David Lowe’s recognition system

Lecture 5 - Spring 2005
EE/CNS/CS 148 - P. Perona
Invariance to translation
Invariance to orientation
Invariance to scale
Moreover...

• Clutter
• Lighting
• Occlusion
• ...
Abstract
This paper presents a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pose parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.
Database of individual objects
Interest points detector

Difference of gaussians
Figure 1: For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space images shown on the left. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images on the right. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated.
Detection & scale of features

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).
Figure 3: The top line of the first graph shows the percent of keypoints that are repeatably detected at the same location and scale in a transformed image as a function of the number of scales sampled per octave. The lower line shows the percent of keypoints that have their descriptors correctly matched to a large database. The second graph shows the total number of keypoints detected in a typical image as a function of the number of scale samples.
Figure 4: The top line in the graph shows the percent of keypoint locations that are repeatably detected in a transformed image as a function of the prior image smoothing for the first level of each octave. The lower line shows the percent of descriptors correctly matched against a large database.
Figure 5: This figure shows the stages of keypoint selection. (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.
Choice of descriptors

- x, y location
- orientation
- scale
- Description of the local neighborhood by a collection of gradients
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.
Stereo system
Matching across images
Matching across viewpoints
Matching across viewpoints
Figure 6: The top line in the graph shows the percent of keypoint locations and scales that are repeatably detected as a function of pixel noise. The second line shows the repeatability after also requiring agreement in orientation. The bottom line shows the final percent of descriptors correctly matched to a large database.
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.
Repeatability vs affine distortion

Figure 9: This graph shows the stability of detection for keypoint location, orientation, and final matching to a database as a function of affine distortion. The degree of affine distortion is expressed in terms of the equivalent viewpoint rotation in depth for a planar surface.
Figure 10: The dashed line shows the percent of keypoints correctly matched to a database as a function of database size (using a logarithmic scale). The solid line shows the percent of keypoints assigned the correct location, scale, and orientation. Images had random scale and rotation changes, an affine transform of 30 degrees, and image noise of 2% added prior to matching.
Kd-tree search

An exhaustive search is needed to obtain the nearest neighbor. This is time consuming!!
Kd-tree search - backtracking
Kd-tree search - backtracking
Kd-tree search - backtracking

Good choice!
Figure 11: The probability that a match is correct can be determined by taking the ratio of distance from the closest neighbor to the distance of the second closest. Using a database of 40,000 keypoints, the solid line shows the PDF of this ratio for correct matches, while the dotted line is for matches that were incorrect.
Geometrical consistency

Transform predicted by this match:
- $\Delta x = x_2 - x_1$
- $\Delta y = y_2 - y_1$
- $\Delta s = s_2 / s_1$
- $\Delta \theta = \theta_2 / \theta_1$

$\Rightarrow$ Voting performed in the space of transform parameters
We look at the space of parameters (Hough transform) (localization, orientation, scale $\Rightarrow$ 4 dimensions)
Lowe’s algorithm at a glance

- Detect features using scale-space DOG
- Compute SIFT descriptors
- Index into feature database
- Enforce pose consistency
- Count votes and declare winner(s)
Results: scene recognition
Results: multiple object instances
Recognition in presence of occlusions
Recognition with a change of scale
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 affine transformation solved for during recognition. Smaller squares indicate the keypoints that were used for recognition.