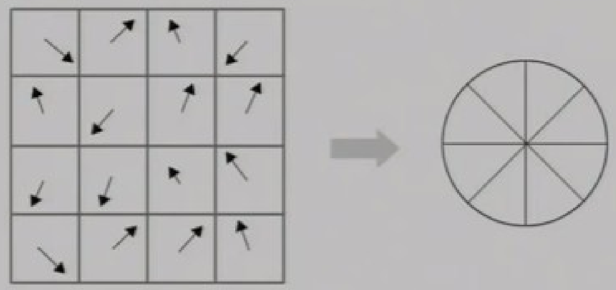
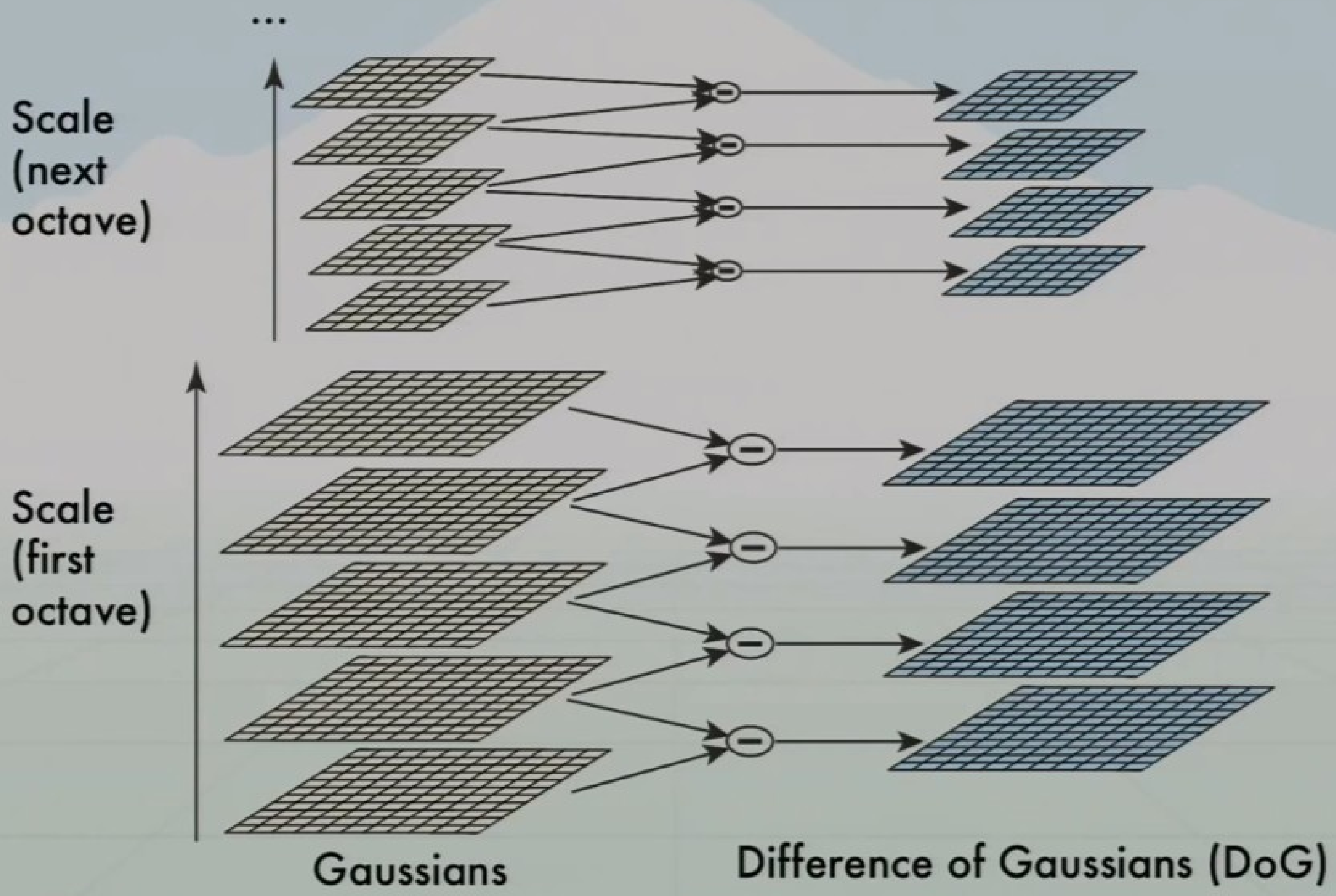
Histogram of Oriented Gradients (HOG)



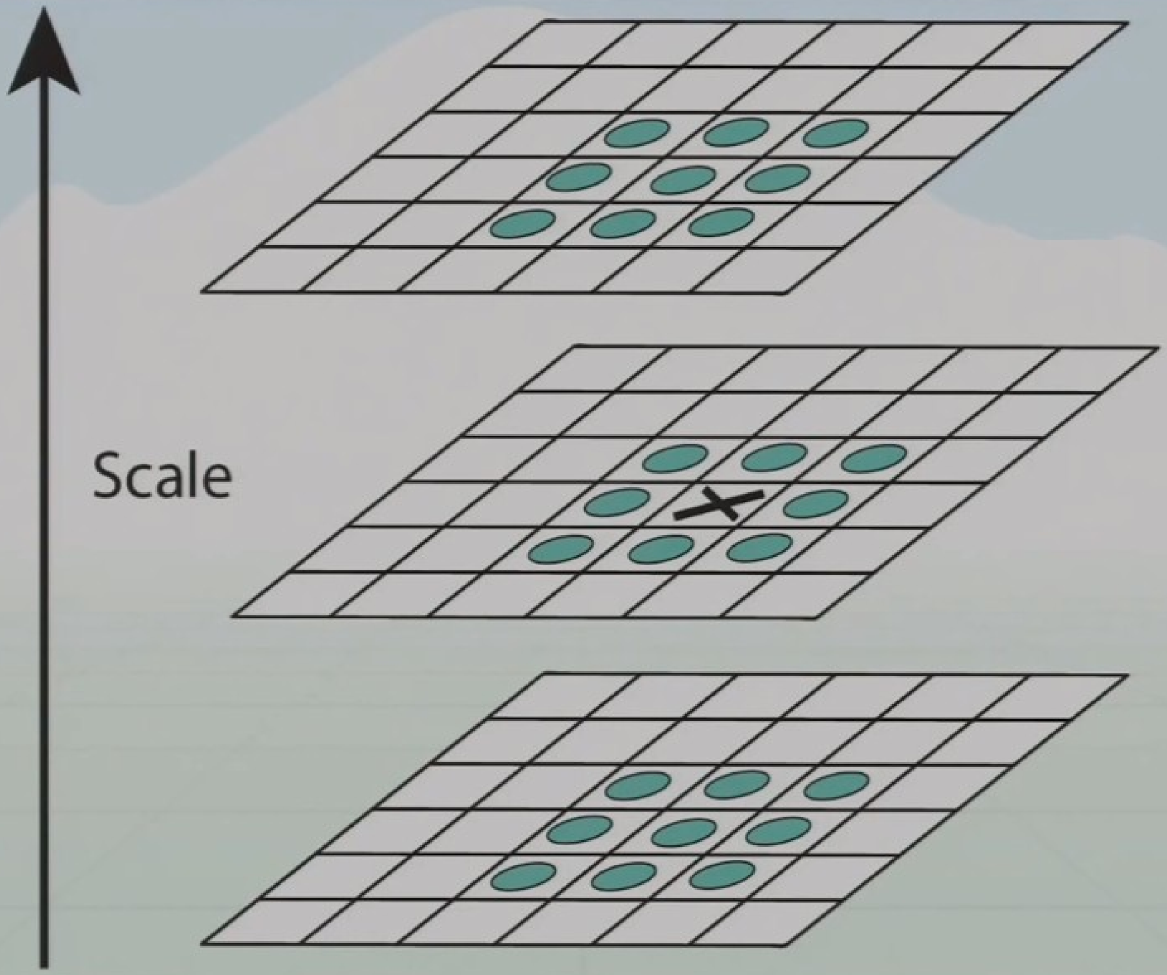
- Dalal and Triggs 2005
- Better image descriptor
  - Compute gradients
  - Bin gradients
  - Aggregate blocks (4x4, 16x16 cells)
  - Normalize gradient magnitudes
- Not reliant on magnitude, just direction
  - Invariant to some lighting changes
- Train SVM to recognize people

Binning from http://aishack.in/tutorials/sift-scale-invariant-feature-transform-features/

Extract DoG features at multiple scales



Find local-maxima in location and scale



Throw out weak responses and edges

- Estimate gradients
  - Similar to before, look at nearby responses
  - Not whole image, only a few points! Faster!
  - Throw out weak responses
- Find cornery things
  - Same deal, structure matrix, use det and trace information

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

Find main orientation of patches

- Look at weighted histogram of nearby gradients
  - Any gradient within 80% of peak gets its own descriptor
    - Multiple keypoints per pixel
  - Descriptors are normalized based on main orientation

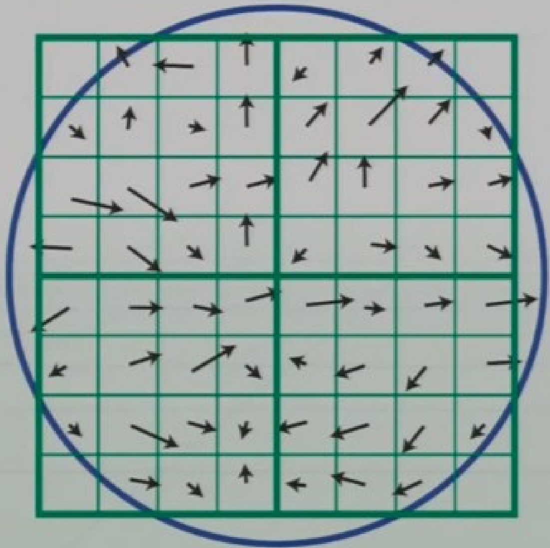
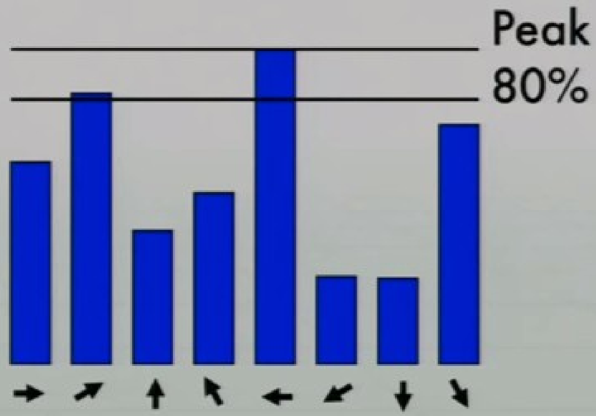
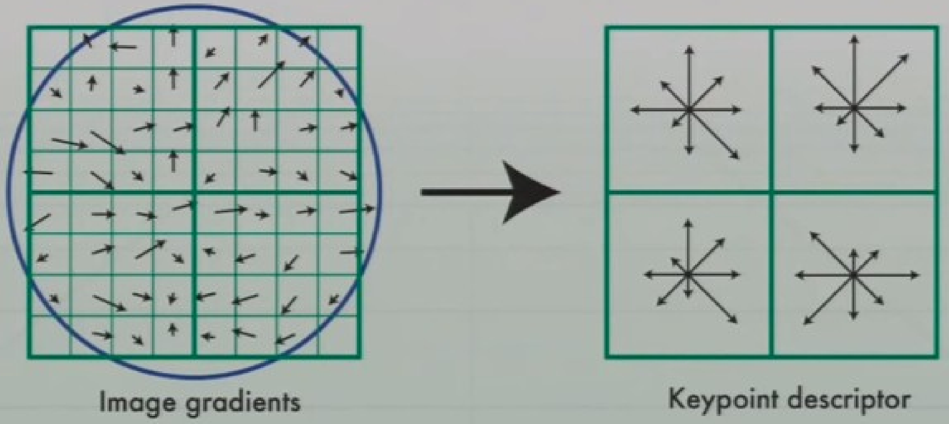


Image gradients



Keypoints are normalized gradient histograms

- Divide into subwindows (2x2, 4x4)
- Bin gradients within subwindow, get histogram
  - Normalize to unit length
  - Clamp at maximum .2
  - Normalize again
  - Helps with lighting changes!



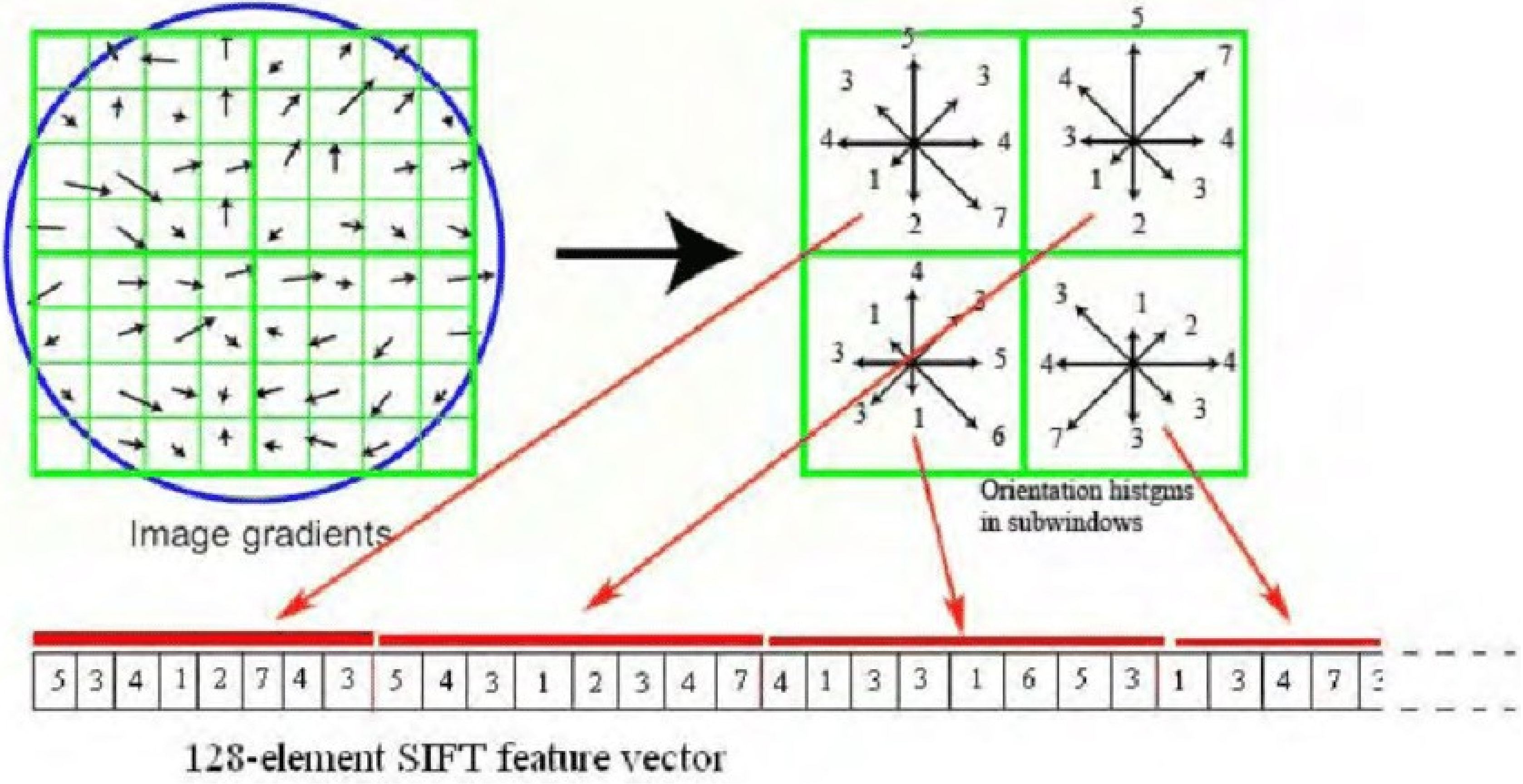
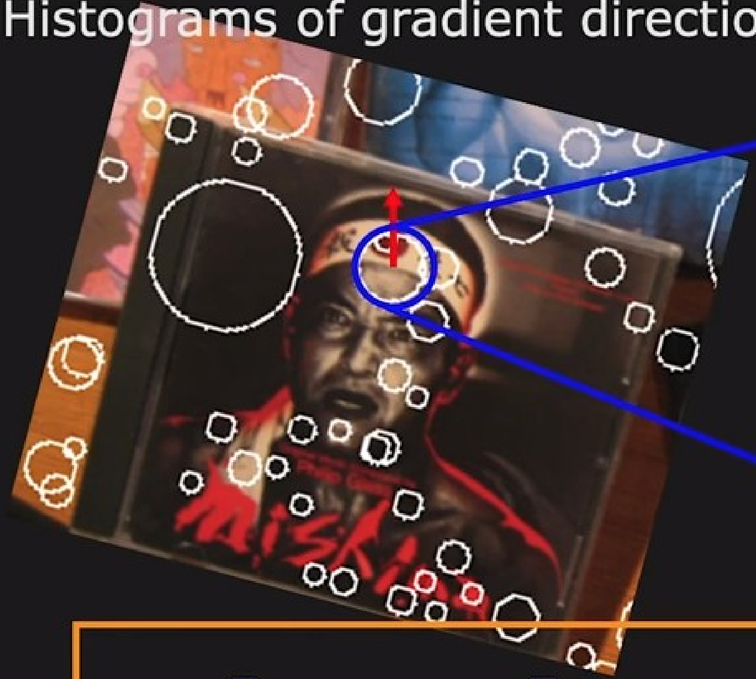
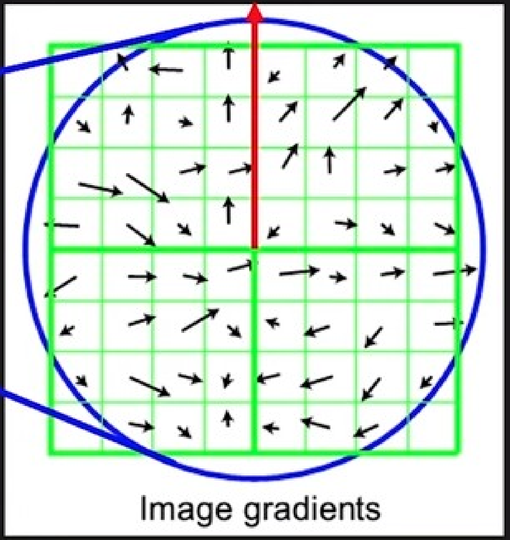


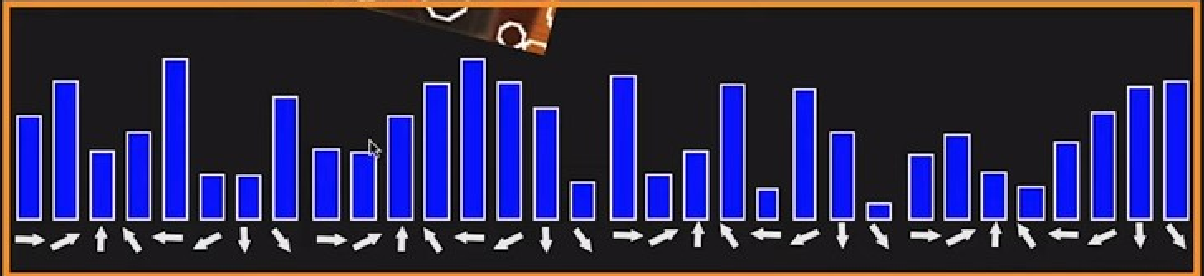
Image gradients
Orientation histgms
in subwindows
5 3 4 1 2 7 4 3 5 4 3 1 2 3 4 7 4 1 3 3 1 6 5 3 1 3 4 7
128-element SIFT feature vector

SIFT Descriptor

Histograms of gradient directions over spatial regions







Normalized Histogram: Invariant to Rotation, Scale, Brightnes

[Low

Comparing SIFT Descriptors

Essentially comparing two arrays of data.

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N.

L2 Distance:

$$d(H_1, H_2) = \sqrt{\sum_k \left(H_1(k) - H_2(k)\right)^2}$$

Smaller the distance metric, better the match.

Perfect match when $d(H_1, H_2) = 0$

Comparing SIFT Descriptors

Essentially comparing two arrays of data.

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N.

Normalized Correlation:

$$d(H_1, H_2) = \frac{\sum_k [(H_1(k) - \bar{H}_1)(H_2(k) - \bar{H}_2)]}{\sqrt{\sum_k (H_1(k) - \bar{H}_1)^2} \sqrt{\sum_k (H_2(k) - \bar{H}_2)^2}}$$

where: $\bar{H}_i = \frac{1}{N} \sum_{k=1}^{N} H_i(k)$

Larger the distance metric, better the match.

Perfect match when $d(H_1, H_2) = 1$

Comparing SIFT Descriptors

Essentially comparing two arrays of data.

Let $H_1(k)$ and $H_2(k)$ be two arrays of data of length N.

Intersection:

$$d(H_1, H_2) = \sum_k \min\left(H_1(k), H_2(k)\right)$$

Larger the distance metric, better the match.