Scale and Affine Invariant Interest Point Detectors

Krystian Mikolajczyk and Cordelia Schmid

** Sources: Schmid (CVPR’03), Tuytelaars (ECCV’06).

Why Local Features?
- Robust to noise, occlusion and clutter.
- Distinctive and repeatable.
- No explicit segmentation required – represent objects (classes).
- Invariance to image transformations + robust to illumination changes.
- Applications: SLAM, object (class) recognition, matching…

Some keywords…
- Harris corner detector.
  - Scale sensitive…
- Difference-of-Gaussian (DoG).
  - Lowe’s paper: approx. to normalized LoG.
- Laplacian-of-Gaussian (LoG).
  - Normalized => Extrema in scale-space.
- Related to second moment matrix (SMM): second-order derivates of kernel-convolved image.

The scale-adapted SMM
- Terms: differentiation scale, integration scale, based on variance of kernel.
  \[ \mu(x, \sigma_x, \sigma_y) = \sigma_G^2 \left[ L(x, \sigma_x) L(x, \sigma_y) \right] \]
  \[ \sigma_G = \sqrt{\sigma_x \sigma_y} \]
- Ref: Elimination of edge responses in Lowe’s paper using eigen values…

Characteristic Scale – scale invariance
- Apply local operator at scales: scale where operator best matches local structure.
- LoG better than scale-adapted Harris.

Characteristic scale selection
- Multi-scale Harris.
- Characteristic scale with Laplacian.
Scale invariant feature selection

- Harris-Laplace (HL) detector:
  - Harris measure, 8-neighborhood IP – larger scale ratio.
  - Iterate using LoG until convergence – smaller scale ratio.

- Simplified HL:
  - Reduce scale diff, find IP, keep those with LoG extremum.

- Simplified HL almost as good as HL.

Affine transforms – why?

- Viewpoint changes ~ affine transform.
- Scale changes by different amounts.
- Harris, HL not affine invariant.
- Operate in affine Gaussian scale-space: ellipses as point neighborhoods.

Affine Invariance – linear algebra 101 ☺

- Basis: Anisotropy is affine-transformed isotropy. High-dim search space.
- Constraints on $\Sigma$ of Gaussian kernels:
  - recover affine shape,
  - reduce to orthogonal transform in normalized frames.
- Patterns in normalized frames are isotropic with respect to SMM.
- Estimation of $\Sigma_i, \Sigma_j$ - iterative algorithm.

Affine Invariance – a picture

- Eigen values – yes, again!
- Ratio of eigen values of SMM: eigen values equal=>$ is isotropy.
  \[ Q = \frac{\lambda_{\text{max}}(\mu)}{\lambda_{\text{min}}(\mu)} \]
- Once more, a measure of the skew/stretch.
- Ref: Lowe’s feature rejection based on the $r$-factor.
Algorithm \((\Sigma_1, \Sigma_2)\) – iterate until convergence

- Shape adaptation – normalize window using a function of SMM.
- Select \(\sigma_y\) - remember characteristic scale.
- Select \(\sigma_D\) - equalize eigenvalues.
- Spatial localization of IP (interest point) – Harris detectors.
- Compute SMM and update normalization matrix.

Affine invariant Harris points

- Iterative estimation of localization, scale, neighborhood

Notes

- Convergence based on reasonable choice of scales and initial estimates.
  - Initial estimates of IPs not affine invariant.
- Averaging of similar features.
- Only (20-30)% of initial IPs used.
- Repeatability criterion.
- More robust to large viewpoint changes.
- Smallest number of features found.
- Largest time complexity.
Descriptors and Matching
- Normalized Gaussian gradient descriptors – weak!
  - Cause of matching failure – use SIFT descriptors (ref: Moreels + Perona evaluation)...
- Matching based on Mahalanobis distance and filters.
- Comparable performance under scale changes and localization errors.
- Performance much better under significant viewpoint changes.
- Next, some ‘lab made’ image results 😊

Matches – HarAff, large change in viewpoint
- 33 correct matches

Matches – SIFT, large change in viewpoint
- 12 correct matches – hmm...

Images – difference in feature selection...

Some SIFT matching – good...
SIFT Matching – not so good...

Some other methods – MSER

SIFT Matching – not so good...

Some other methods – IBR
Observations…

- Local features intuitively appealing – *a lot of open questions still*.
- Scale, rotation, affine invariance, robust to viewpoint and illumination changes.
- Depend on *texture* in images – absence of texture can make it unreliable.
- Can add other features – Color? Texture? Structure?
- Can combine with feature-learning approaches?

That’s all folks 😊