Detectors of salient points or regions

Václav Hlaváč

Czech Technical University in Prague
Czech Institute of Informatics, Robotics and Cybernetics
166 36 Prague 6, Jugoslávských partyzánů 1580/3, Czech Republic
http://people.ciirc.cvut.cz/hlavac, vaclav.hlavac@cvut.cz
also Center for Machine Perception, http://cmp.felk.cvut.cz

Courtesy: T. Werner.
Establishing correspondence across views, a motivation

1. **Detection** identifies the interest points or regions.

2. **Description** calculates descriptors (feature vectors) from the local neighborhood of each interest point or region.

3. **Matching** compares feature description pairwise across views, ranks them, and selects the most prominent one (or a few most prominent ones to increase robustness).
The aim

Design a detector/descriptor that finds interest points or interest regions in an image such that:

- There is only a small number of isolated points/detectors detected as compared to entire number of pixels in the image.
- Image semantics is not taken into account.
- The points are reasonably invariant
  - to small geometric changes as affine transformations, e.g. rotation and scale,
  - to small radiometric variations, e.g. a change of illuminations.
  - different sampling and quantization.

Note: The standard detector satisfying these requirements is the Harris corner detector from 1988. However, the idea was around already earlier.
Desired properties of good detections/descriptions

- **Repeatability** – The same detections with similar descriptors can be found in slightly different images despite geometric and photometric distortions.

- **Saliency** – Detections/descriptions are distinctive.

- **Compactness and efficiency** – Significantly less detections/features than image pixels used.

- **Locality** – A descriptor is calculated from a small neighborhood of a detection. It brings robustness to clutter and occlusion.

- **Invariance to small geometric/photometric changes** – enables the use in diverse applications cf. next slide.

Points of interest or salient regions are usually stable across viewpoint or illumination changes and provide a good localization (cf. a line vs. a cross).
Some applications of detections/descriptions

- Image alignment
- 3D reconstruction
- Motion tracking
- Indexing and retrieval in the image database
- Object recognition
- Robot navigation
References


ZOO of useful detectors

Hessian & Harris [Beaudet 1978], [Harris 1988]
Laplacian, DoG [Lindeberg ‘98], [Lowe 1999]
Harris-/Hessian-Laplace [Mikolajczyk & Schmid 2001]
Harris-/Hessian-Affine [Mikolajczyk & Schmid 2004]
EBR and IBR [Tuytelaars & Van Gool 2004]
MSER [Matas et al. 2002]
Salient Regions [Kadir & Brady 2001]
Others . . .
Corner detection, the basic idea pictorially

- The interest point should be detected locally. Consider the analogy as prospecting the image by a small shifting viewing window.

- Slight shifting this viewing window in any direction should give a large change in intensity.

```
“flat” region: no change in all directions

“edge”: no change along the edge direction

“corner”: significant change in all directions
```

Idea: A. Efros
Maximally Stable Extremal Regions (MSERs)
The basic idea

- MSERs (Matas et al. 2002) are regions characterized by almost uniform intensity, surrounded by contrasting background.
- MSERs are constructed by trying multiple thresholds while selecting those regions, which maintain the same area over changing thresholds.

orig. gray scale  |  landscape  |  thresholds  |  MSERs
**Autocorrelation function**

- How similar is the image function \( I(x, y) \) at point \((x, y)\) to itself, when shifted by \((\Delta x, \Delta y)\)?

- This is given by the autocorrelation function

\[
c(x, y; \Delta x, \Delta y) = \sum_{(u,v) \in W(x,y)} w(u,v) \left( I(u,v) - I(u + \Delta x, v + \Delta y) \right)^2
\]

where

- \( W(x, y) \) is a window centered at point \((x, y)\)
- \( w(u, v) \) is either constant or (better) Gaussian

\[
\exp \left( \frac{-(u-x)^2 - (v-y)^2}{2\sigma^2} \right).
\]

(Further on, we will replace \( \sum_{(u,v) \in W(x,y)} w(u,v) \) with \( \sum_{W} \) for simplicity)
Quadratic approximation of the autocorrelation function (1)

Approximate the shifted function by the first-order Taylor expansion:

\[
I(u + \Delta x, v + \Delta y) \approx I(u, v) + I_x(u, v)\Delta x + I_y(u, v)\Delta y
\]

\[
= I(u, v) + [I_x(u, v), I_y(u, v)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix},
\]

where \(I_x, I_y\) are partial derivatives of \(I(x, y)\).
Quadratic approximation
of the autocorrelation function (2)

Autocorrelation function $c(x, y; \Delta x, \Delta y)$:

$$c(x, y; \Delta x, \Delta y) = \sum_W \left( I(u, v) - I(u + \Delta x, v + \Delta y) \right)^2$$

$$\approx \sum_W \left( \begin{bmatrix} I_x(u, v) & I_y(u, v) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right)^2$$

$$= \begin{bmatrix} \Delta x, \Delta y \end{bmatrix} Q(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} ,$$

where $Q(x, y) = \sum_W \begin{bmatrix} I_x(x, y)^2 & I_x(x, y)I_y(x, y) \\ I_x(x, y)I_y(x, y) & I_y(x, y)^2 \end{bmatrix}$

$$= \begin{bmatrix} \sum_W I_x(x, y)^2 & \sum_W I_x(x, y)I_y(x, y) \\ \sum_W I_x(x, y)I_y(x, y) & \sum_W I_y(x, y)^2 \end{bmatrix}$$
The autocorrelation function has been approximated by a quadratic function

\[ c(x, y; \Delta x, \Delta y) \approx [\Delta x, \Delta y] Q(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = [\Delta x, \Delta y] \begin{bmatrix} A & B \\ B & C \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \]

- Elongation and size of the uncertainty ellipse is given by eigenvalues \( \lambda_1, \lambda_2 \) of \( Q(x, y) \)
- The rotation angle of the ellipse is given by eigenvectors of \( Q(x, y) \). We do not need the rotation information here.
- Uncertainty ellipses have the equation \([\Delta x, \Delta y] Q(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = 1\).
Uncertainty ellipses, illustration

- **Flat region**: Both eigenvalues small
- **Edge**: One small, one large
- **Corner**: Both eigenvalues large
How to find isolated feature points?

- Characterize ‘cornerness’ $H(x, y)$ by eigenvalues of $Q(x, y)$:
  - $Q(x, y)$ is symmetric and positive definite $\Rightarrow \lambda_1, \lambda_2 > 0$
  - $\lambda_1 \lambda_2 = \det Q(x, y) = AC - B^2$, $\lambda_1 + \lambda_2 = \text{trace } Q(x, y) = A + C$
  - Harris suggested: Cornerness $H = \lambda_1 \lambda_2 - 0.04(\lambda_1 + \lambda_2)^2$
  - Image $I(x, y)$ and its cornerness $H(x, y)$:

- Find corner points as local maxima of cornerness $H(x, y)$:
  - Local maximum in the image is defined as a point greater than its neighbors (in $3 \times 3$ or even $5 \times 5$ neighborhood)
Harris corners, typical result
(on a larger image)
Harris corners, algorithm summary

- Compute partial derivatives $I_x(x, y), I_y(x, y)$ by finite differences:

\[
I_x(x, y) \approx I(x + 1, y) - I(x, y), \quad I_y(x, y) \approx I(x, y + 1) - I(x, y)
\]

Before this, it is good (but not necessary) to smooth image with Gaussian with $\sigma \sim 1$, to eliminate noise.

- Compute images

\[
A(x, y) = \sum_W I_x(x, y)^2, \quad B(x, y) = \sum_W I_x(x, y) I_y(x, y), \quad C(x, y) = \sum_W I_y(x, y)^2
\]

E.g., image $A(x, y)$ is just the convolution of image $I_x(x, y)^2$ with the Gaussian. Use MATLAB function `conv2`.

- Compute cornerness $H(x, y)$

- Find local maxima in $H(x, y)$. This can be parallelized in MATLAB by shifting the whole image $H(x, y)$ by one pixel left/right/up/down.