SIFT computation
SIFT computation
Orientation Histogram

- 4x4 spatial bins (16 bins total)
- 8-bin orientation histogram per bin
- 8 x 16 = 128 dimensions total
- Normalized to unit norm
SIFT descriptor

- Image patch descriptor
- Location and characteristic scale: Blob/Cornor detector
- Find orientation from orientation histogram
SIFT invariances

- Spatial binning: tolerance to small shifts in location
- Orientation normalization
- Photometric normalization by making all vectors unit norm
- Orientation histogram: robustness to small local deformations
Application: Image Matching
Assumption: images undergo global deformations with a few degrees-of-freedom (e.g. scaling, rotation)

Correspondences of a few points suffice (found e.g. with SIFT)

Richard Szeliski’s talk, tomorrow
Introduction to Interest Point Detectors and Descriptors

Open source implementation: www.vlfeat.org

This section features a number of tutorials illustrating some of the main algorithms implemented in VLFeat. The tutorials can be categorized. The first class of algorithms detect and describe image regions (features). The second class of algorithms cluster.

Features

- Covariant detectors. An introduction to computing co-variant features like Harris-Affine.
- Histogram of Oriented Gradients (HOG). Getting started with this ubiquitous representation for object recognition.
- Scale Invariant Feature Transform (SIFT). Getting started with this popular feature detector and descriptor.
- Dense SIFT (DSIFT) and PHOW. A state-of-the-art descriptor for image categorization.
- Maximally Stable Extremal Regions (MSER). Extracting MSERs from an image.
- Image distance transform. Compute the image distance transform for fast part models and edge matching.

Clustering

- Integer optimized k-means (IKM). A quick overview of VLFeat fast k-means implementation.
- Hierarchical k-means (HIKM). Create a fast k-means tree for integer data.
- Agglomerative Information Bottleneck (AIB). Cluster discrete data based on the mutual information between the data and class.
- Quick shift. An introduction which shows how to create superpixels using this quick mode seeking method.
- SLIC. An introduction to SLIC superpixels.

Other

- Pegasos SVM. Learn a binary classifier and check its convergence plotting the energy value.
- Forests of kd-trees. Approximate nearest neighbor queries in high dimensions using an optimized forest of kd-trees.
- Plotting functions for rank evaluation. Learn how to plot ROC, DET, and precision-recall curves.
- MATLAB Utilities. A list of useful MATLAB functions bundled with VLFeat.
Further reading (literature ‘seeds’)

- **Compact Codes & Large-scale Retrieval**
  - A. Babenko and V. Lempitsky, The Inverted Multi-Index, CVPR 12
  - R. Arandjelović, A. Zisserman, All about VLAD, CVPR 2013

- **Fast/Compact Descriptors**
  - SURF, FAST, ORB, FREAK,...
Further reading (literature ‘seeds’)

- **Feature encoding**
  - The devil is in the details: an evaluation of recent feature encoding methods, K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, BMVC, 2011.

- **Descriptor Learning**
Dense descriptors

- Interest point detection revisited

D. Lowe, Perceptual Organization and Visual Recognition, 1985
F. Jurie, B. Triggs, Sampling strategies for bag-of-work classification, 2005
Dense descriptors

- Interest point detection revisited: there is nothing special about corners

D. Lowe, Perceptual Organization and Visual Recognition, 1985
F. Jurie, B. Triggs, Sampling strategies for bag-of-work classification, 2005
Histogram of Orientated Gradients (HOG) descriptor

- Dalal and Triggs, ICCV 2005
  - Like SIFT descriptor, but for arbitrary box aspect ratio, and computed over all image locations and scales
  - Highly accurate detection using linear classifier

Feature vector $f = [\ldots, \ldots, \ldots]$
Part score computation

\[ s[x] = \sum_{y} \langle h[x + y], w[y] \rangle \]
Part score \[ s[x] = \sum_y \langle h[x + y], w[y] \rangle \]