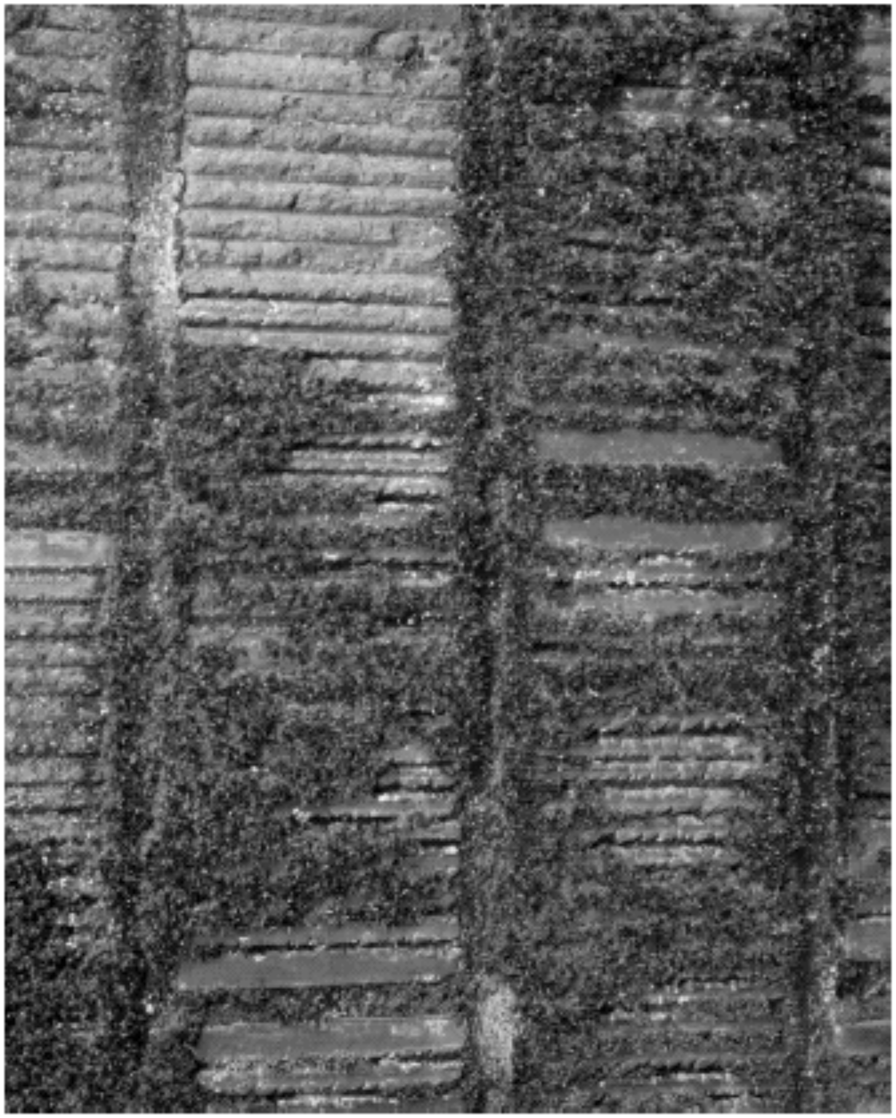
Texture review



A typical filter bank

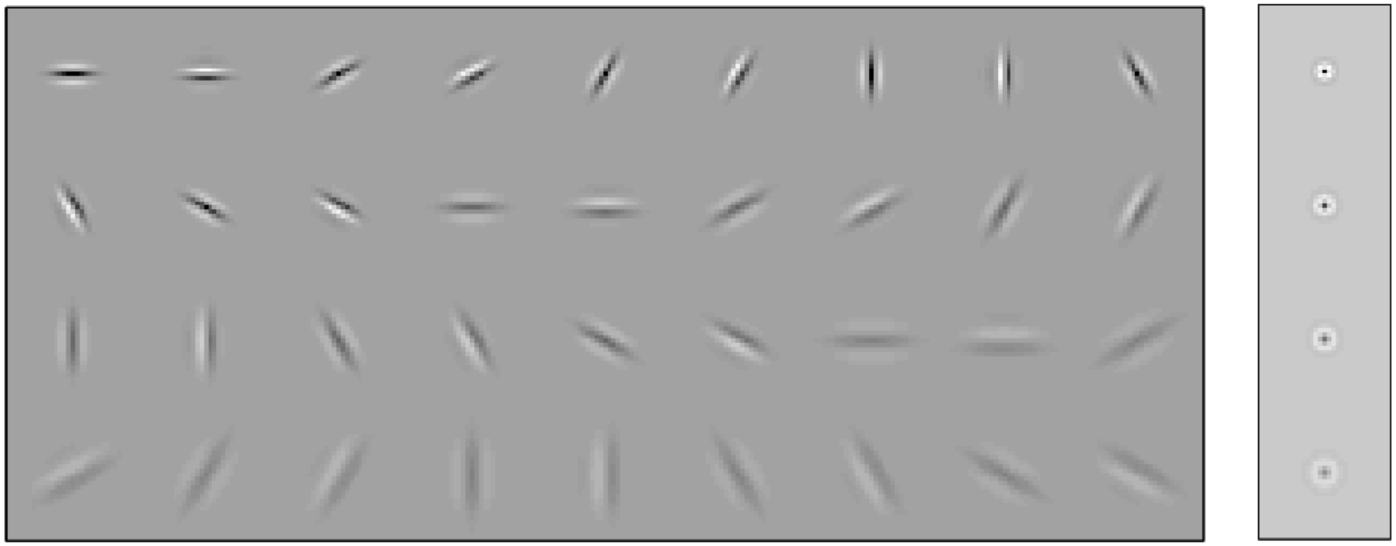
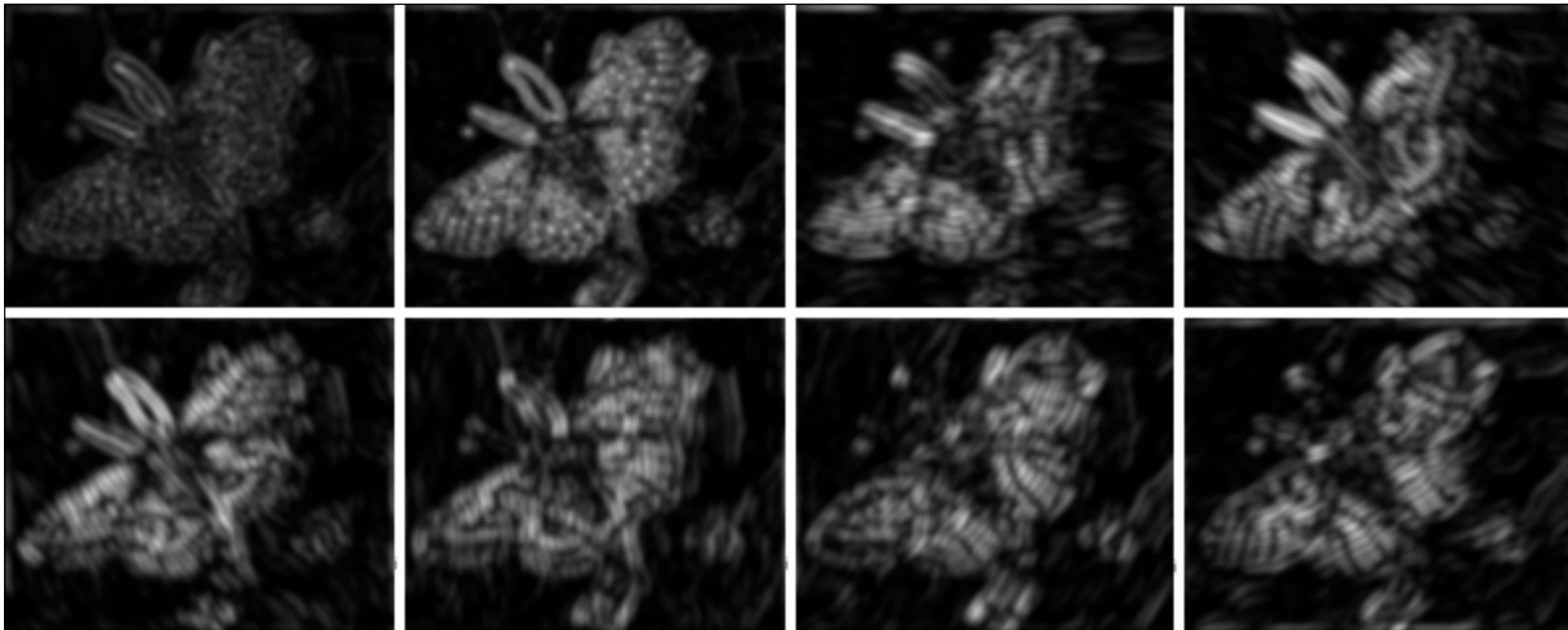


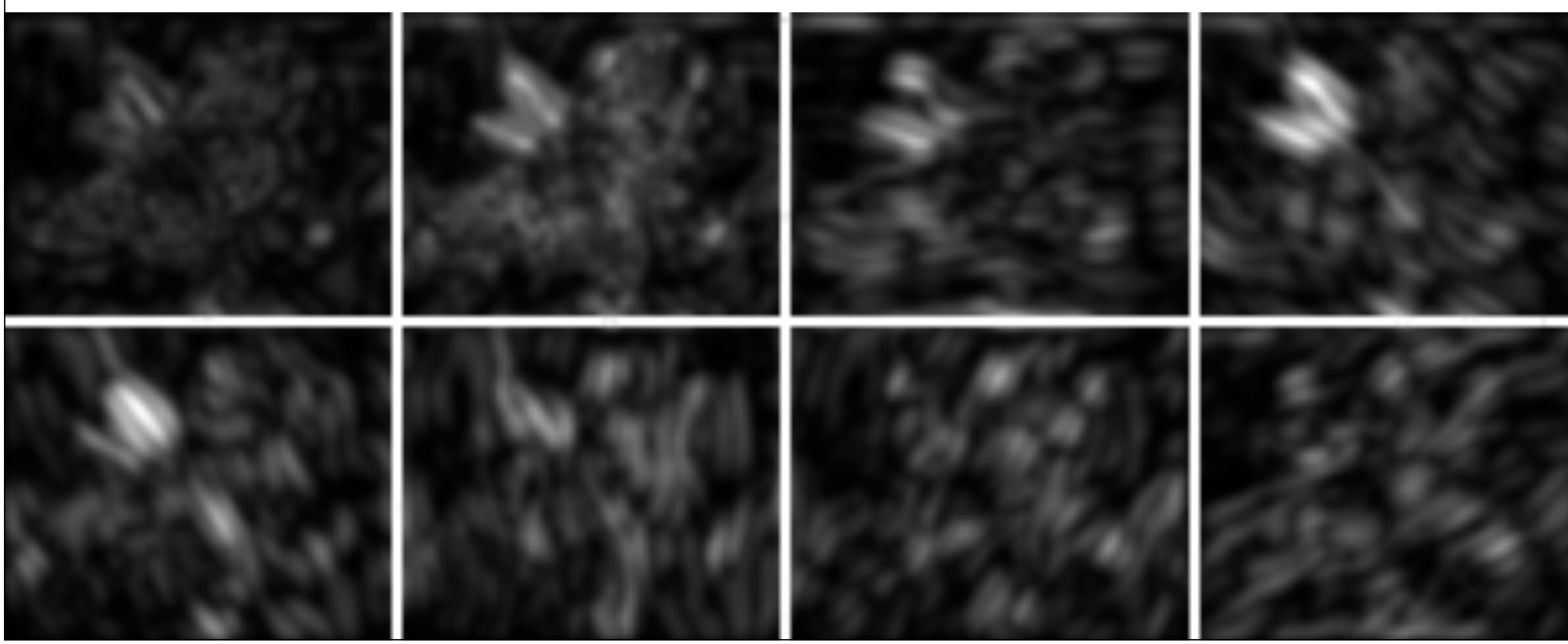
Figure 4. Left: Filter set f_i consisting of 2 phases (even and odd), 3 scales (spaced by half-octaves), and 6 orientations (equally spaced from 0 to π). The basic filter is a difference-of-Gaussian quadrature pair with $3:1$ elongation. Right: 4 scales of center-surround filters. Each filter is L_1-normalized for scale invariance.

From Malik et al., “Contour and texture analysis for image segmentation”

(We have an implementation for this filter bank, as part of the N-cuts software from Berkeley).



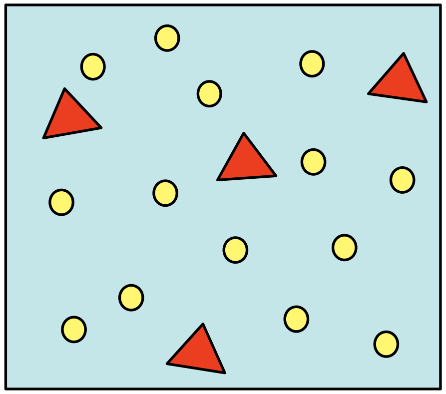
Smaller Scale



Larger Scale (Image from previous slide made larger to compare)

Textons

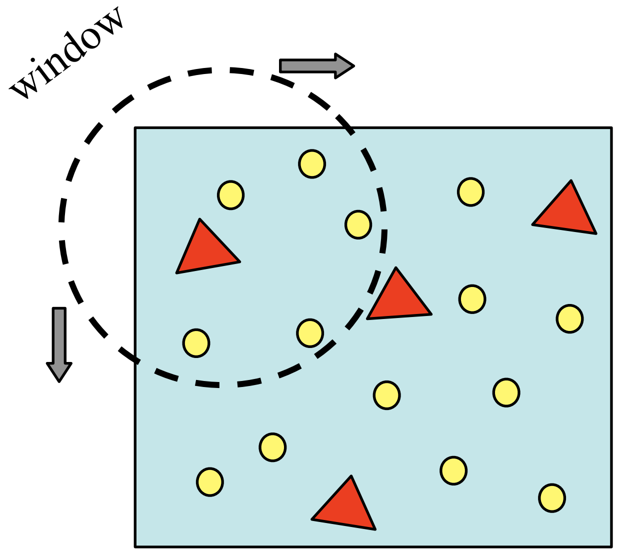
- Global statistics (e.g., mean of filter response magnitude) ignore spatial correlations
  - Some filters fire for the dots, other for the triangles.

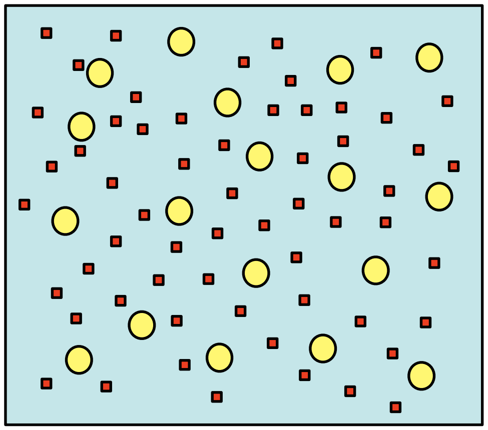


- A complete representation of filter responses over windows will not work
  - expensive, represents textures that are never seen, does not exploit internal similarity in texture (e.g., multiple dots).

- One good solution is to **cluster** point data in “textons”
  - Texture in a window is a histogram of texton popularity.

Example





A variety of filter shapes and scales provide numbers that can distinguish these textures and many others.

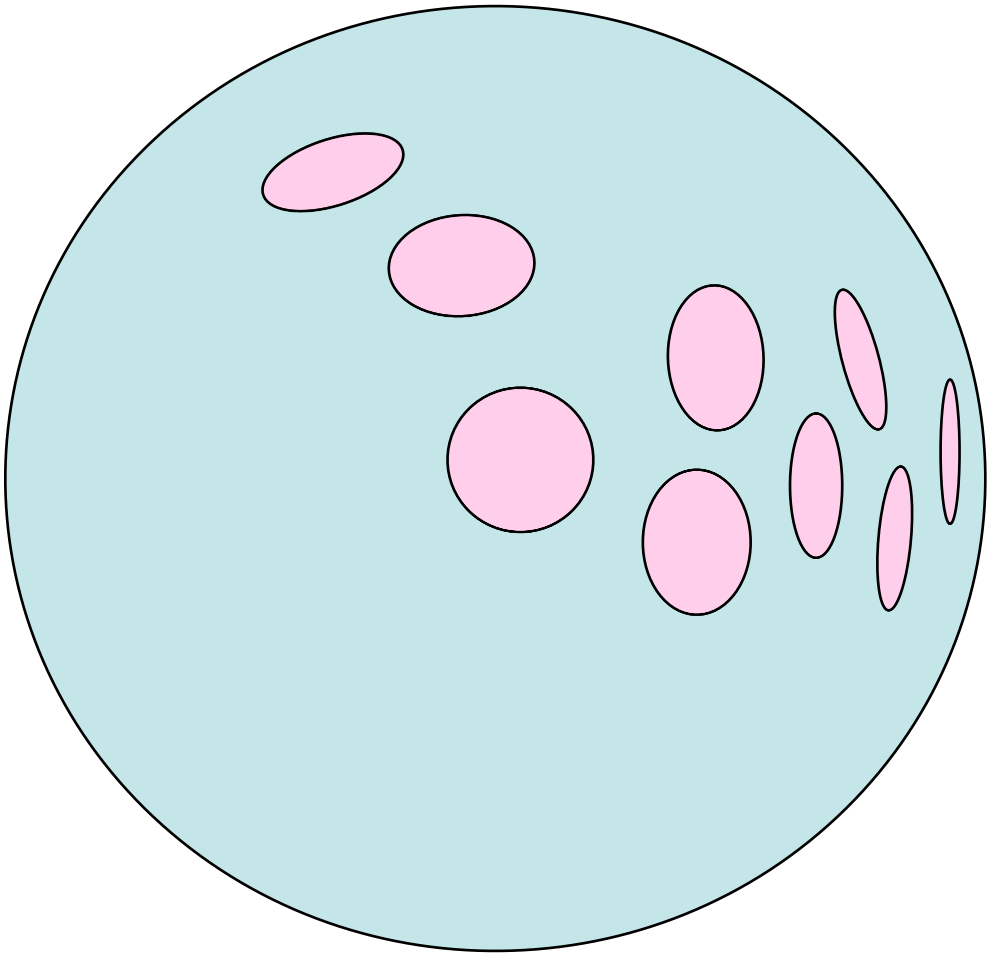
However, simple statistics (e.g., filter response magnitude and variance) do not capture the spatial correlation well.

Commonly co-occurring filter responses can be represented by clusters (textons).

Textures can be represented by histograms over textons.

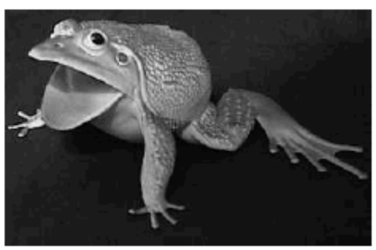
Texture (and color) segmentation

Shape from Texture



Invariant feature detection

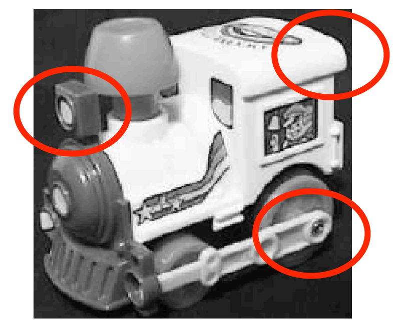
From Lowe,
IJCV 2004



Can you find the locomotive? Can a computer program?

Invariant feature detection

- Consider representing an image of an object as a collection of descriptive local features
  - Distinct
    - Good chance that match is correct
  - Invariant to scale and rotation
    - Match even if rotated
    - Match even if bigger or smaller
    - Match even brighter or darker



- Canonical example is David Lowe's SIFT (Scale Invariant Feature Transform) method.*

*Good reference is Lowe, IJCV, 2004

Invariant feature detection*

- Consider representing an image of an object with a collection of descriptive local features
- Most useful if these occur in “edgy” areas.
- Common modern strategy is do detect somewhat robust “interest points” and form a descriptor for the local area.
- Example descriptor is a histogram of edge orientations (local texture).

*Good reference is Lowe, IJCV, 2004

Distinctive Key-Points

- Edges are interesting, but are they really distinctive?
  - Not for many applications because they do not have good localization
  - More distinctive is a corner or a grid point
- Various strategies exist for finding "key-points" that are distinctive and localizable
- One idea is to look for edgy areas where one edge direction does NOT overly dominate the other
  - EG, a corner has both horizontal and vertical responses

Scale invariance

- SIFT achieves scale stability by focusing attention on structure that is defined in terms of scale.

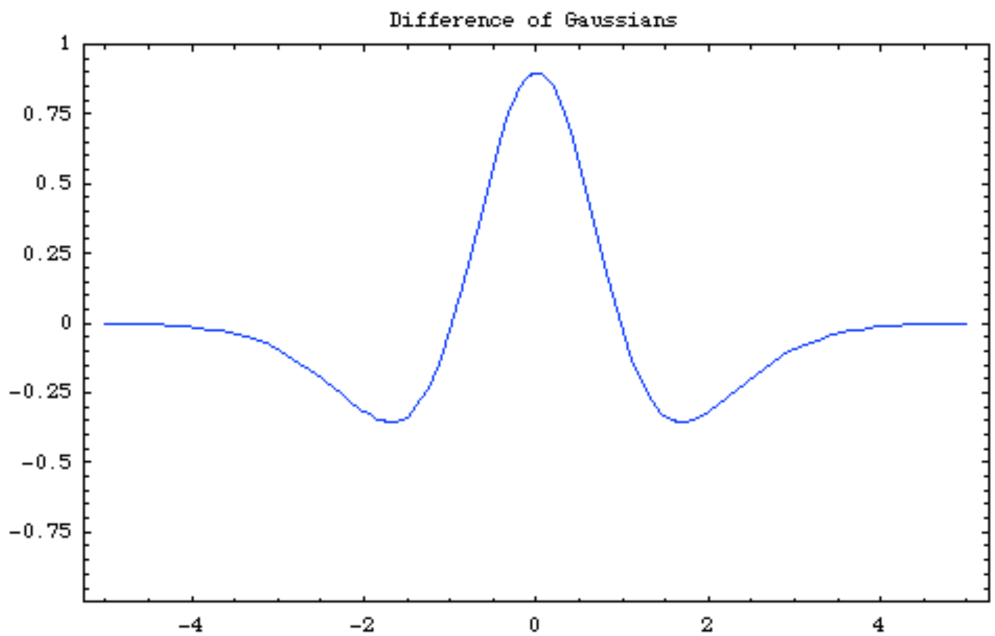
- If a structure has an inherent scale, then it can be extracted from an image of unknown scale by considering that image at different scales.

- *Image scale space*
  - Consider the images at many scales
  - Each scale leads to a different blurry image
  - Consider sigma as a 3rd coordinate
  - So an image “cube” now is (x,y,sigma)
  - In what follows, we use a *difference* scale space

Scale space

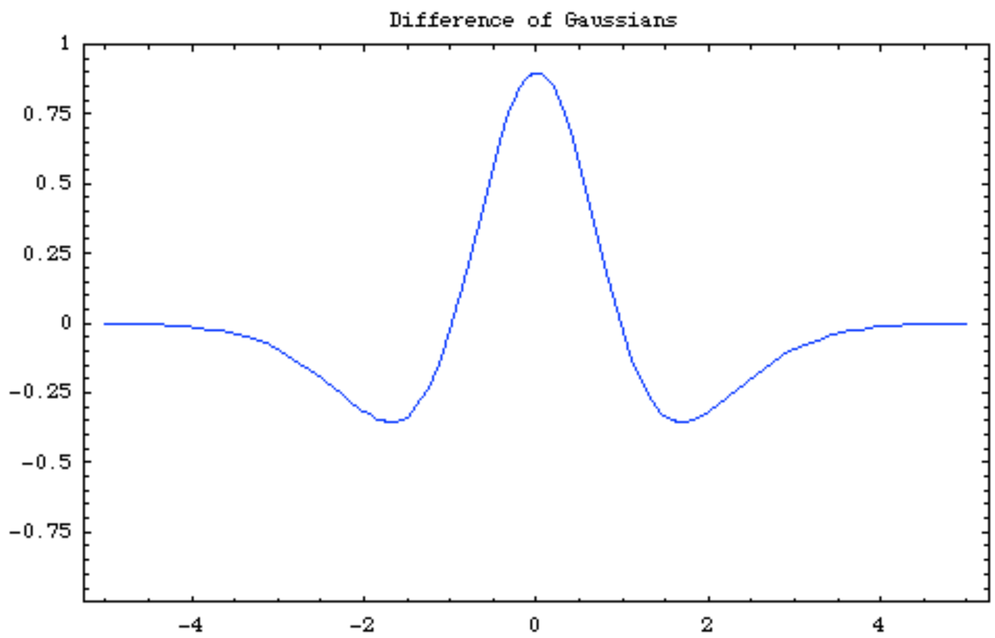
Difference of Gaussians

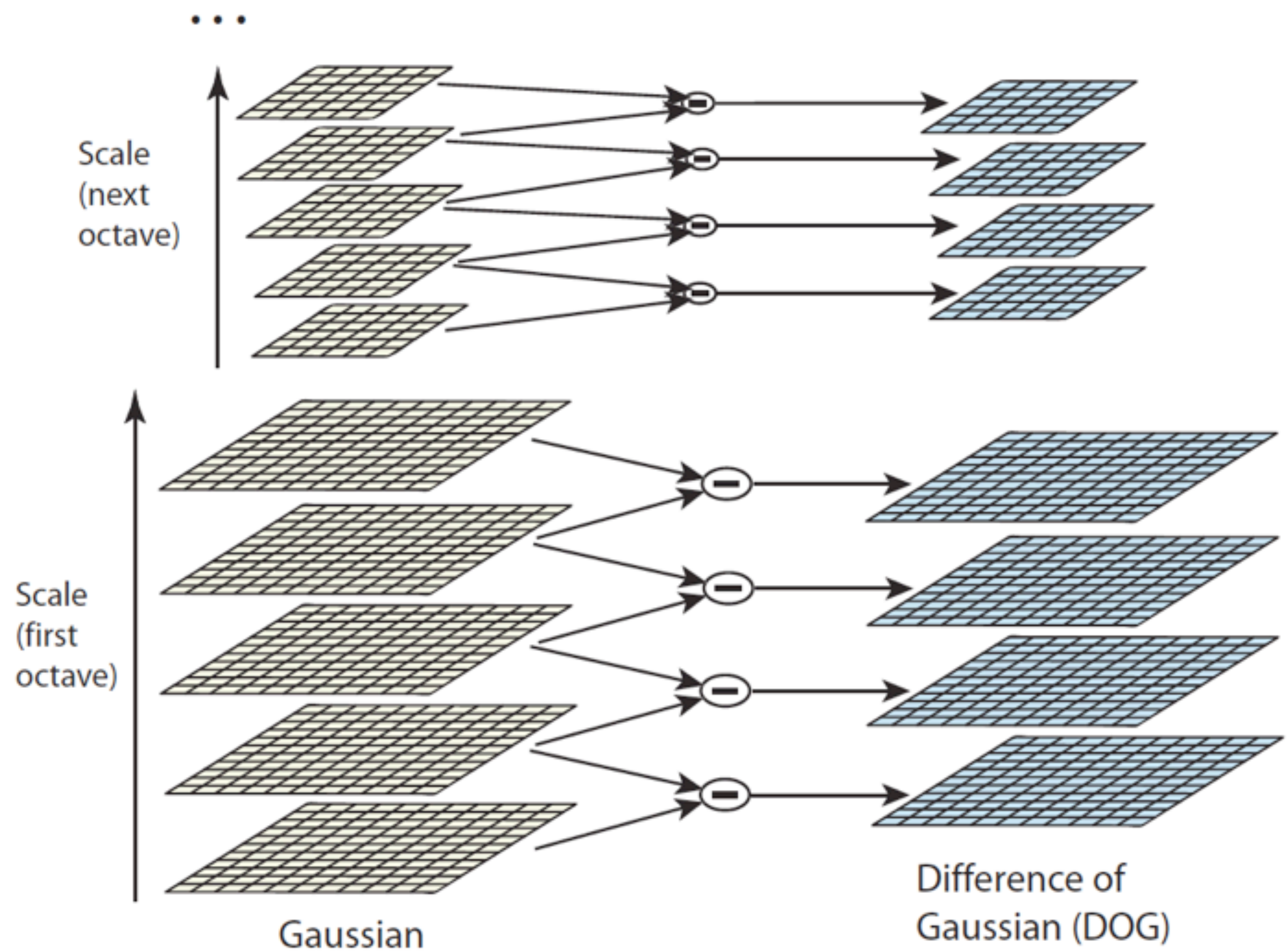
- A difference of two appropriate Gaussians is a blob detector
- Approximates a Laplacian of Gaussian (LOG)
  - Laplacian is a 2D analog of second derivative
- Essentially a center-surround filter



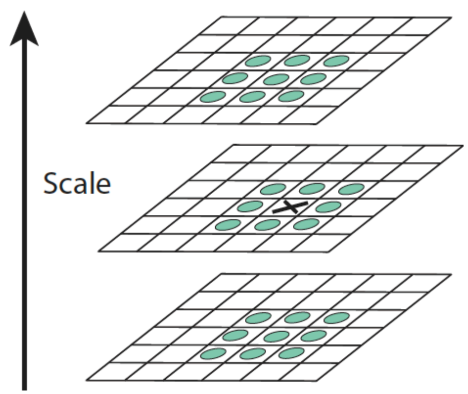
Difference of Gaussians

- A difference of two appropriate Gaussians is a blob detector
- Approximates a Laplacian of Gaussian (LOG)
  - Laplacian is a 2D analog of second derivative
- Essentially a center-surround filter




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

- For each scale we use the difference between the image at two successive (discrete) scales (σ and $k\sigma$).
- This detects structure in a scale invariant way (think blobs)
- Keep points that are optimal in three directions: (x, y, σ).



From Lowe 2004

Figure 2: Maxima and minima of the difference-of-Gaussian images are detected by comparing a pixel (marked with X) to its 26 neighbors in 3x3 regions at the current and adjacent scales (marked with circles).

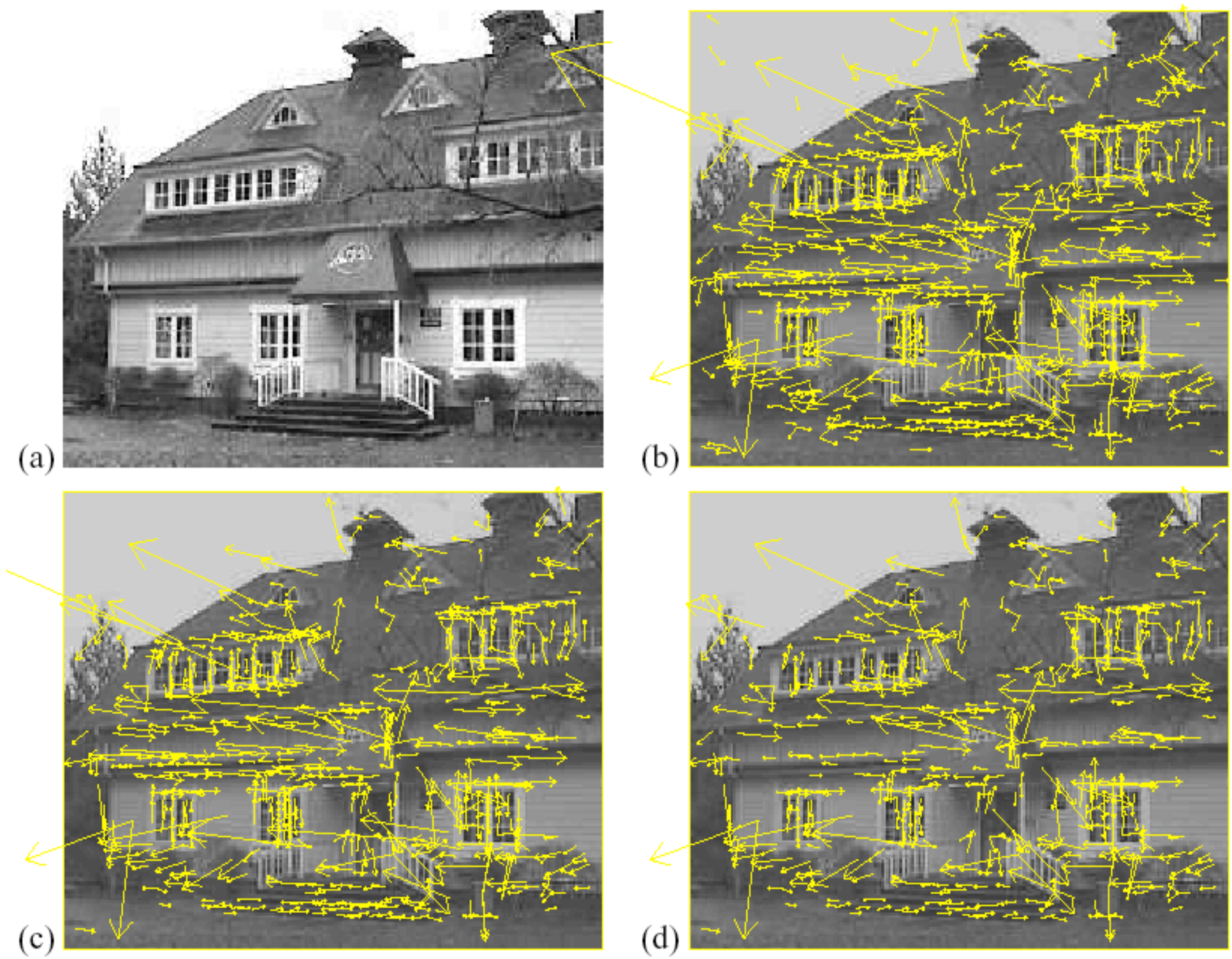
Distinctiveness

- For points that are optimum from previous …
  - These points have an associated scale

- Only points in regions that have significant edges in two directions are considered "distinctive" interest points (reject the rest)
  - Corners localize better than edges
  - The gradient gives only a single direction
  - We need to consider additional information around the point to distinguish corners from edges
  - One method is to collect edge information in a region around the point
    - Want the region to some energy in all directions
  - Lowe (04) uses a method based on "principle curvature"

Direction

- For points that have not been rejected …
  - These points have an associated scale σ.
- Consider a disc with radius $3\sigma/2$
- Look at edge direction a locations in this disc, and build a histogram of the angles.
- Values in a dominant peak provide a direction
  - A second direction can create a second keypoint if it is at least 80% as popular.
- From scale and direction, we can establish a coordinate system



From Lowe, IJCV 2004

(a) The 233x189 pixel original image. (b) The initial 832 keypoints locations at maxima and minima of the difference-of-Gaussian function. Keypoints are displayed as vectors indicating scale, orientation, and location. (c) After applying a threshold on minimum contrast, 729 keypoints remain. (d) The final 536 keypoints that remain following an additional threshold on ratio of principal curvatures.

Descriptor

- Capture the edge structure (essentially texture) in the region (of size of order σ) around the point in a vector (Lowe 04 uses 128 elements)

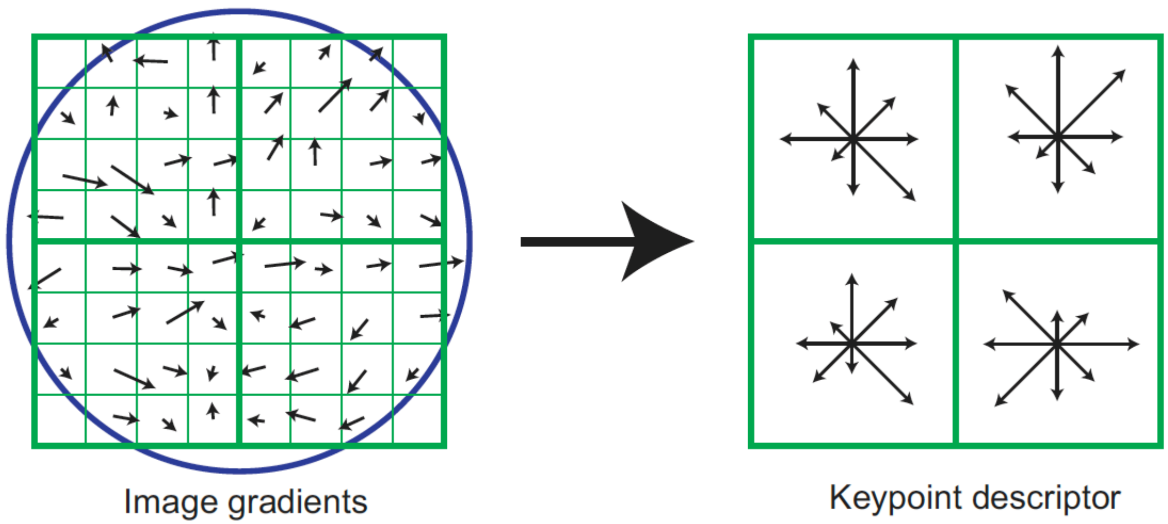


Figure 7: A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 subregions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

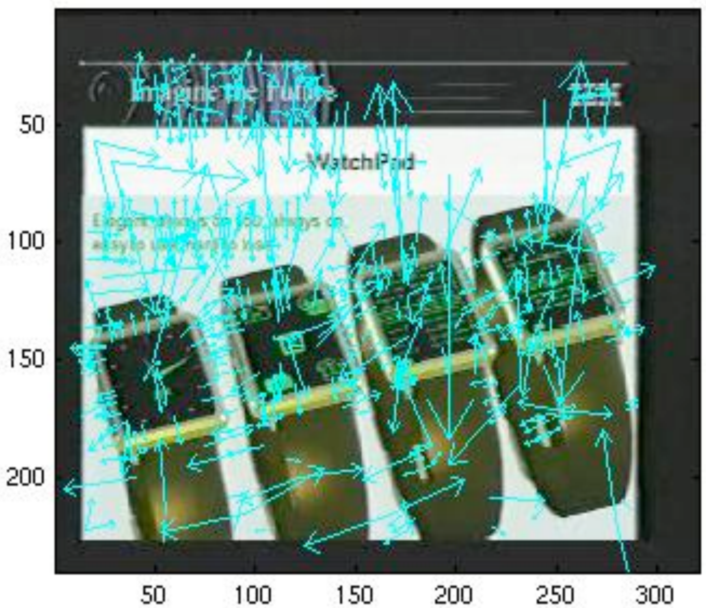
Invariant feature detection

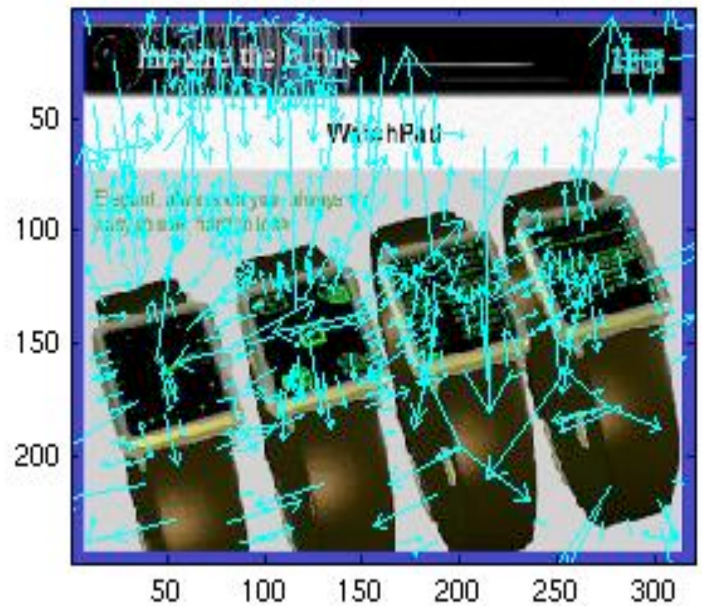
- Descriptor is invariant to scale and in plane rotation
  - Feature had a natural scale
  - We established a direction

- Scaling and rotation can approximate out of plane camera rotation view changes for small patches (locally planar)

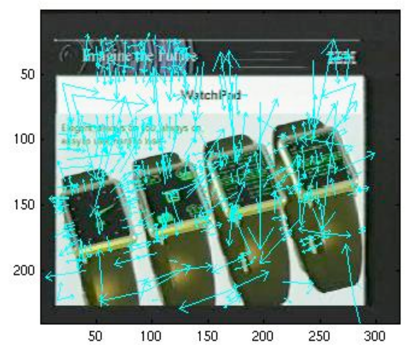
Invariant feature matching

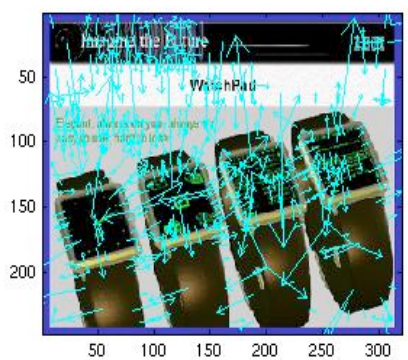
- To “find” the object, match the local features



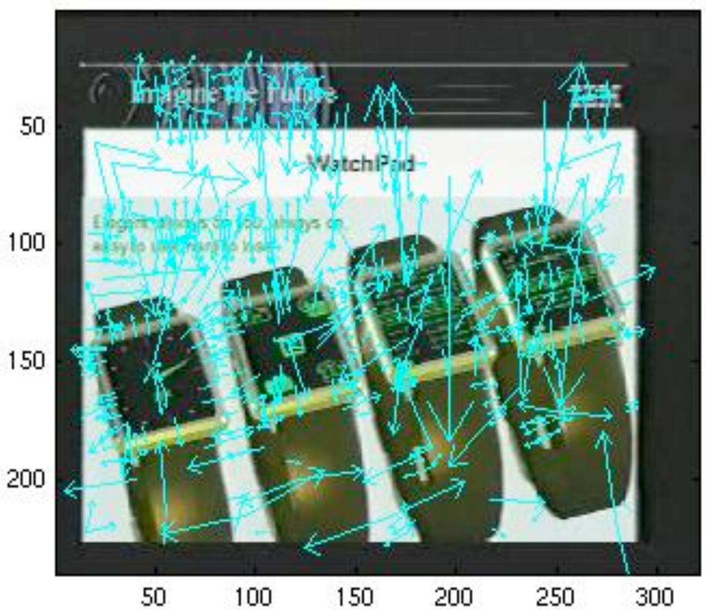


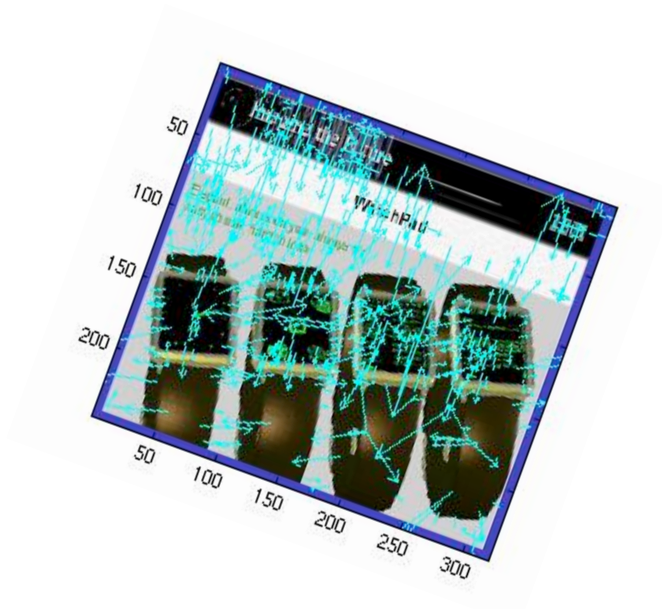
Invariant feature matching



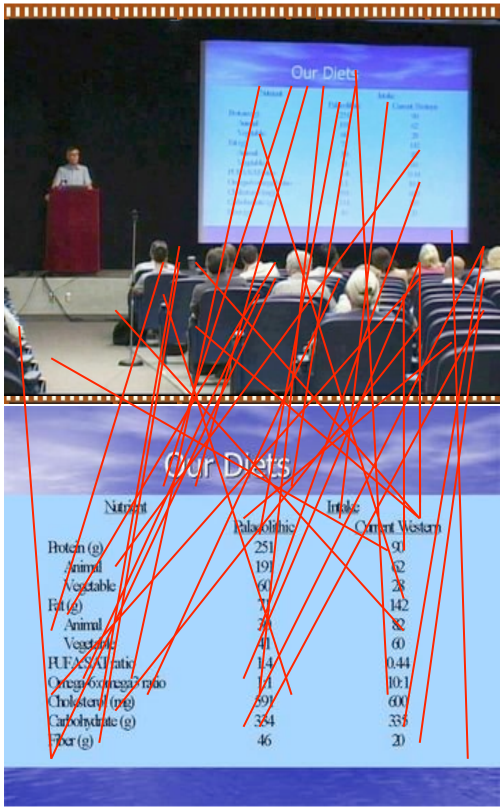


Nearest Neighbor search





Should work similarly to non-rotated case.



Initial matching

Constraining to correct part of image based on other information

After pruning outliers

(Covered later)

- Keypoints from an object map into an image in an organized way.
- We will study how to improve matching on this bases in the context of *grouping*.