SIFT

Scale Invariant Feature Transform

Introduction

- SIFT: Scale-invariant feature transform.
- SIFT method was developed in 1999.
- Its accuracy and robustness were significantly higher than previous template matching methods.
- Its extensions/modifications (PCA-SIFT, ASIFT, SURF, ORB, BRIEF, etc.) are very popular and applied even in the age of deep learning.
  - even if nowadays deep-learning is the most popular;
  - e.g. SuperPoints/SuperGlue.

SIFT was protected by a patent. It was expired in 2020.

Steps of SIFT feature matching

As it is usual, two main steps form the method

1. Feature Detection
   a. ↠ descriptor
   b. Features are represented by a high dimension vector.
2. Feature Matching
   a. by comparing feature vectors
   b. ↠Euclidean distance can be used.

Feature detection

↠ Scale space is applied due to scale invariance.
↠ Rotation invariance is also considered.

Steps:

1. Difference of Gaussian (DoG) operator
    - for detecting ‘good’ features
2. Orientation estimation
    - histogram-based

Scale space

Image are rescaled to several variants

- Here, 100%, 75%, 50%, 25% are visualized

Gaussian Filtering

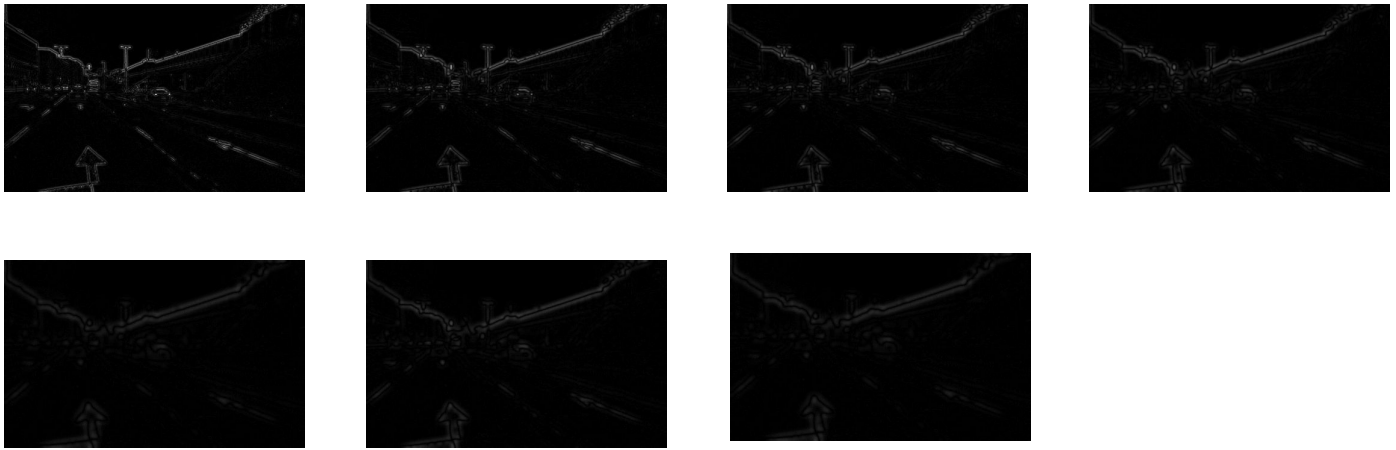
Gaussian filtering with increasing sigma




Difference of Gaussians (DoG)

Laplacian of Gaussian (LoG) can be approximate by a Difference of two Gaussians (DoG) at different scales

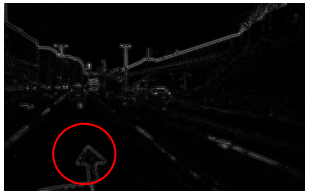
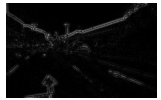
Here are some examples for the differences:



DoGs at different scales

Peak selection

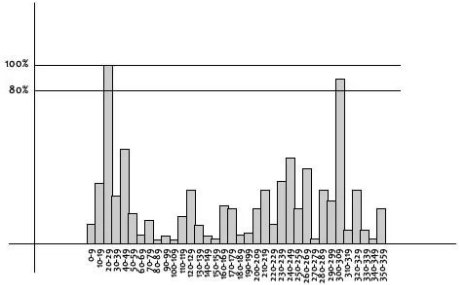


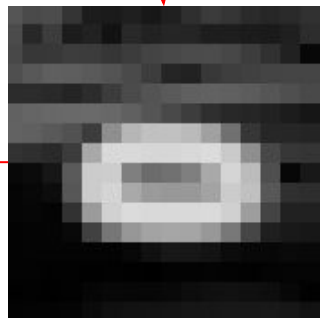



Orientation estimation

The patch around keypoint location selected

- Gradient orientation computed
  - using Prewitt or Sobel operator
- Histogram created from gradient direction
  - 36 bins applied
  - 0˚-9˚, 10˚-19˚,..., 350˚-359˚
- Peak gives orientation of the patch
  - 80% rule: if second peak is close, keypoint deleted.



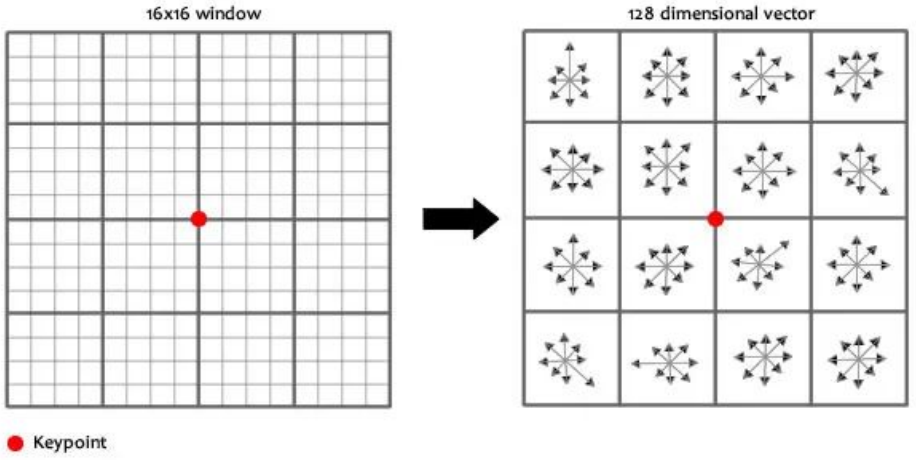


Orientation and scale

- Every feature has now
  - the scale from scale-space
  - dominant orientation
- From now, every feature descriptor is given with respect to the dominant orientation in the processed scale

Keypoint descriptor

- 16x16 pixel patches applied
  - divided into 4x4 patterns
- Histogram for orientation of sub-patterns
  - 8 bins / pattern
- Feature described by 128D vector
  - Containing every histogram value
  - 4x4x8=128



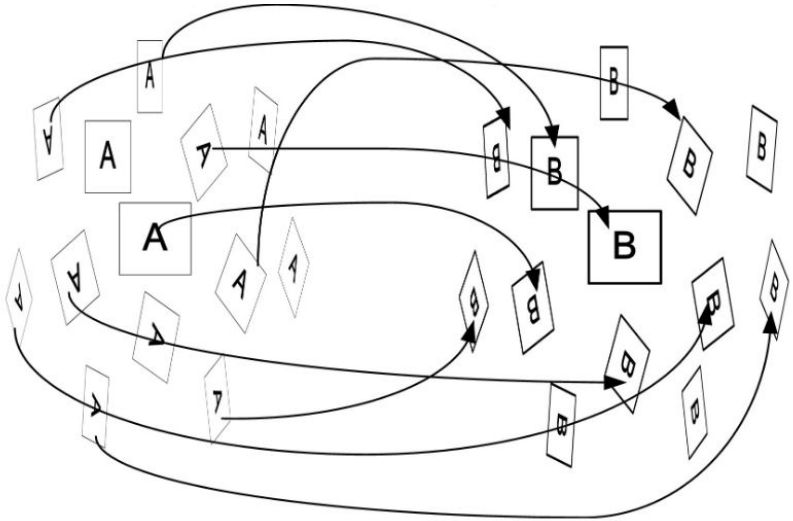
Imag from medium.com

Keypoint matching

- 128D vectors simply compared between images
  - difference between vectors (absolute value)
- 80% rule again
  - score differences for second best target higher than 80% score of best match
  - features discarded

Trivial modifications: Affine SIFT

- Images transformed by several affine transformations (ATs)
- ATs simulate viewpoint difference
  - Then SIFT applied



Guoshen Yu, Jean-Michel Morel: ASIFT: An Algorithm for Fully Affine Invariant Comparison. Image Process on Line (2011)

Example

- Affine transformation between images
  a. ORB features
     - initial scales and angles obtained
  b. brute force template matching
     - varying scale and angles
  c. Lucas-Kanade method for final refinement

Thank you for your attention