SIFT and SIFT-inspired Feature Detectors

CRV Tutorial Presentation

Geoffrey Treen

Carleton University

Detecting Local Interest Points

- Generally, consists of two processes:
  - Feature point detection
  - Feature point description
- Detection:
  - Goal is to find interesting points in an image that are robust to image transformations (rotation, scaling), viewpoint changes and image noise.
- Description:
  - Goal is to construct a unique signature/identifier for the feature point such that it can be reliably identified from thousands (or millions) of other feature points (feature matching).
- It's OK to mix and match detectors/descriptors (i.e. Harris corner detector with SIFT descriptor)

SIFT

- Stands for “Scale-Invariant Feature Transform”
- Invented in 1999 by David Lowe (UBC).
- Very popular “high-end” detection/description algorithm
    - Robust detection, distinct description but computationally expensive
- Invariant to image rotation, scaling, linear illumination
- Partially invariant to 3D viewpoint change
- Takes about 1s to compute 1000 SIFT features in a typical image (standard dual-core processor)
- SIFT features are described with 128-byte vectors

SIFT (cont'd)

- SIFT features used for:
  - Content-based image retrieval
  - Video event classification
  - Object recognition / tracking
  - Image classification
  - Markerless motion capture
  - Building panoramas
  - Mobile surveillance
  - Face authentication
  - etc.

SIFT (cont'd)

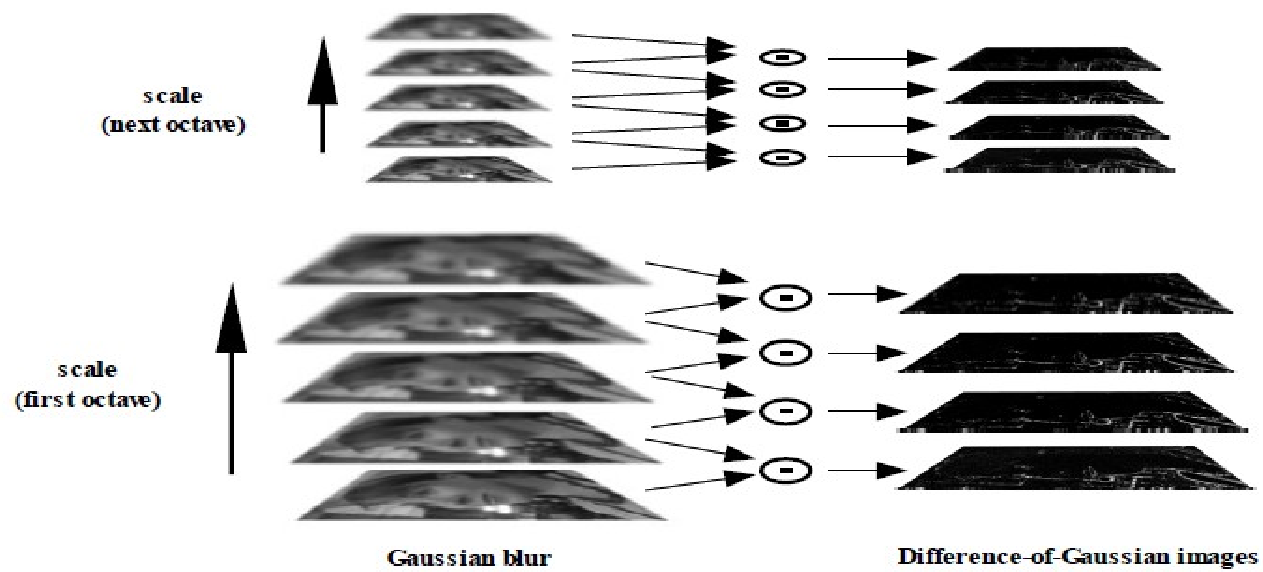
- Four-stage, cascading algorithm:
  1. Scale-space extrema detection
  2. Keypoint localization and filtering
  3. Orientation assignment
  4. Descriptor construction

} **detection** (stages 1–2)

} **description** (stages 3–4)

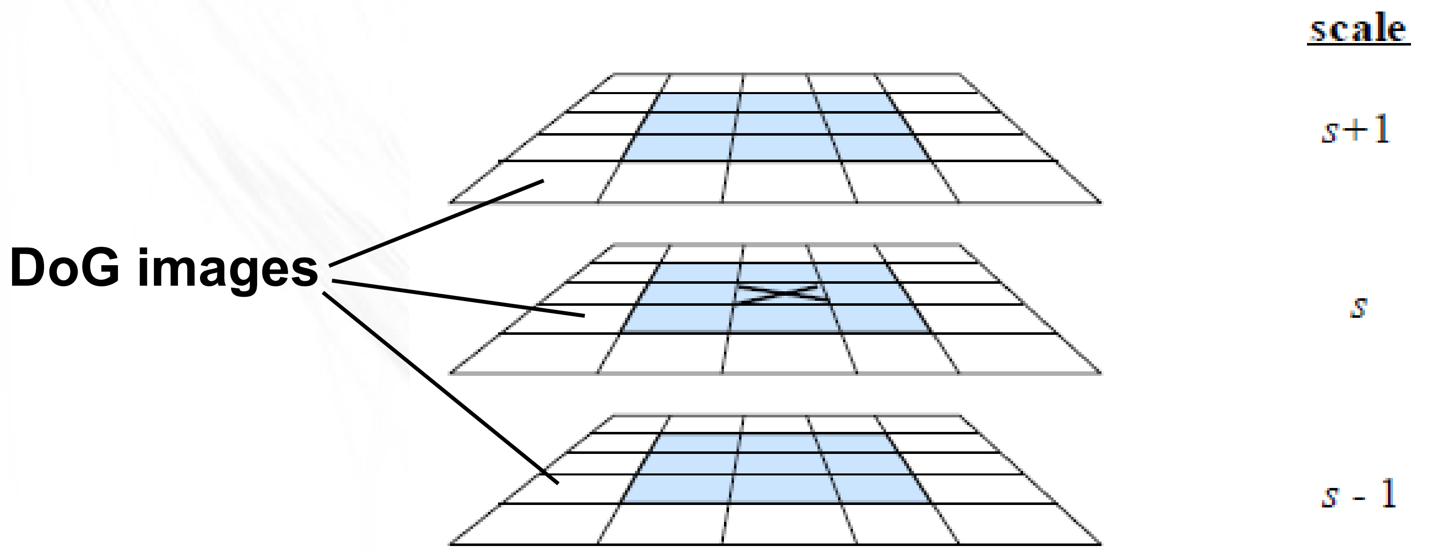
SIFT Stage 1: Scale-space extrema detection

- Scale space:
  - Created by repeatedly convolving an input image with a Gaussian kernel of increasing δ.
  - After every octave, or doubling of δ, the image is downsampled by a factor of two and the blurring iterations are re-started.
  - Adjacent-scale images are subtracted (Difference-of-Gaussians function)



SIFT Stage 1: Scale-space extrema detection (cont'd)

- We then search the DoG images for local minima and maxima to establish our initial feature point locations.
- A DoG pixel needs to be either greater than or less than all pixels in its immediate neighbourhood, as well as all pixels in corresponding neighbourhoods in adjacent DoG images.



SIFT Stage 2: Keypoint Localization and Filtering

- After the first stage, we are left with a set of discrete integer 3D points (x, y, δ) representing physical pixel location and the scale at which they were found.
- These DoG peaks are then fitted to a 3D quadratic function to determine the interpolated sub-pixel (floating point) location and scale of the extreme point
  - Provides a substantial improvement in stability and matching
  - Also used to filter out smaller peaks (low contrast)
- Keypoints along edges are rejected because they have poorly-defined DoG peaks (hard to localize). To do this, we use a ratio of principal curvatures (as in Harris corners).

SIFT Stage 3: Orientation assignment

- We now have feature points in sub-pixel location and scale space
- To achieve rotation invariance, we assign an orientation to each keypoint and we describe the keypoint with respect to this orientation
- We choose the orientation from the most dominant gradient in the local image patch
  - Local gradients contributions are summed into 36 orientation bins (representing 10° increments)
  - Largest bin is chosen as the keypoint's orientation
  - If other bins come within 80% of this peak value, separate keypoints are created with these other orientations

SIFT Stage 4: Descriptor construction

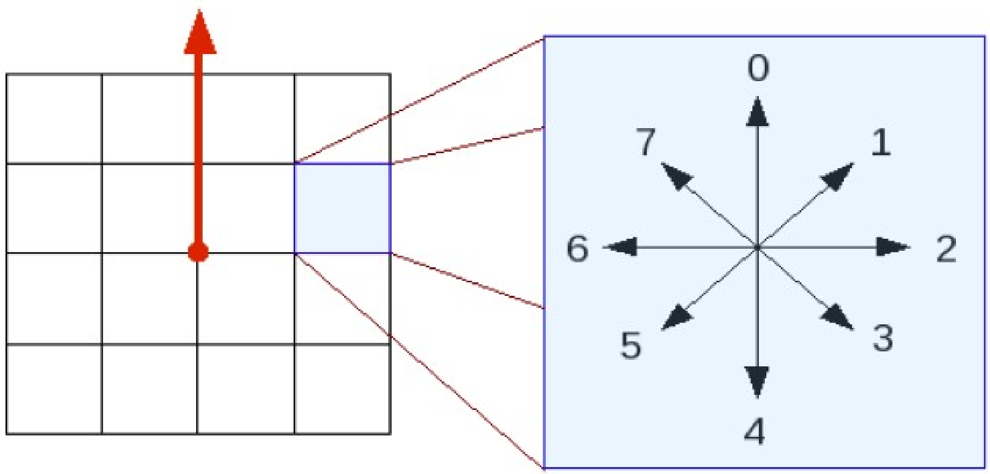
- SIFT keypoint descriptor:
  - 128-byte vector derived from local gradient patch

SIFT Stage 4: Descriptor construction

- Gradient patch is rotated with respect to the keypoint orientation, then divided into 4 x 4 sub-regions, consisting of 16 pixels each

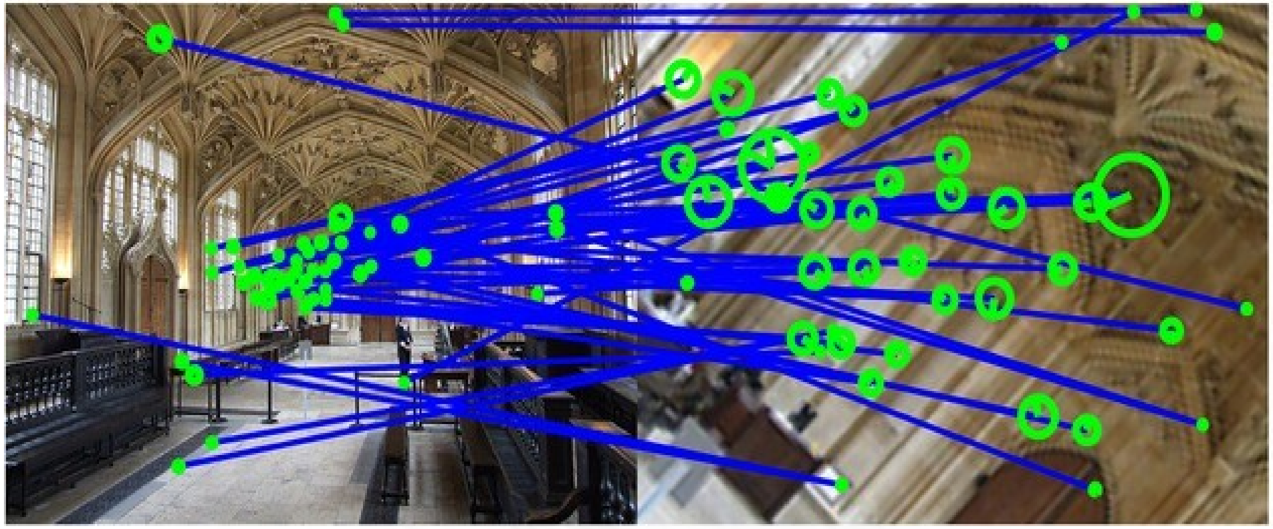
SIFT Stage 4: Descriptor construction (cont'd)

- Each sub-region is characterized by the gradient contributions to an 8-bin orientation histogram
- Concatenation of the 16 orientation histograms creates SIFT's 128-element descriptor vector (16 sub-regions x 8 orientation bins)



SIFT Matching

- Nearest-neighbour search between two feature sets, using Euclidean distance
- Distance ratio test is used to declare a match:
  - If closest vector distance is less than 0.6 * next-closest vector distance, declare the point matched (otherwise declare no match)



SIFT-inspired local detectors/descriptors

- **PCA-SIFT:**
  - Uses the same SIFT-DoG detector
  - Attempts to create a more distinct descriptor by sampling from a larger gradient patch and reducing the descriptor to 36 elements using principal component analysis
  - Achieves improvement in matching speed (36 vs. 128 descriptor elements) without compromising recall/precision performance

- **GLOH (Gradient Location and Orientation Histogram):**
  - Uses SIFT-DoG detector
  - Adds granularity to gradient orientation histogram bins
  - Uses PCA for data compression
  - Achieves improved robustness over SIFT and PCA-SIFT for common image transformation frameworks

SIFT-inspired local detectors/descriptors (cont'd)

- **“Fast Approximated SIFT”:**
  - Uses integral images
  - Substitutes SIFT-DoG detector for a DoM (difference-of-means) to achieve significant speed increase (scale space is approximated by mean, rather than Gaussian, blurring)
  - SIFT localization and post-processing techniques omitted
  - Eightfold speedup in computation is offset by poorer robustness

- **Mahalanobis-SIFT:**
  - SIFT vectors are post-processed to compensate for the relative standard deviations of the individual vector elements (effectively transforming them into Mahalanobis space)
  - Matching performance shows improvement in binary tree structures

SIFT-inspired local detectors/descriptors (cont'd)

- **SURF (Speeded Up Robust Features):**
  - Uses integral images and box filters (rather than a circular window) to approximate Gaussian second-order partial derivatives for scale space
  - 64 descriptor elements instead of 128, with a binary 65th element that effectively represents a cornerness measure and can be used to split the search space in half
  - Finds about 2/3 the number of SIFT features in the same image
  - About 5X faster to compute than SIFT, and about 4X faster matching speed
  - Better than SIFT for some image transformations (i.e. noisy images) but worse for others (i.e. rotation, scaling)

Important Papers

- **Main SIFT journal article:**

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

- **PCA-SIFT:**

  A. Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In Proc. CVPR, pages 506--513, June 2004.

- **GLOH:**

  Mikolajczyk, K. & Schmid, C., A performance evaluation of local descriptors. In Proc. CVPR Int. Conf. on Computer Vision and Pattern Recognition', pp. 257—263, 2003.

- **Fast Approximated SIFT:**

  M. Grabner, H. Grabner, H. Bischof. Fast Approximated SIFT. Asian Conference on Computer Vision, Hyderabad, India, pp. 918-927, 2006.

- **Mahalanobis SIFT:**

  Mikolajczyk and J. G. Matas. Improving descriptors for fast tree matching by optimal linear projection. International Conference on Computer Vision, pp. 1-8, 2007.

- **SURF:**

  H. Bay, T. Tuytelaars, and L. J. V. Gool. SURF: Speeded up robust features. In Proc. Europearn Conf. Computer Vision, pages 404-417, 2006.