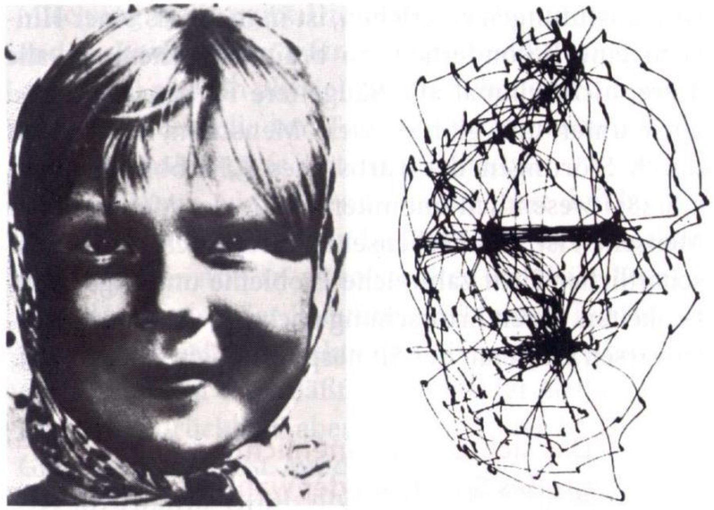
Image Processing

Interest Points

Motivation – Idea

Low-level Vision: **Image → "Image"**

High-level vision: **Image → Description**



From the biology: **Saccades**
Eyes scan the scene → data reduction

Outlines:

1. Interest points: **where** is something interesting in the scene?
2. Image features: **what** is interesting here?
3. Applications: what it can be **used** for?

What would be a “good” detector

1. Should produce a relatively few interest points in order to remove redundant data efficiently

2. Should be invariant against:

   a. Color transformation – additive (lightning change), multiplicative (contrast), linear (both), monotone etc.;

   b. Discretization (e.g. spatial resolution, focus);

   c. Geometric transformation – scaling, rotation, translation, affine transformation, projective transformation etc.

Harris detector

Idea – the question: how similar is the image $I(x, y)$ at a particular position (x, y) to itself if it is shifted by $(\Delta x, \Delta y)$?

Autocorrelation function:

$$c(x, y, \Delta x, \Delta y) = \sum_{(u,v) \in W(x,y)} w(u, v) \Big(I(u, v) - I(u+\Delta x, v+\Delta y) \Big)^2$$

$W(x, y)$ is a small vicinity (window) around (x, y)

$w(u, v)$ is a convolution kernel, used to decrease the influence of pixels far from (x, y), e.g. the Gaussian

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

Harris detector

$$c(x, y, \Delta x, \Delta y) = \sum_{(u,v)\in W(x,y)} w(u, v)\Big(I(u, v) - I(u{+}\Delta x, v{+}\Delta y)\Big)^2$$

One is interested in **properties** of $c(x, y, \Delta x, \Delta y)$ at each position (x, y)

A problem: the image function $I(u{+}\Delta x, v{+}\Delta y)$ as a function of $(\Delta x, \Delta y)$ is an **arbitrary** one.

The way out – linear approximation:

$$\begin{aligned} I(u{+}\Delta x, v{+}\Delta y) &\approx I(u, v) + \frac{\partial I(u, v)}{\partial x}\Delta x + \frac{\partial I(u, v)}{\partial y}\Delta y \\ &= I(u, v) + [I_x(u, v), I_y(u, v)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \end{aligned}$$

with partial directional derivatives $I_x(u, v)$ and $I_y(u, v)$ at (u, v).

Harris detector

Put it together:

$$
\begin{aligned}
c(x, y, \Delta x, \Delta y) &= \sum_{(u,v)\in W(x,y)} \Big(I(u, v) - I(u+\Delta x, v+\Delta y)\Big)^2 \\
&\approx \sum_{(u,v)\in W(x,y)} \left([I_x(u, v), I_y(u, v)] \begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}\right)^2 \\
&= [\Delta x, \Delta y]\, Q(x, y) \begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix}
\end{aligned}
$$

with

$$
Q(x, y) = \begin{bmatrix} \sum_W I_x(u, v)^2 & \sum_W I_x(u, v) I_y(u, v) \\ \sum_W I_x(u, v) I_y(u, v) & \sum_W I_y(u, v)^2 \end{bmatrix} = \begin{bmatrix} A & B \\ B & C \end{bmatrix}
$$

Harris detector

The autocorrelation function

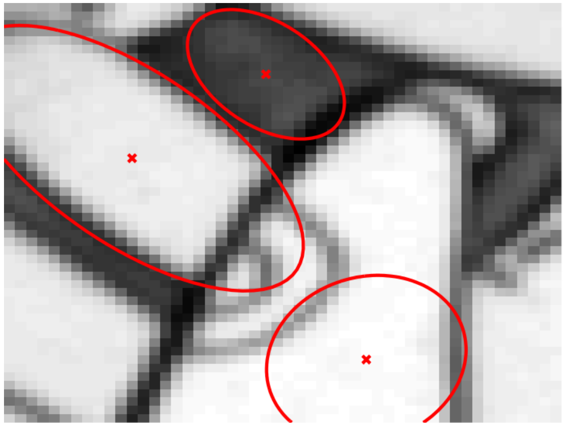
$$c(x, y, \Delta x, \Delta y) = [\Delta x, \Delta y] Q(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

is (now, after approximation) a **quadratic** function in Δx and Δy

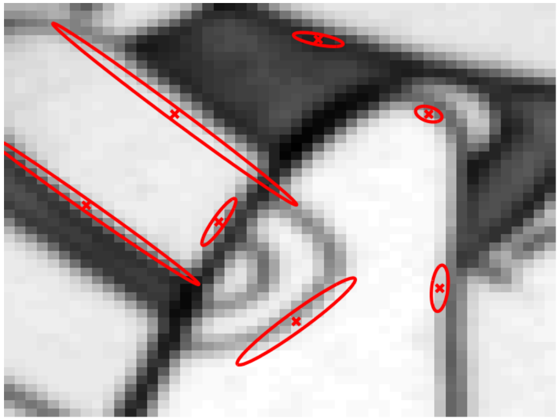
- Isolines are ellipses ($Q(x, y)$ is symmetric and positive definite);
- Eigenvalues $\lambda_1, \lambda_2 > 0$ define prolongations;
- Eigenvectors define orientations (here not relevant, because the detector should be rotationally invariant).

Harris detector

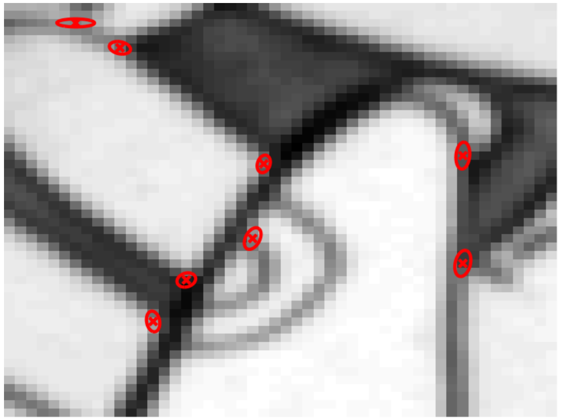
Some examples – isolines for $c(x, y, \Delta x, \Delta y) = 1$:



(a) Flat



(b) Edges



(c) Corners

a. Homogenous regions: both λ-s are small

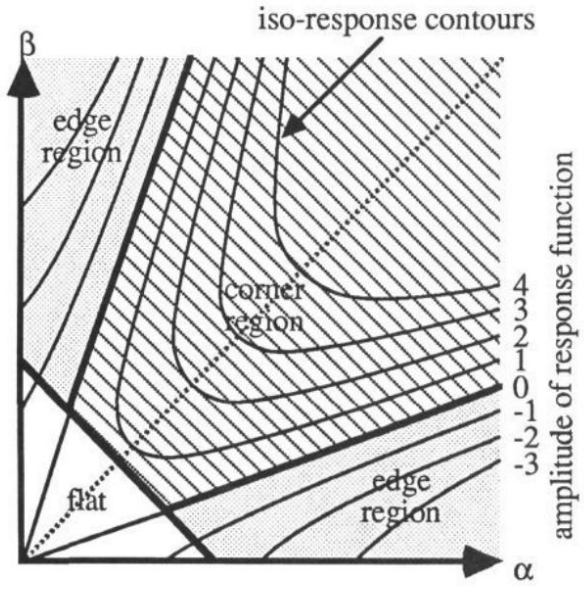
b. Edges: one λ is small the other one is large

c. Corners: both λ-s are large

"**Cornerness**" is a characteristic of $Q(x, y)$

$$\lambda_1 \lambda_2 = \det Q(x, y) = AC - B^2, \quad \lambda_1 + \lambda_2 = \mathrm{trace}\, Q(x, y) = A + C$$

Proposition by Harris: $H = \lambda_1 \lambda_2 - 0.04(\lambda_1 + \lambda_2)^2$



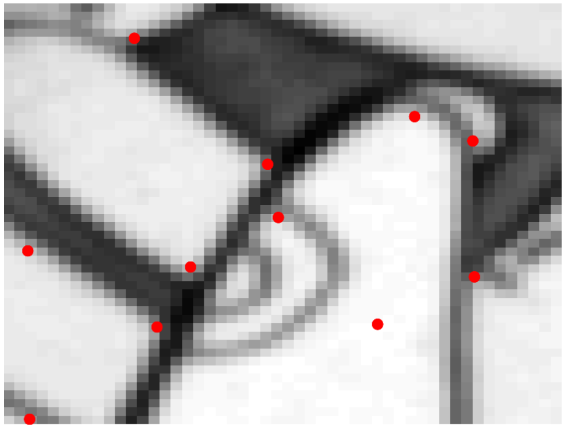
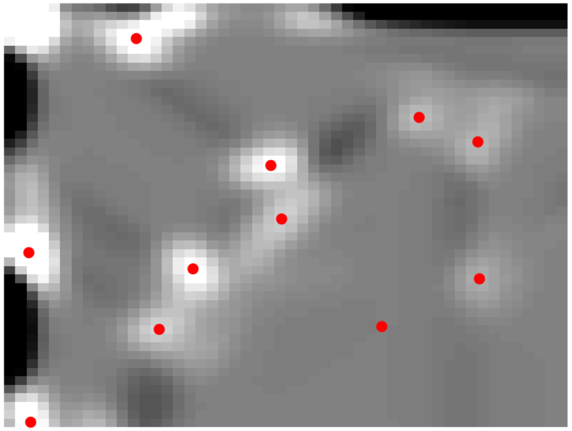


Image $I(x, y)$



Cornerness $H(x, y)$

Interest points are **local maxima** of the cornerness.

Harris detector

Harris detector, a naïve algorithm (top-down)

Search for local maxima:

```
for ( alle Pixel p )
    current = computeH(p);
    flag = true;
    for ( alle p′ ∈ W′(p) )
        if ( computeH(p′) > current ) flag = false;
    if ( flag ) „p ist besonders";
```

Computation of the cornerness:

computeH(p)

for (alle $p' \in W(p)$)

$A+ = I_x(p)^2$;

$C+ = I_y(p)^2$;

$B+ = I_x(p) I_x(p)$;

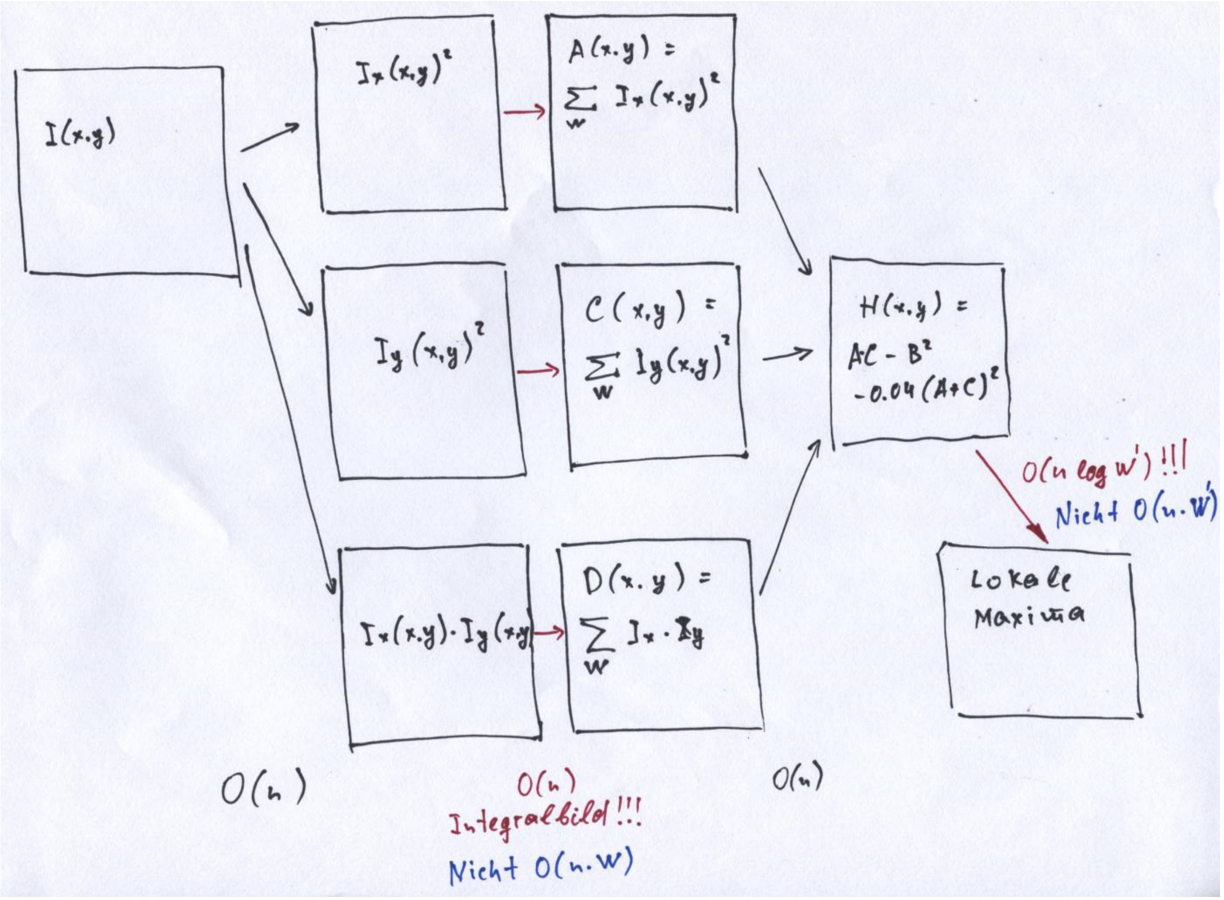
return $AC - B^2 - 0.04(A + C)$

Time complexity:

$O(n \cdot W \cdot W')$

(very bad ☹)

Harris detector, a better algorithm



- Compute nothing twice
- “Integral image”-approach for summations
- Special data structures for local maxima

→ time complexity: $O(n \log W')$

Maximally stable extremal regions (MSER)

Pre-requisites:

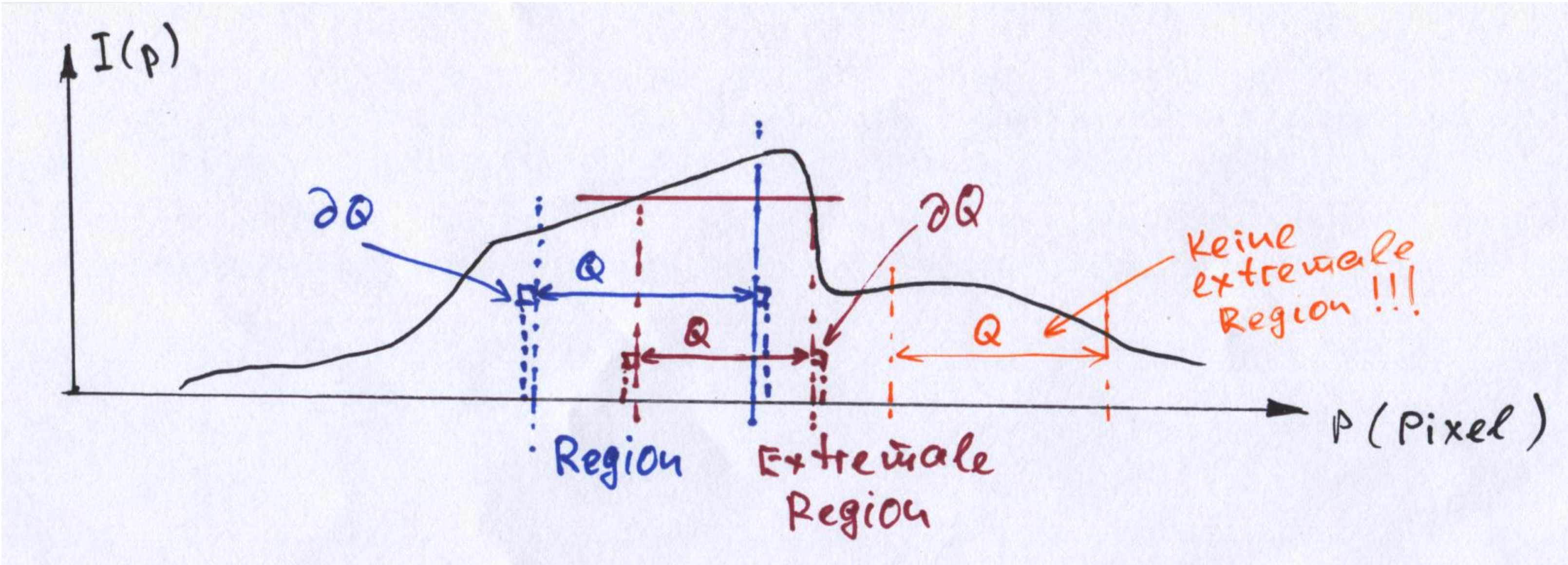
Image is a mapping $I : D \subset \mathbb{Z}^2 \to S$

S is a **fully ordered** set, (e.g. gray-values $\{0 \ldots 255\}$ or $S = \mathbb{R}$)

There is a **neighborhood relation** $A \subset D \times D$,
for example 4-neighborhood, i.e. $pAq \Leftrightarrow |p_x - q_x| + |p_y - q_y| \leq 1$

Otherwise MSER-s can not be defined.

Maximally stable extremal regions



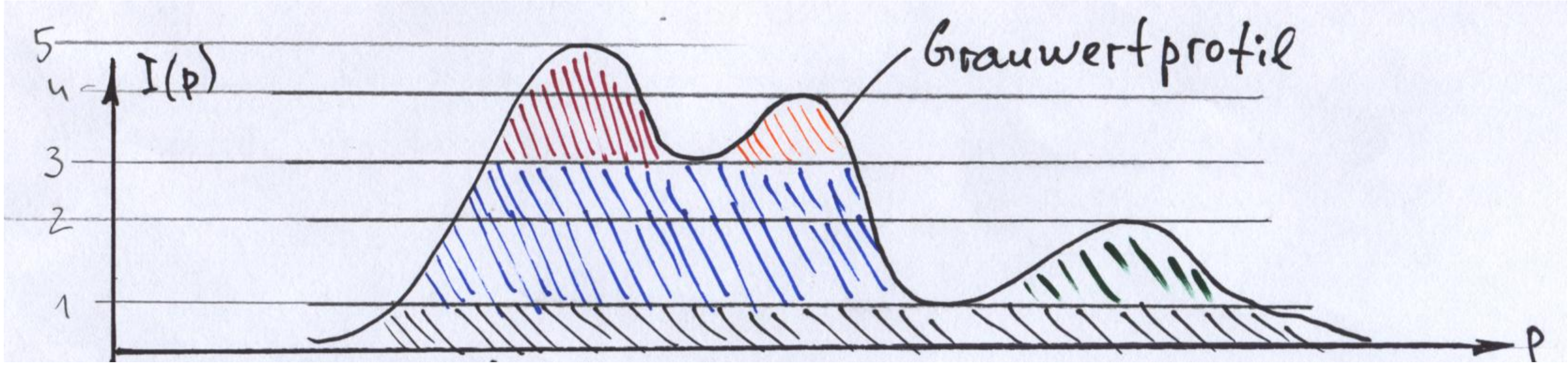
A **region** Q is a connected component of D , i.e. for any pair $p, q \in Q$ there is a path $p, a_1, a_2 \ldots a_n, q$ so, that $pAa_1,\ a_iAa_{i+1},\ a_nAq$ holds.

The (outer) **border** ∂Q is a subset of $D \setminus Q$ so, that for any pixel $q \in \partial Q$ there is at least on pixel $p \in Q$ with pAq .

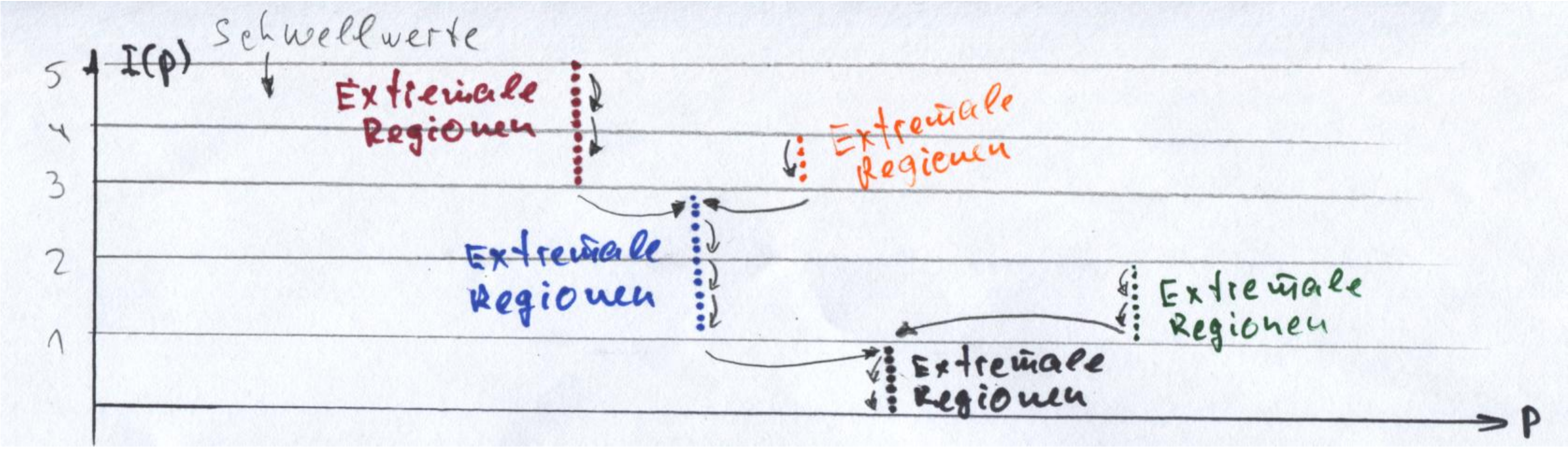
A region is **extremal** if $I(p) > I(q)$ holds for all $p \in Q,\ q \in \partial Q$.

Maximally stable extremal regions

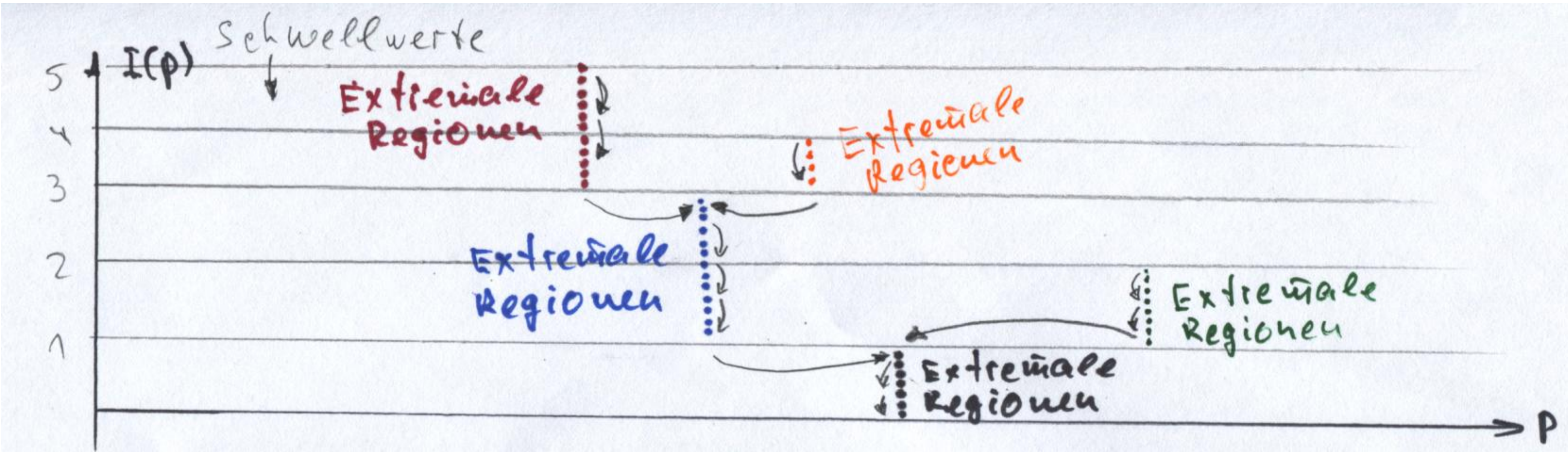
Extremale regions are connected components in the binarized image:



The set of all extremal regions composes a tree-like structure:



Maximally stable extremal regions



Let $Q_1 \subset Q_2 \subset \ldots Q_i \ldots$ be a sequence of nested extremal regions.

An extremal region Q_{i^*} is **maximally stable** if the stability function

$$q(i) = |Q_{i+\Delta} \setminus Q_{i-\Delta}| / |Q_i|$$

has its local minimum at i^*.
($|\cdot|$ is the cardinality, $\Delta \in S$ is a free parameter).

Maximally stable extremal regions

- Invariant to affine transformation of gray-values
- Co-variant to elastic transformation of the domain
- Both small and large structures are detected

Maximally stable extremal regions

A naïve algorithm:

```
for ( alle Schwellwerte )
    Binarisiere das Bild
    Erzeuge alle Zusammenhangskomponente
Baue den Baum auf
Verfolge alle Pfade vom Wurzel zu Blätter, finde MSER-s
```

Time complexity: $O(|S|n)$

Maximally stable extremal regions

A better algorithm:

1. `Sortiere Pixel nach Grauwerten`
2. `Platziere die Pixel der Reihenfolge nach ins Bild, aktualisiere Zusammenhangskomponente und den Baum durch Vereinigung der Teilmengen.`

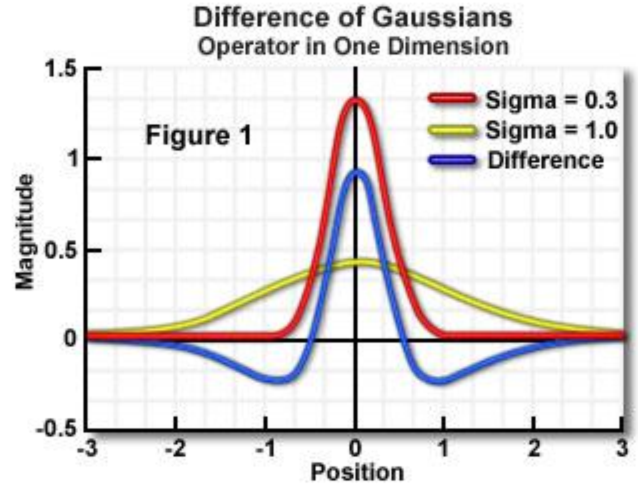
Time complexities:

1. $O(n)$ by BINSORT;
2. $O(n \log \log n)$ by the "Union-find" algorithm.

0.14 seconds on a Linux PC with Athlon XP 1600+ for a 530x350 image

Difference of Gaussians

Convolution with →




Original


DoG


Threshold

→ Edge detection

Literature

- Chris Harris & Mike Stephens: A Combined Corner and Edge Detector (1988)
- J. Matas, O. Chum, M. Urban, T. Pajdla: Robust Wide Baseline Stereo from Maximally Stable Extremal Regions (BMVC 2002)
- K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir: A Comparison of Affine Region Detectors (IJCV 2006)

There is a lot of others interest point detectors …