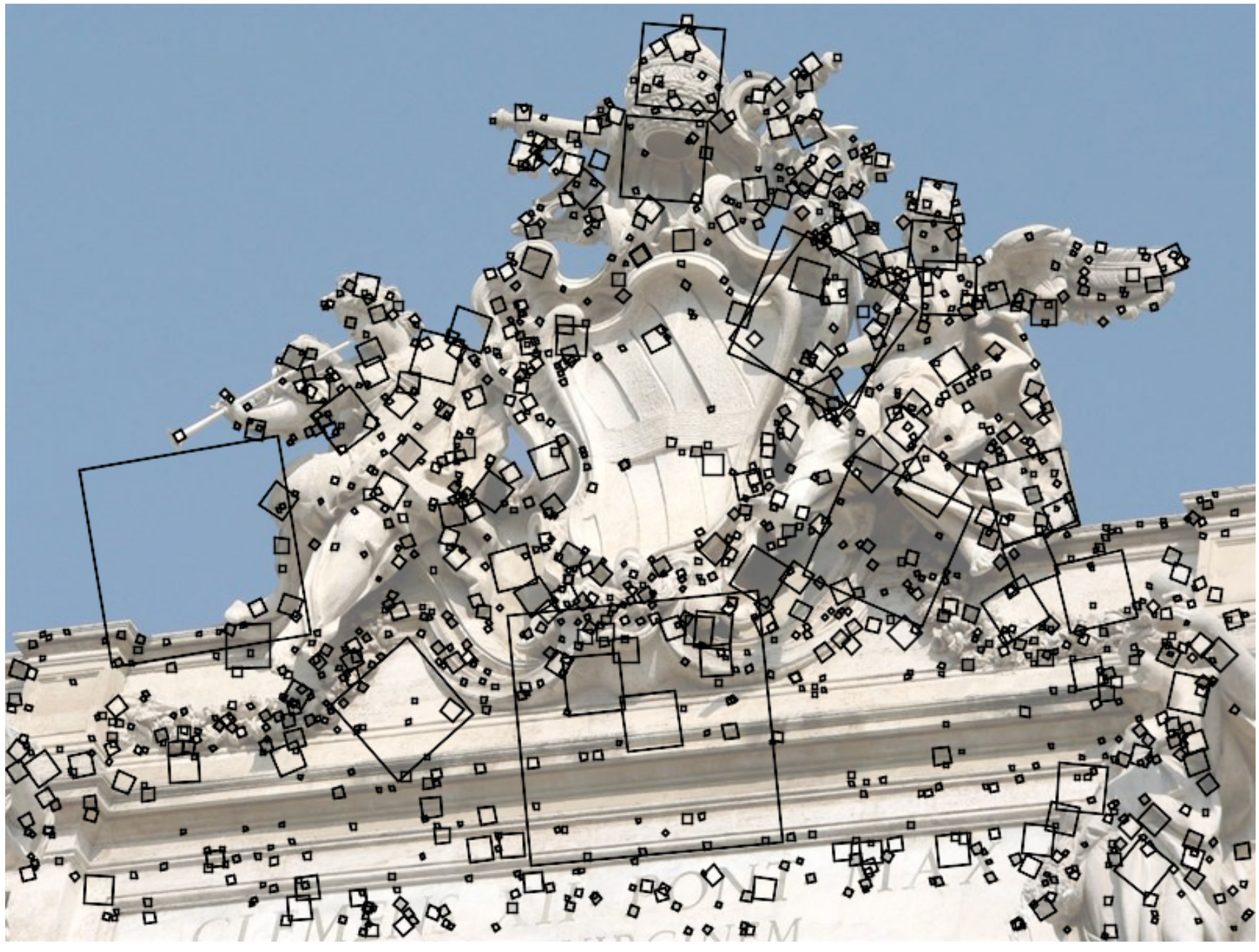
- Automatically estimate
  - position, orientation, and focal length of cameras
  - 3D positions of feature points



© 2006 Noah Snavely

Feature detection

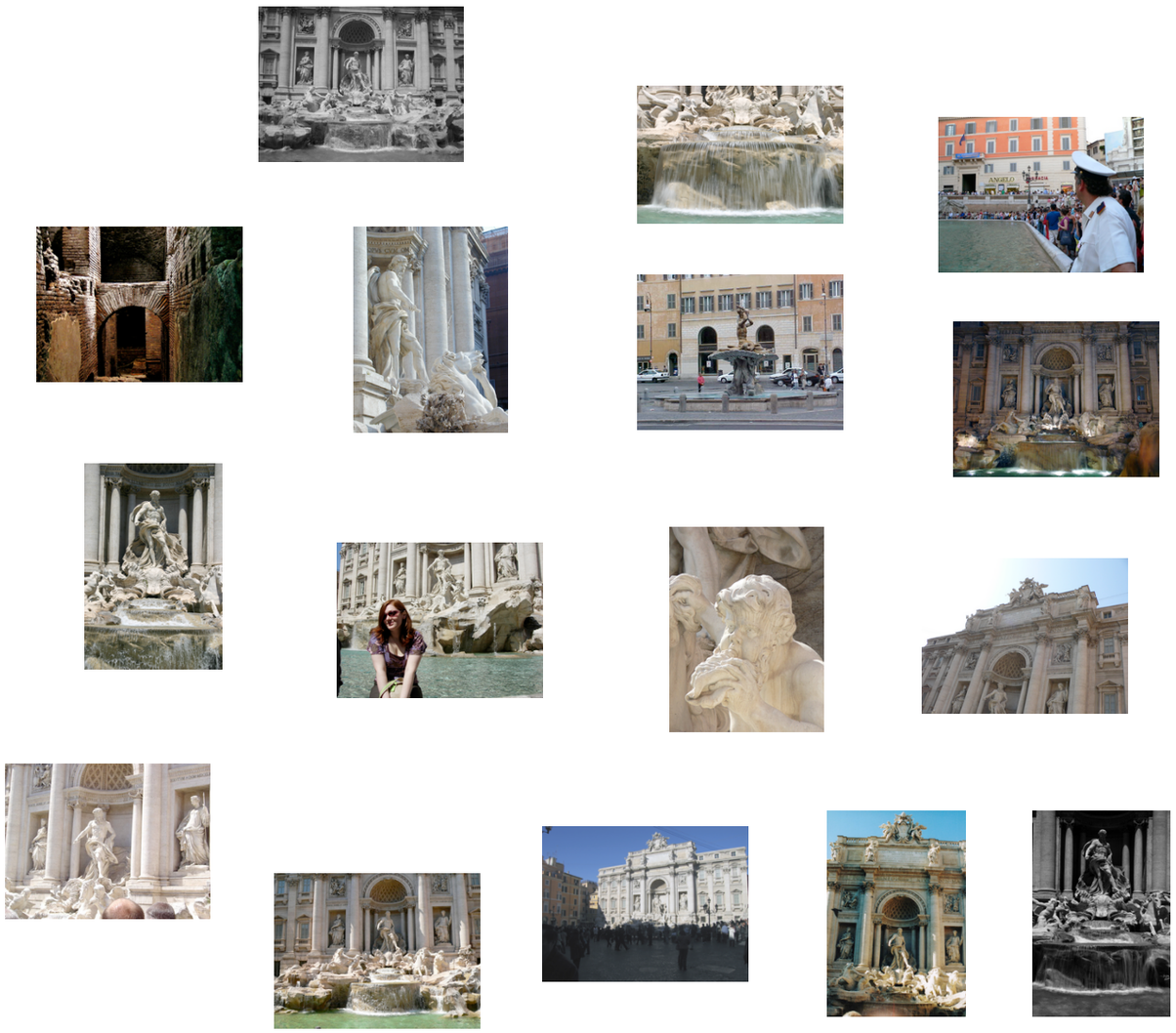
Detect features using SIFT [Lowe, IJCV 2004]



© 2006 Noah Snavely

Feature detection

Detect features using SIFT [Lowe, IJCV 2004]



© 2006 Noah Snavely

Feature detection

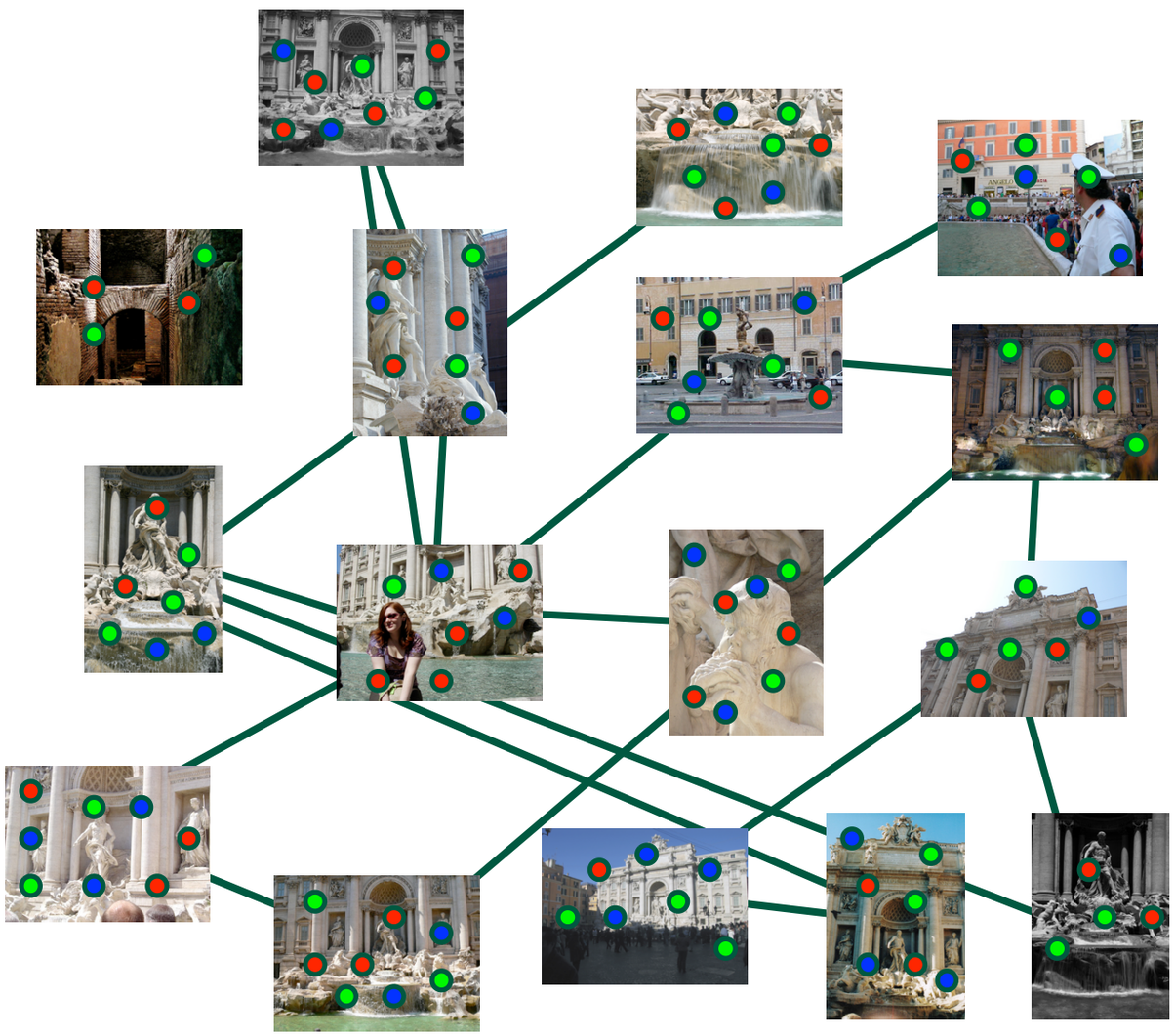
Detect features using SIFT [Lowe, IJCV 2004]



© 2006 Noah Snavely

Feature matching

Match features between each pair of images

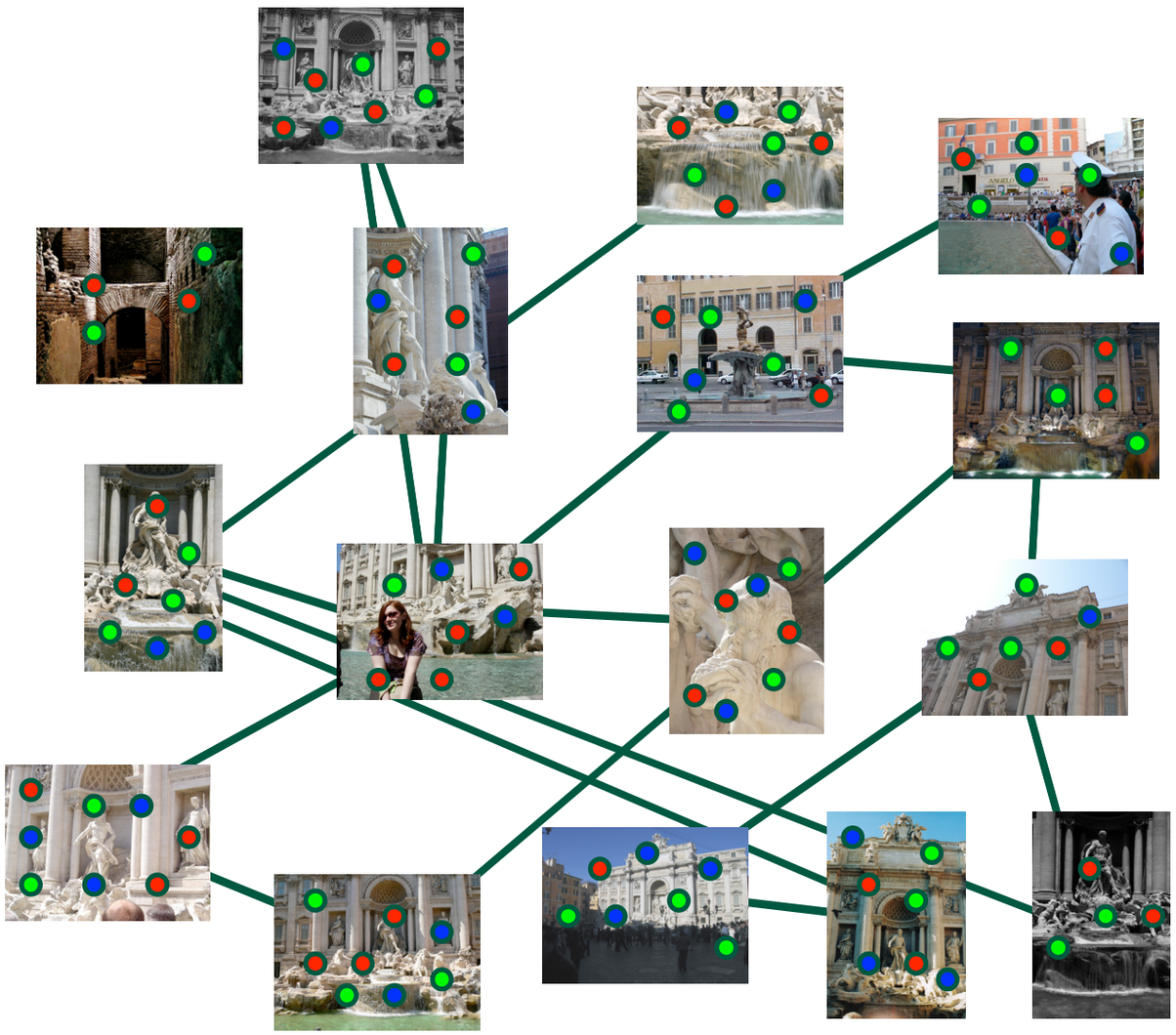


© 2006 Noah Snavely

Feature matching

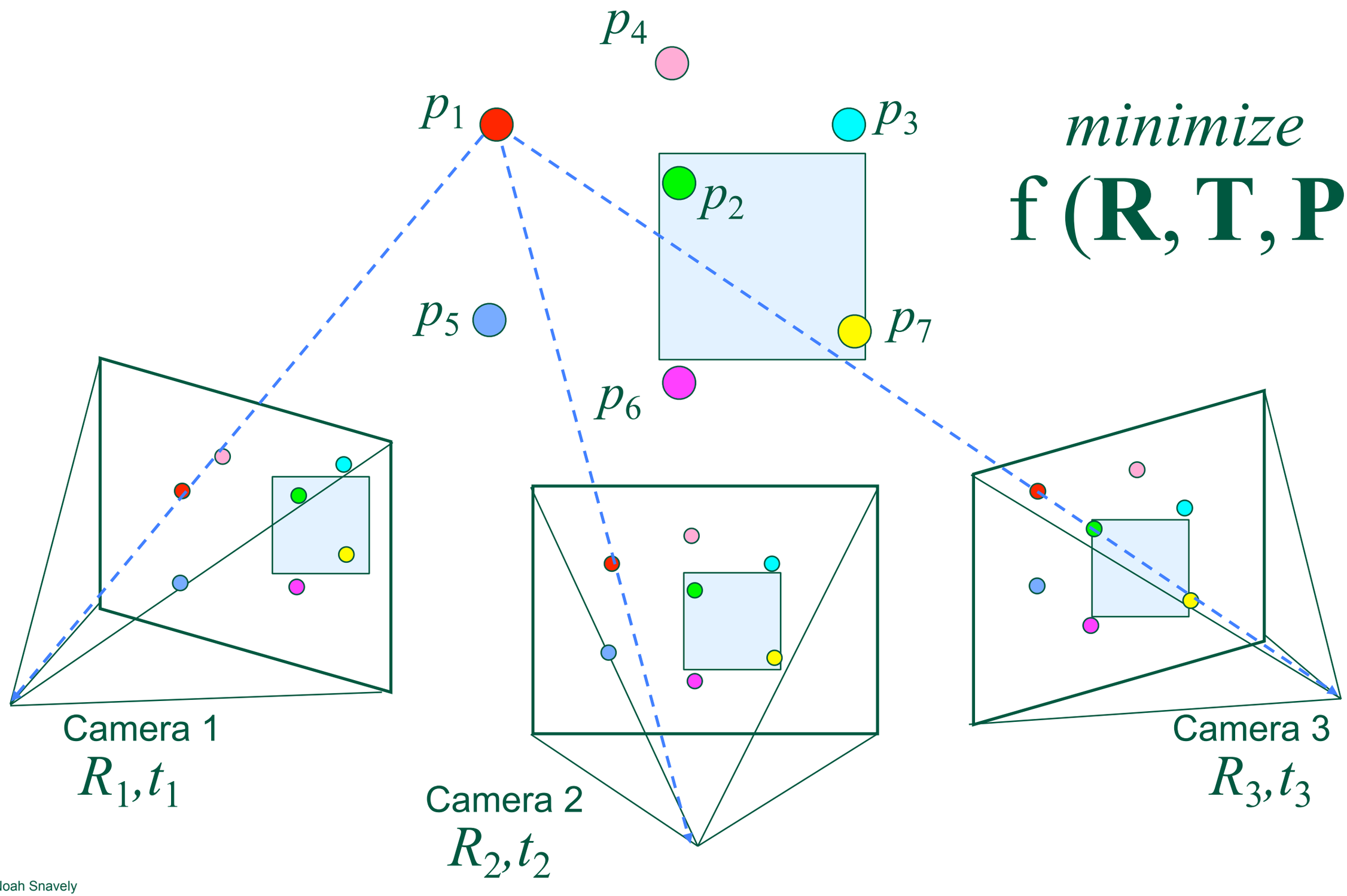
Refine matching using RANSAC [Fischler & Bolles 1987]
to estimate fundamental matrices between pairs

(See 6.801/6.866 for fundamental matrix, or Hartley and Zisserman, Multi-View Geometry)



© 2006 Noah Snavely

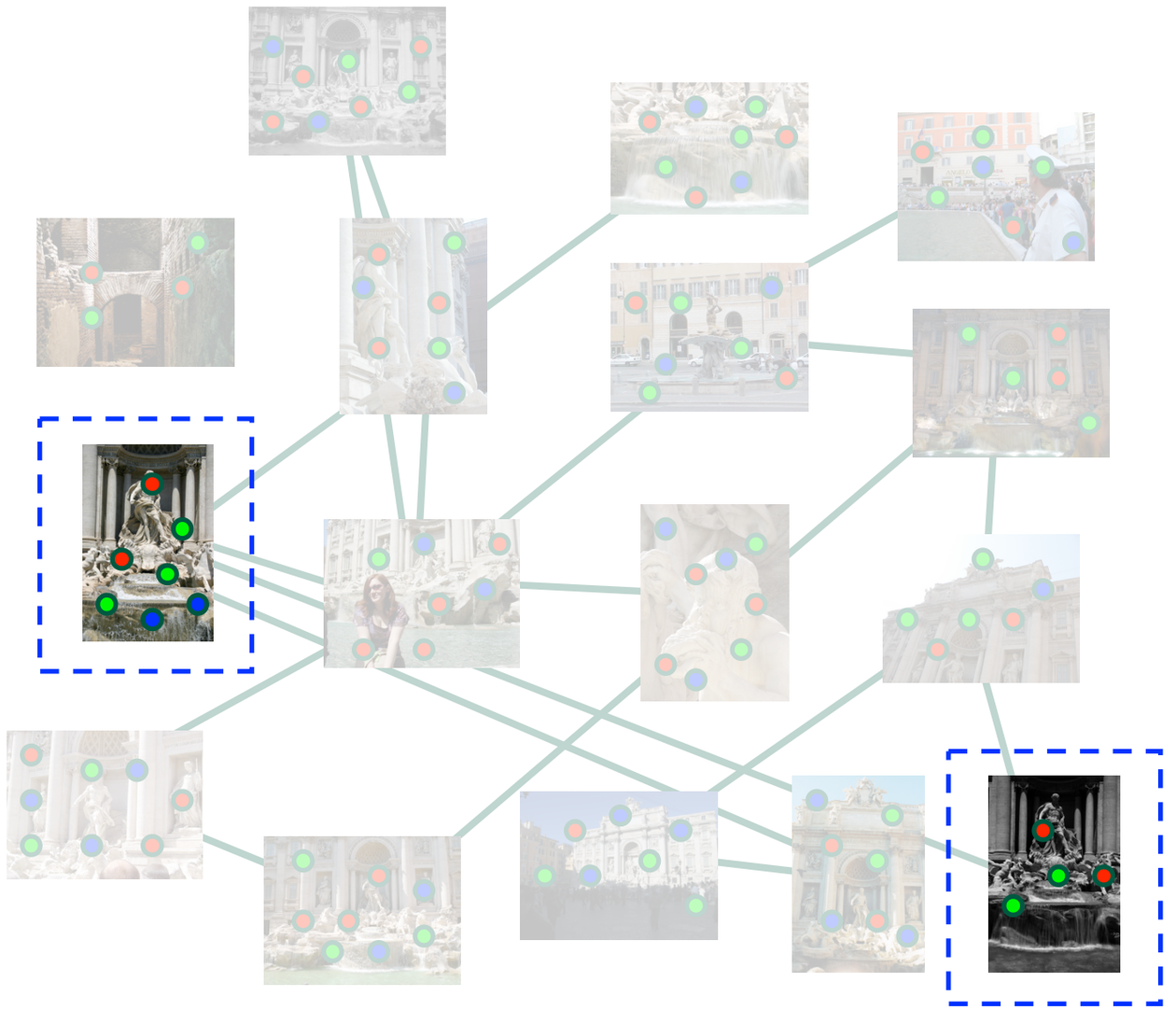
Structure from motion



$$\textit{minimize}\ \mathbf{f}(\mathbf{R}, \mathbf{T}, \mathbf{P})$$

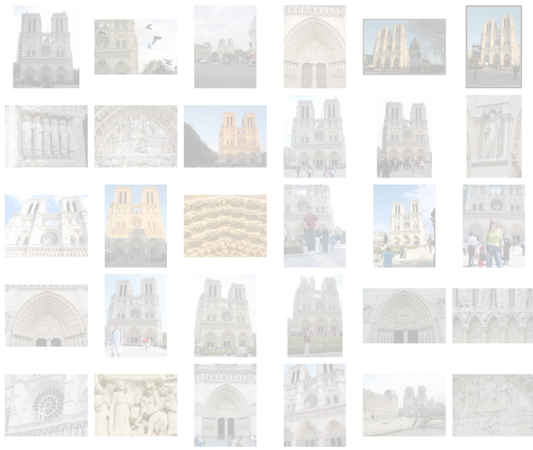
© 2006 Noah Snavely

Incremental structure from motion



© 2006 Noah Snavely

Photo Tourism overview



Input photographs

Scene reconstruction



Photo Explorer

- Navigation
- Rendering
- Annotations

© 2006 Noah Snavely

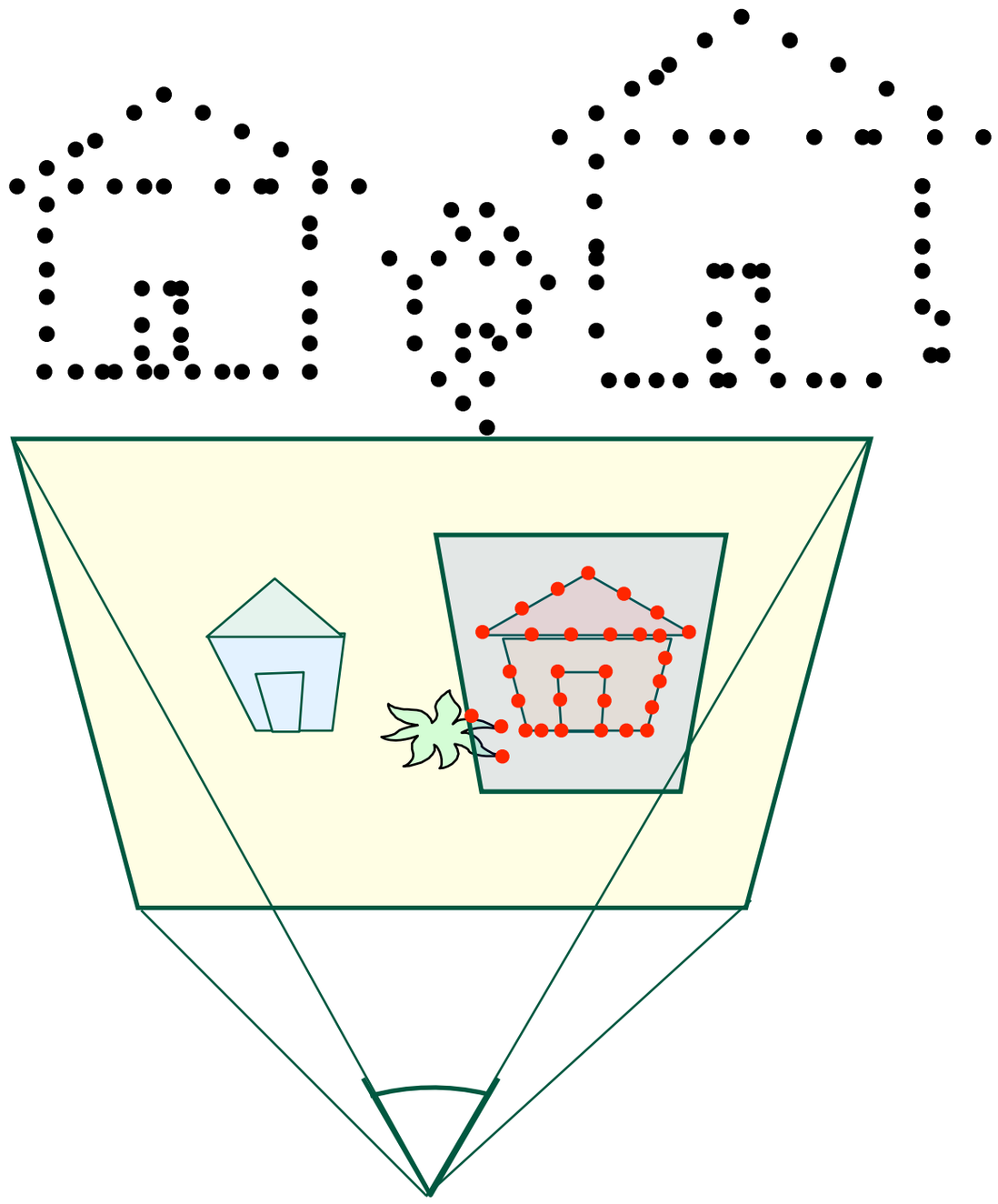
Object-based browsing



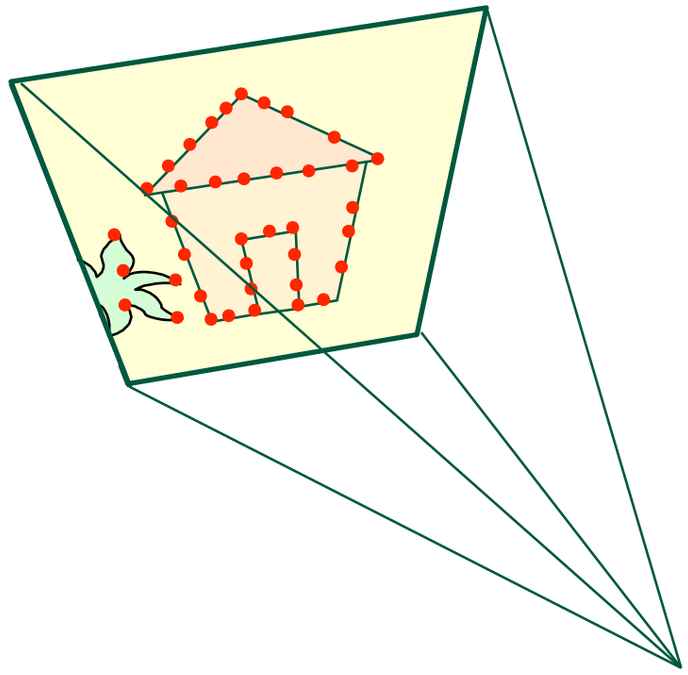


© 2006 Noah Snavely

Object-based browsing

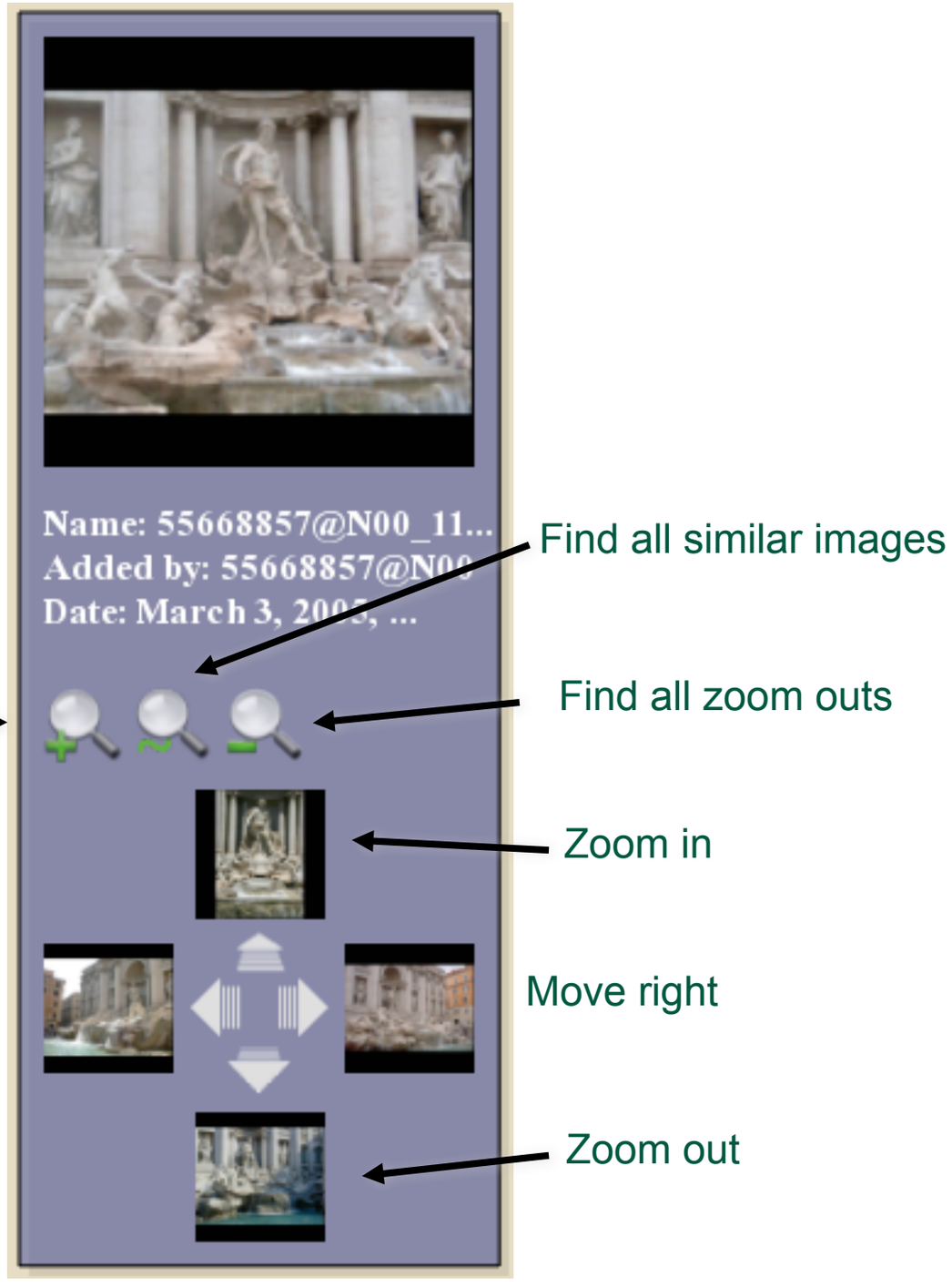


- Visibility
- Resolution
- Head-on view



© 2006 Noah Snavely

Relation-based browsing



© 2006 Noah Snavely

Relation-based browsing

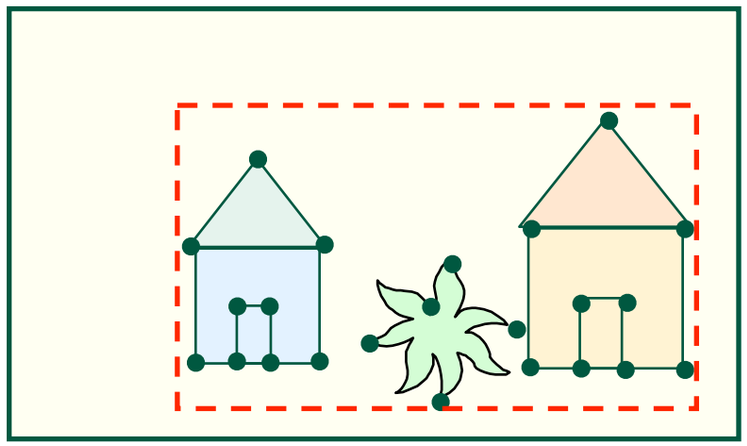


Image A

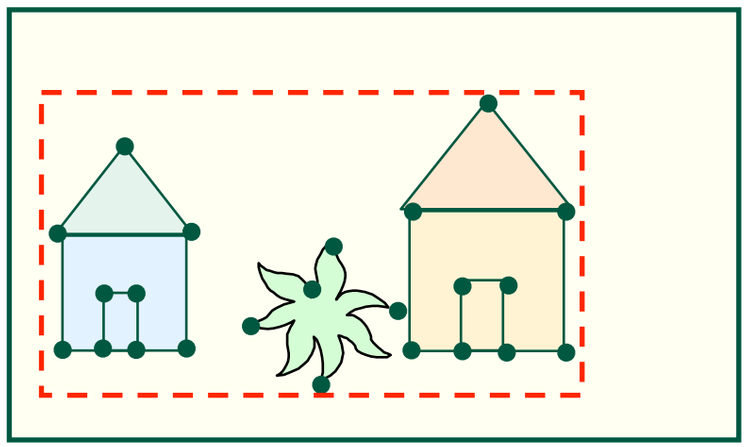


Image B

© 2006 Noah Snavely

Relation-based browsing

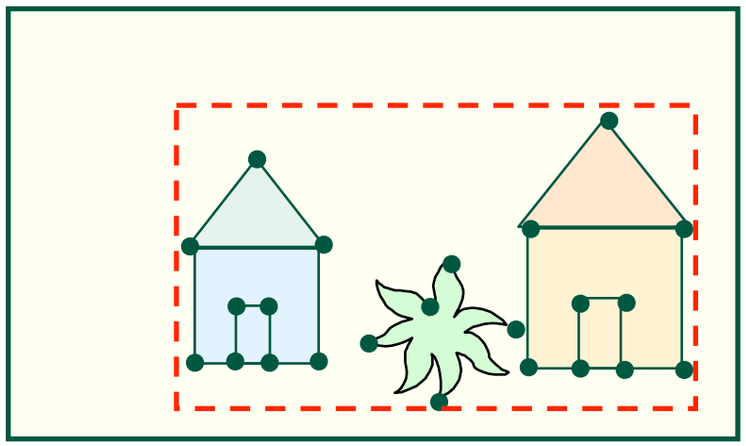


Image A

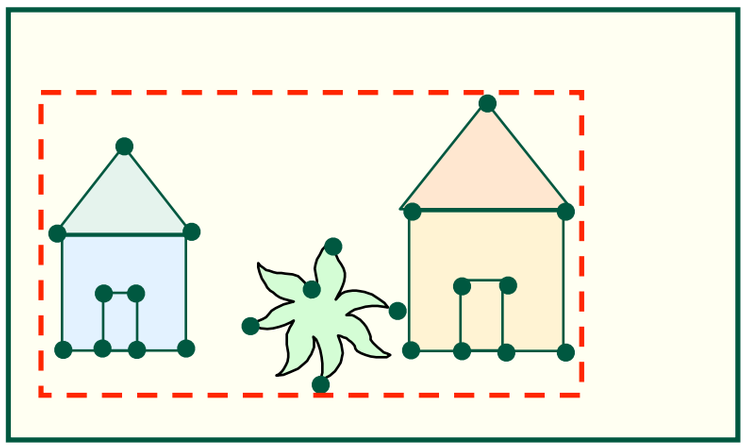
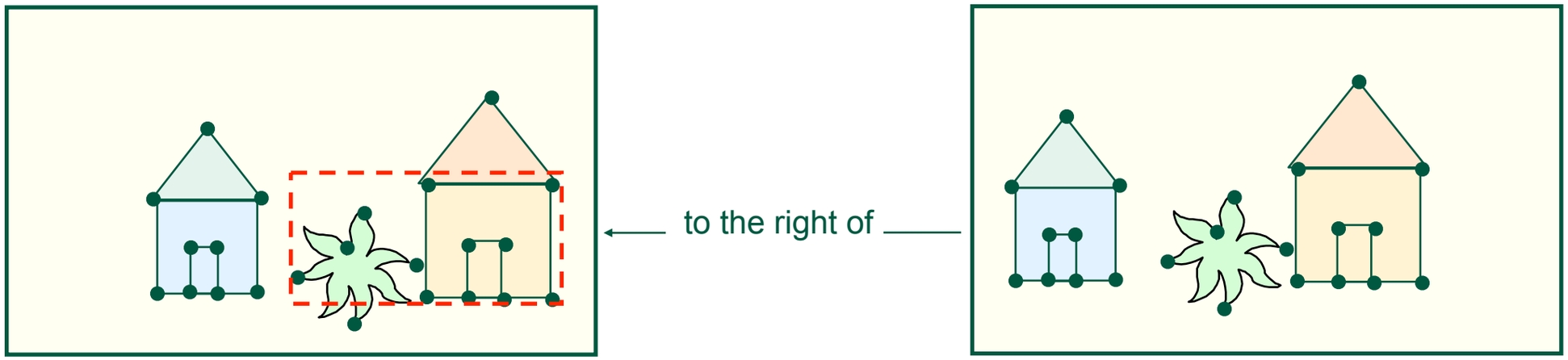
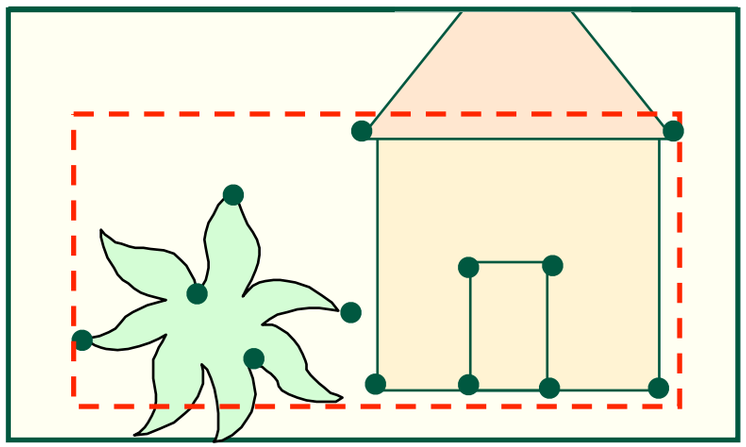


Image B

© 2006 Noah Snavely

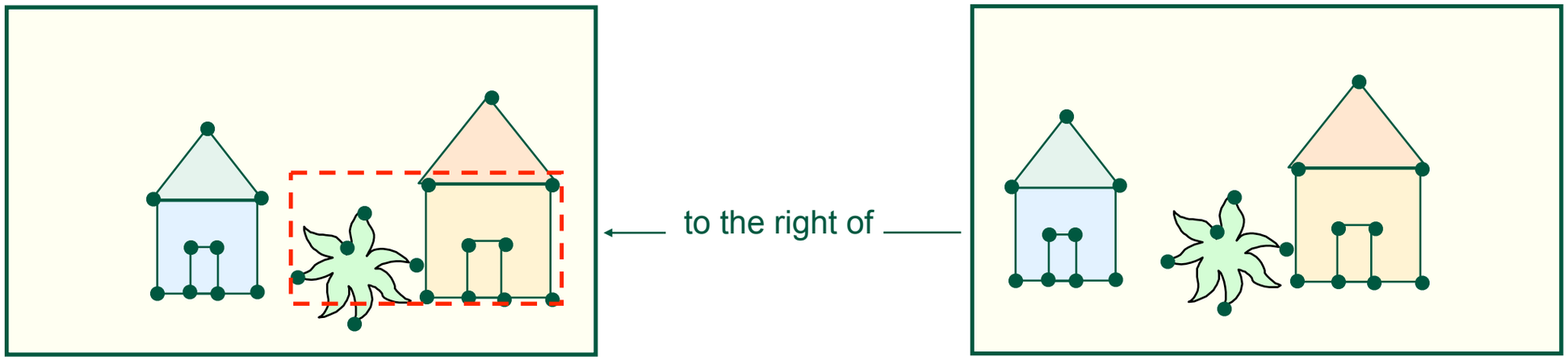
Relation-based browsing

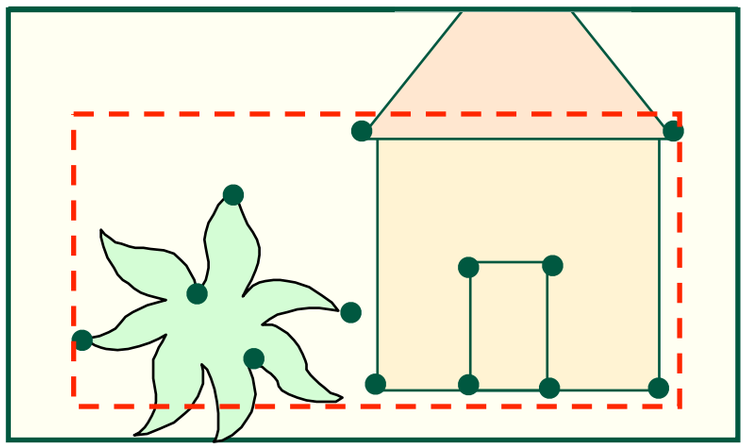




© 2006 Noah Snavely

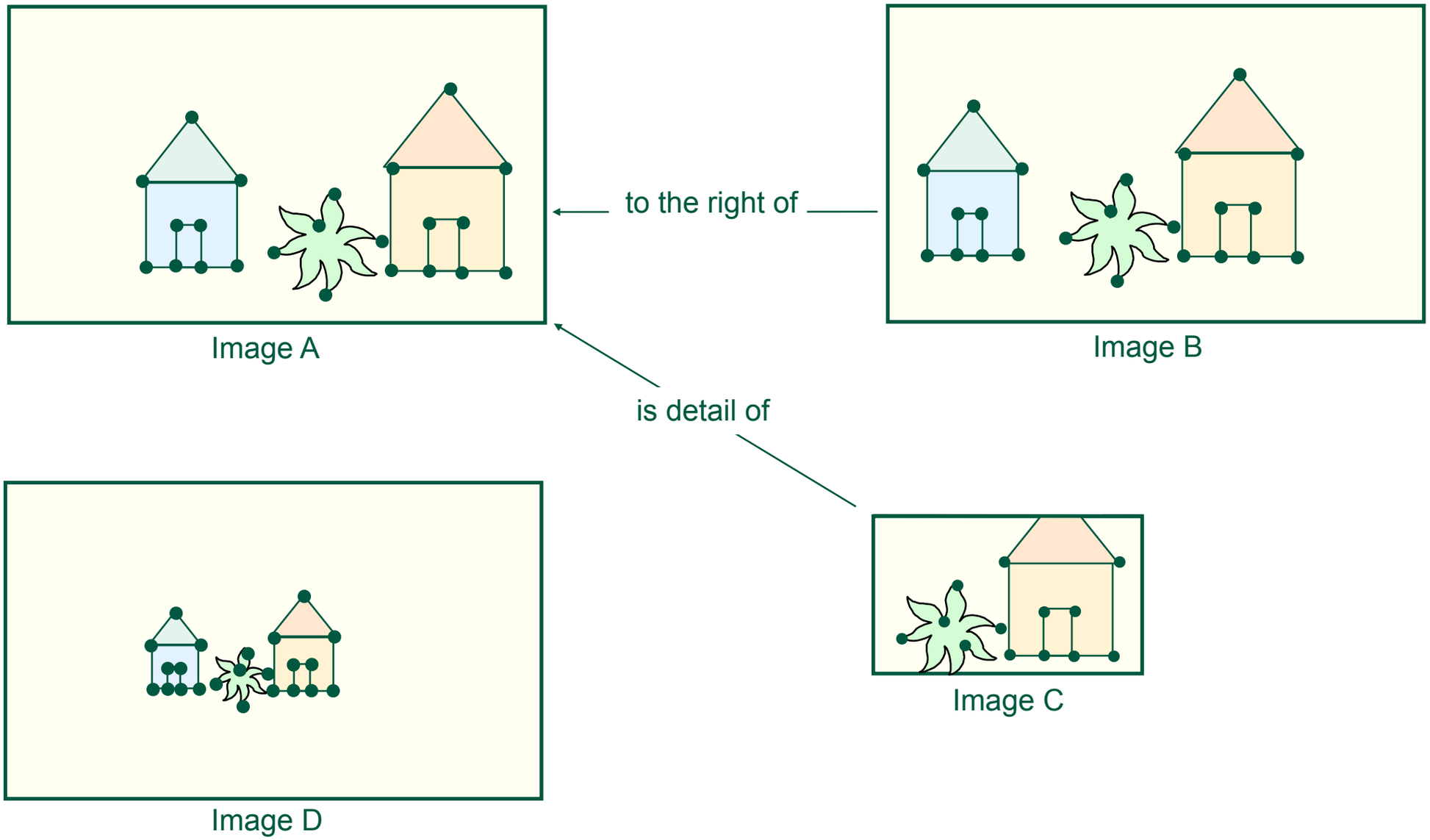
Relation-based browsing





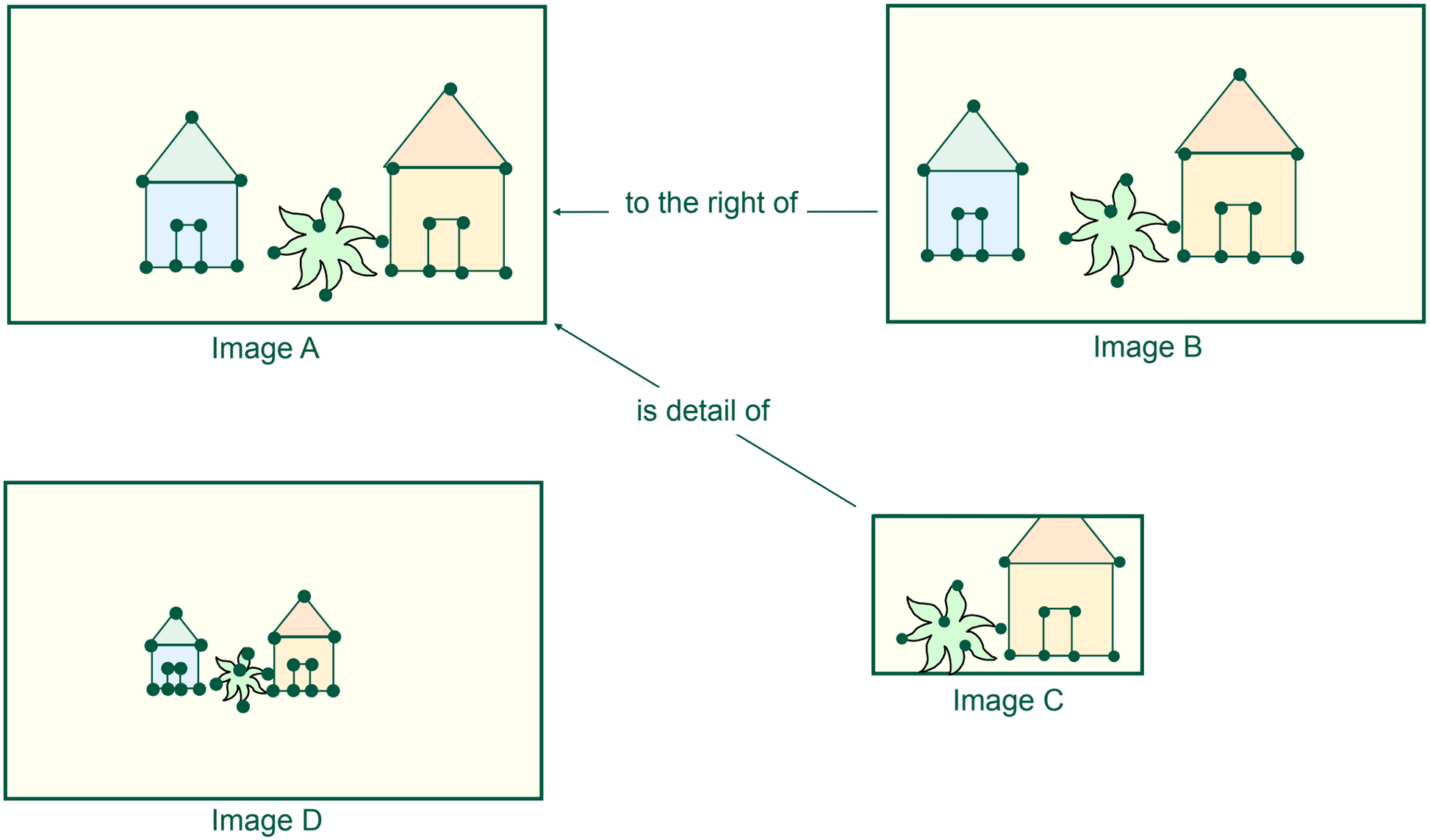
© 2006 Noah Snavely

Relation-based browsing



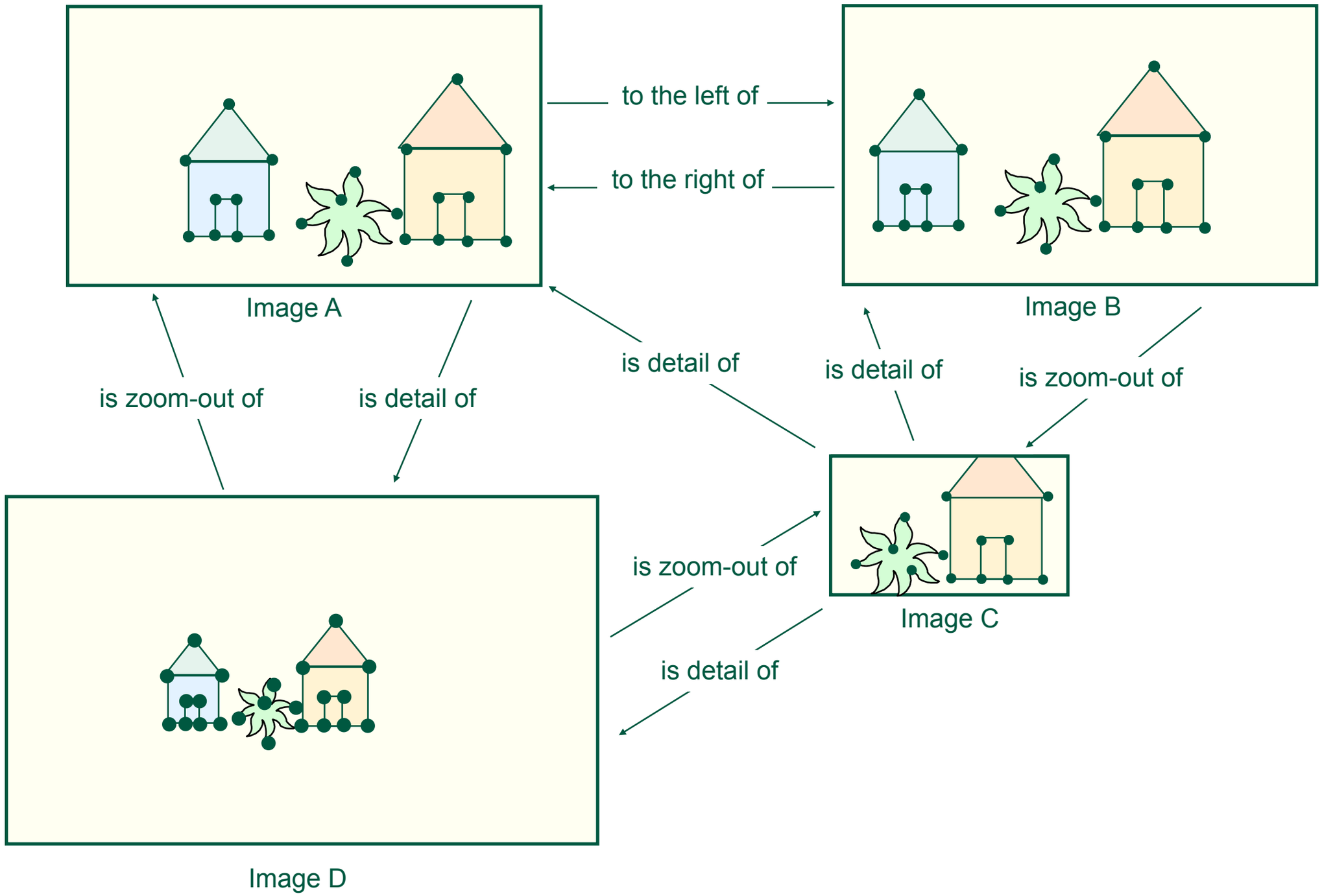
© 2006 Noah Snavely

Relation-based browsing



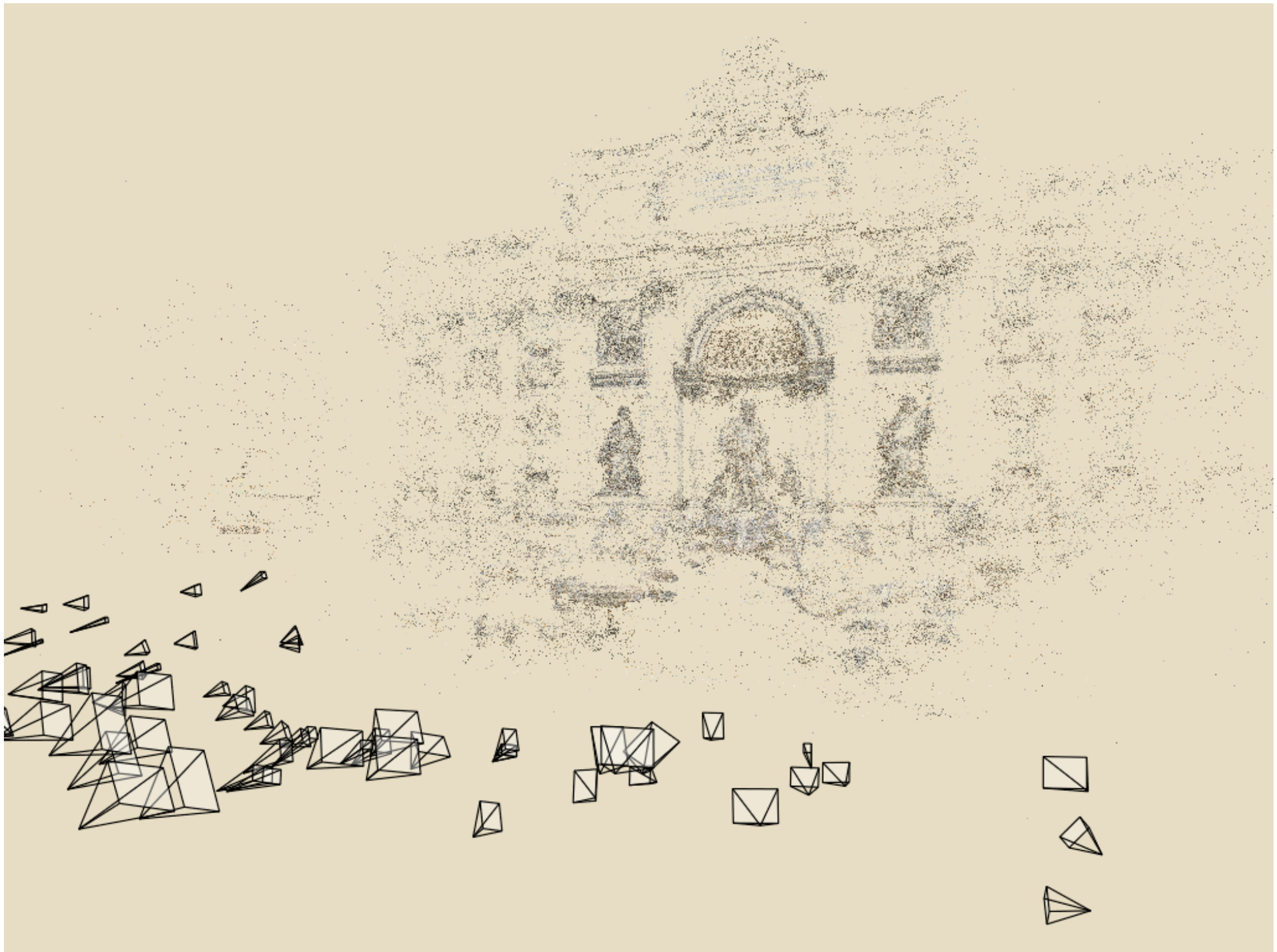
© 2006 Noah Snavely

Relation-based browsing



© 2006 Noah Snavely

Rendering



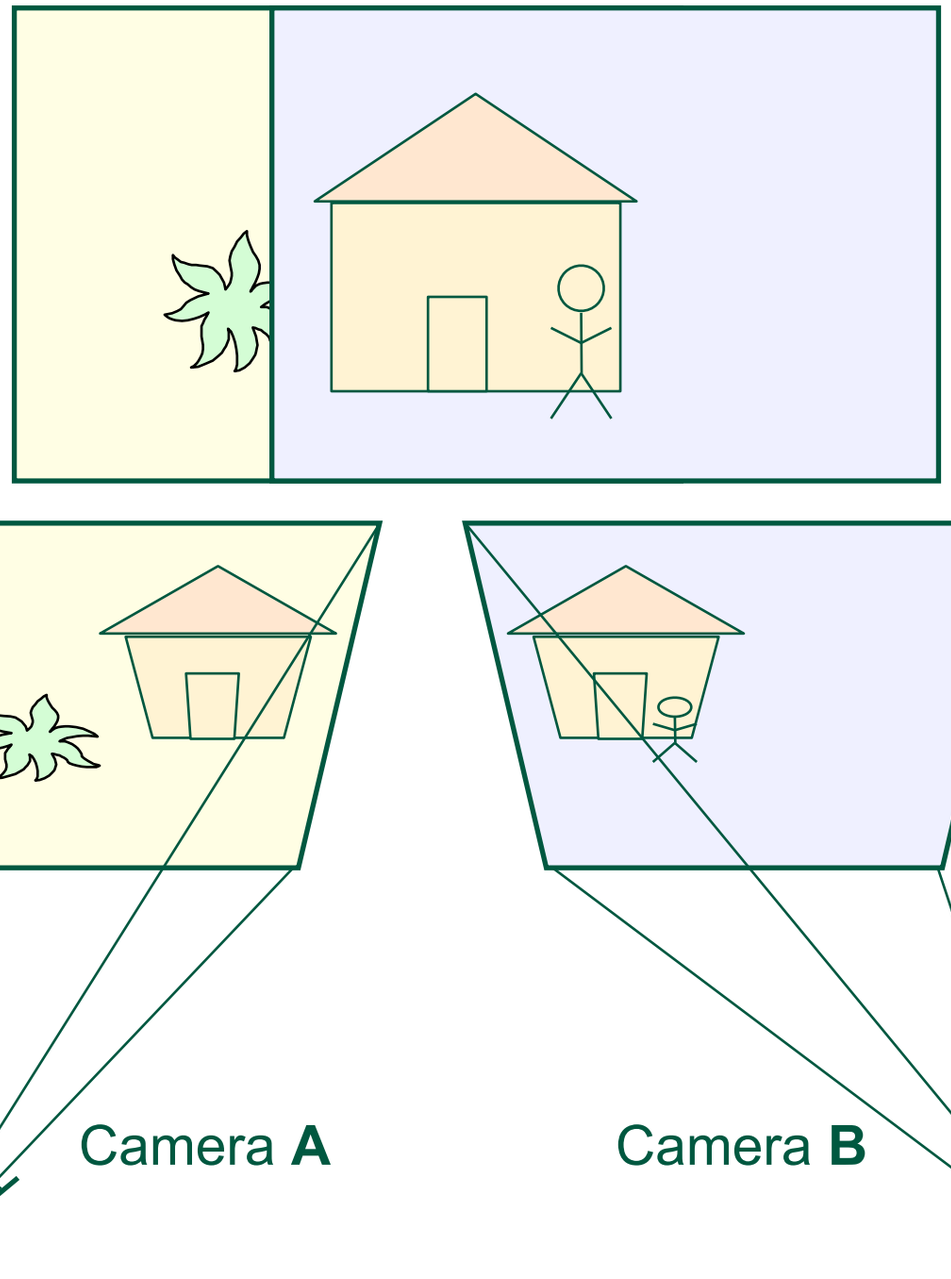
© 2006 Noah Snavely

Rendering



© 2006 Noah Snavely

Rendering transitions



© 2006 Noah Snavely

Photo Tourism

Exploring photo collections in 3D

Noah Snavely Steven M. Seitz Richard Szeliski

University of Washington *Microsoft Research*

SIGGRAPH 2006

Live demo

- http://phototour.cs.washington.edu/applet/index.html
- see also: http://photosynth.net/

© 2006 Noah Snavely

Limitations / Future work

- Not all photos can be reliably matched
  - → Better feature detection / matching
  - → Integrating GPS & other localization info.
- Structure from motion scalability
  - → More efficient (sparse) algorithms
- Plane-based transitions lack parallax
  - → Richer transitions
- Photo explorer scalability…

© 2006 Noah Snavely

Conclusion

Indexing everyone's photos provides a new way to share and experience our world

To find out more:

- http://phototour.cs.washington.edu
- http://research.microsoft.com/IVM/PhotoTourism
- http://labs.live.com/photosynth
- Exhibition booth #2619

Saint Basil's Cathedral

Trafalgar Square

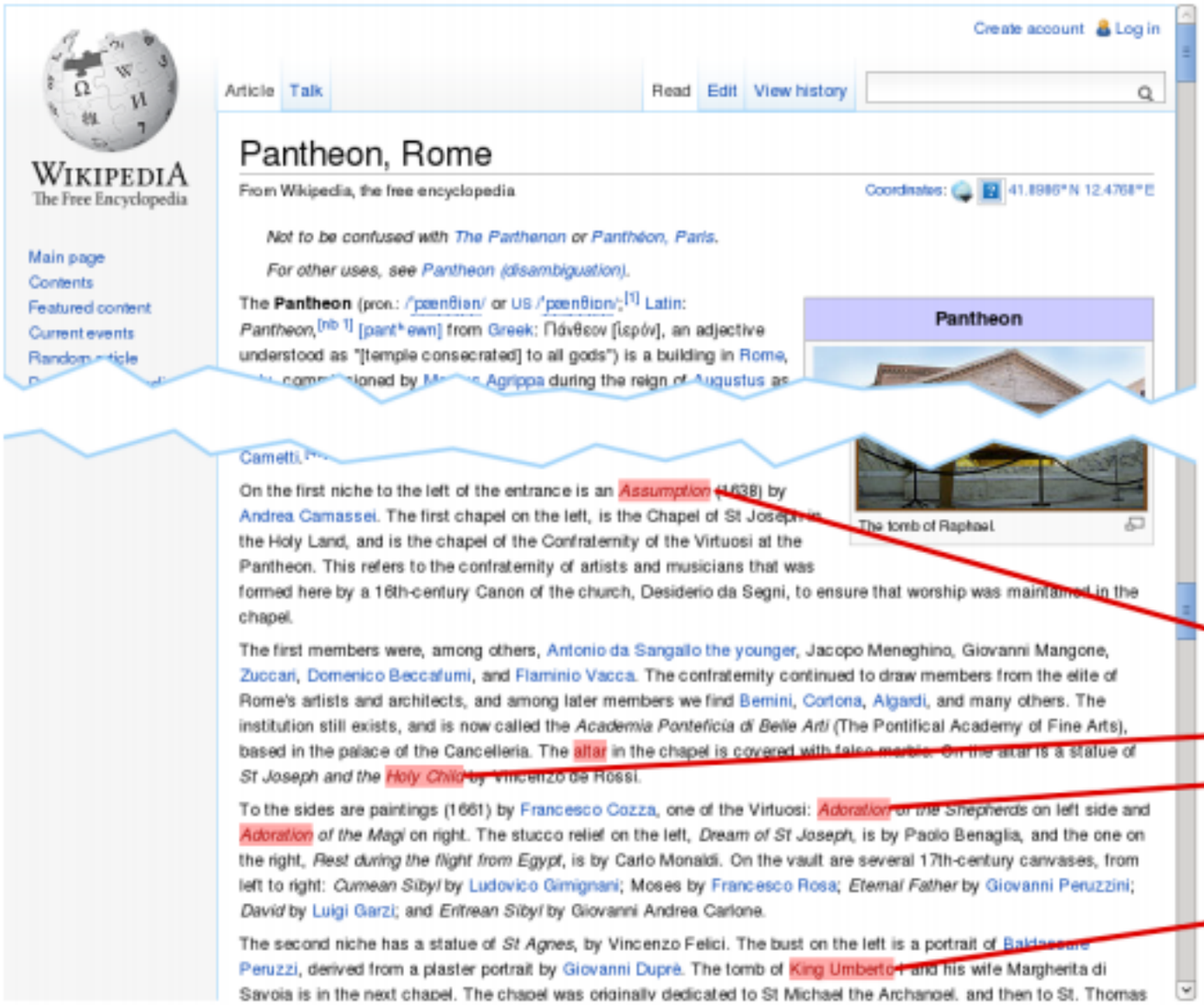
Rockefeller Center

Mount Rushmore

© 2006 Noah Snavely

3D wikipedia

- Bryan Russell et al., Siggraph Asia 2013
- http://grail.cs.washington.edu/projects/label3d/



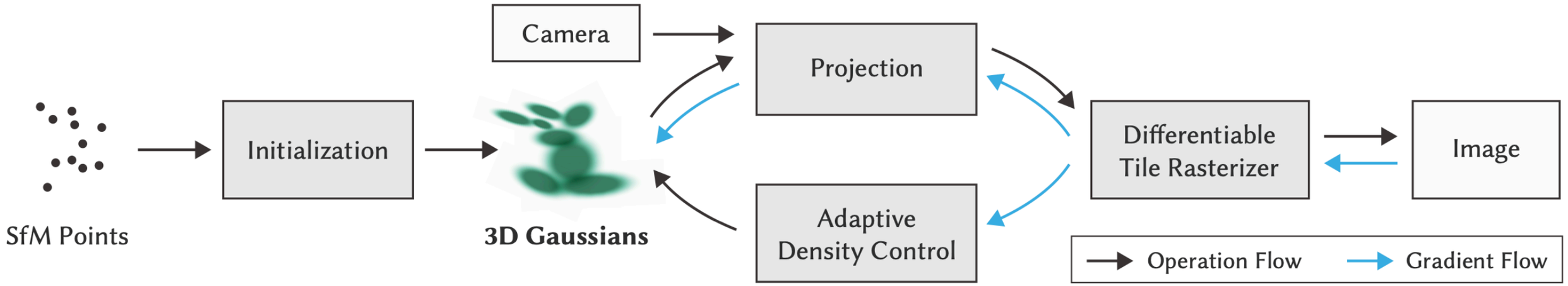
Time Lapse

- https://www.youtube.com/watch?v=oQpq4TM96Ow

© 2006 Noah Snavely

Colmap, Gaussian splatting

- https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/



© 2006 Noah Snavely

Is the future neural?

- https://openaccess.thecvf.com/content/ICCV2023/html/Lindenberger_LightGlue_Local_Feature_Matching_at_Light_Speed_ICCV_2023_paper.html
- https://openaccess.thecvf.com/content_cvpr_2018_workshops/w9/html/DeTone_SuperPoint_Self-Supervised_Interest_CVPR_2018_paper.html
-

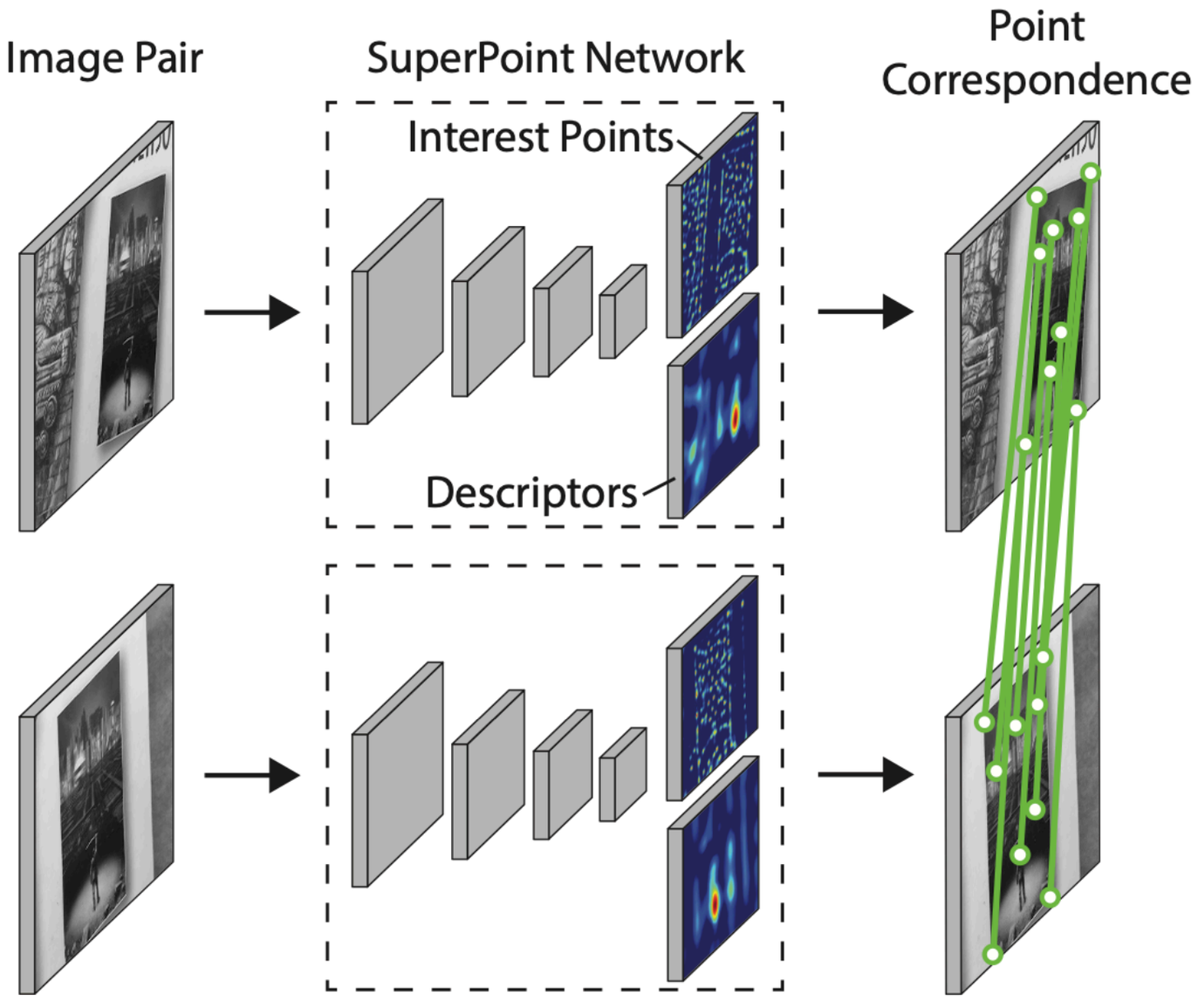


Figure 1. **SuperPoint for Geometric Correspondences.** We present a fully-convolutional neural network that computes SIFT-like 2D interest point locations and descriptors in a single forward pass and runs at 70 FPS on 480 × 640 images with a Titan X GPU.

© 2006 Noah Snavely

Is the future neural?

- https://europe.naverlabs.com/blog/mast3r-matching-and-stereo-3d-reconstruction/
- https://arxiv.org/abs/2409.19152





© 2006 Noah Snavely

Links

- Code available: http://phototour.cs.washington.edu/bundler/
- http://phototour.cs.washington.edu/
- http://livelabs.com/photosynth/
- http://www.cs.cornell.edu/~snavely/
- http://da.vidr.cc/projects/pixelstruct/
- http://www.cs.washington.edu/homes/ccwu/vsfm/

© 2006 Noah Snavely