Institute of Informatics
Eötvös Loránd University
Budapest, Hungary

Basic Algorithms for Digital Image Analysis:

a course

Dmitrij Csetverikov

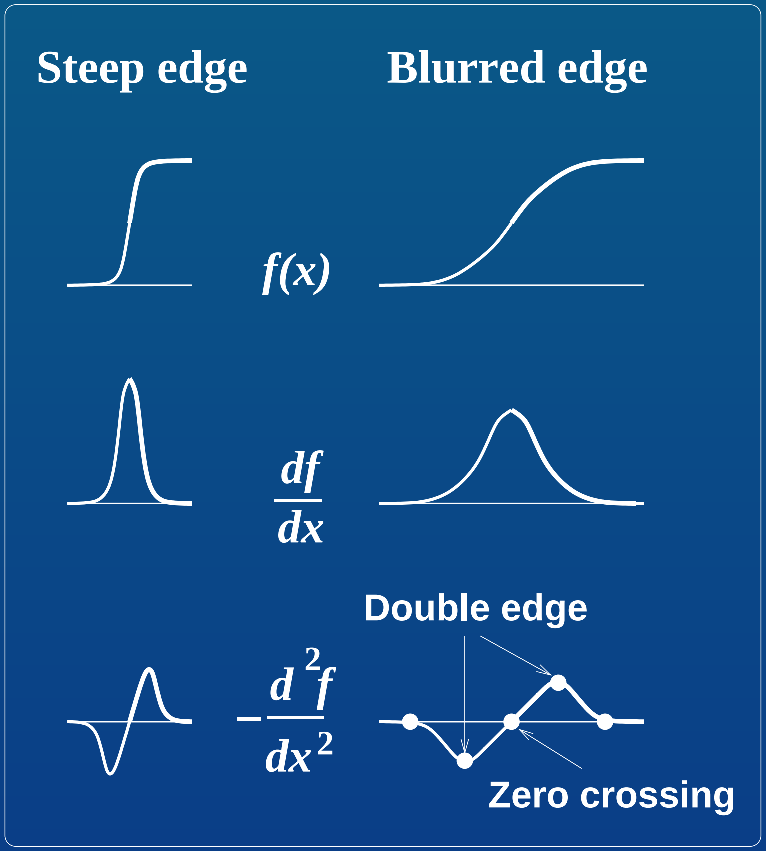
with help of Attila Lerch, Judit Verestóy, Zoltán Megyesi, Zsolt Jankó and Levente Hajder

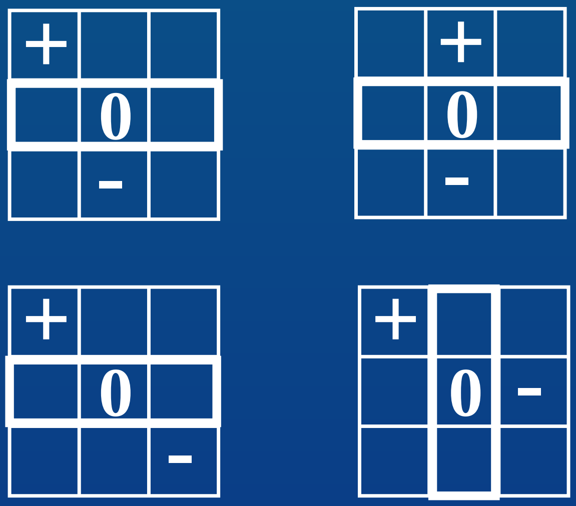
http://visual.ipan.sztaki.hu

Lecture 8: Corner detection

- Zero-crossing edge detector
- Summary of edge detection
- Corner detection in greyscale images
- The local structure matrix
- The KLT corner detector
- The Harris corner detector
- Comparison of the two corner detectors

Zero-crossing edge detector





Left: Principles of zero-crossing edge detector.
Right: Simple masks for detection of zero-crossings.

Implementation of the zero crossing filter: Gaussian smoothing followed by Laplacian filtering.

- Using commutativity and associativity of linear filters and rotation symmetry of Gaussian filter, we obtain the convolution mask of the zero-crossing operator, called Laplacian-of-Gaussian (LoG):

$$w_Z(r) = C\left(\frac{r^2}{\sigma^2} - 1\right)\exp\left\{\frac{-r^2}{2\sigma^2}\right\}$$

  - C: normalisation constant
  - $r^2 = x^2 + y^2$: square distance from centre of mask
  - σ is scale parameter: the smaller the σ the finer the edges obtained

- Discrete zero-crossing mask: Threshold $w_Z(r)$ at a small level.

$\Rightarrow$ Larger mask obtained for larger σ: For example, when $\sigma = 4$ the size of the mask is about 40 pixels.

- Another, more efficient but approximate, implementation of the zero-crossing filter is the difference of two separable Gaussian filters, called DoG.

- Localising the zero-crossings corresponds to edge localisation in gradient-type edge detectors.

  - For more precise localisation, one can locally approximate output of LoG filter by facets (flat patches), then find zero-crossings analytically.

LoG absolute

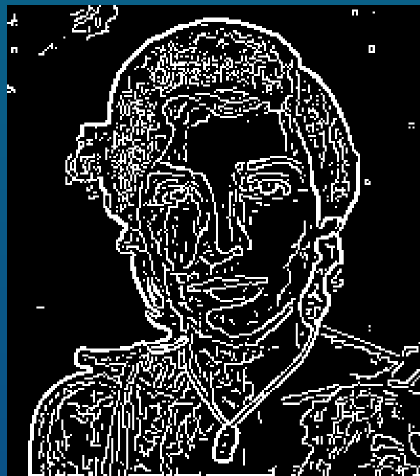
LoG zero

DoG zero

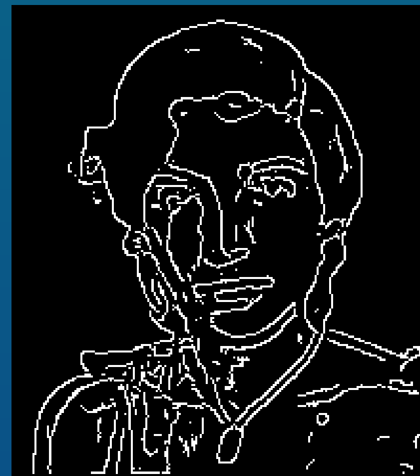
Examples of edge detection by 15×15 *LoG and DoG operators. 'LoG absolute' is absolute value of filter output: dark lines are contours. 'LoG zero' was obtained with removal of weak edges, 'DoG zero' without removal.*

Properties of zero crossing edge detector

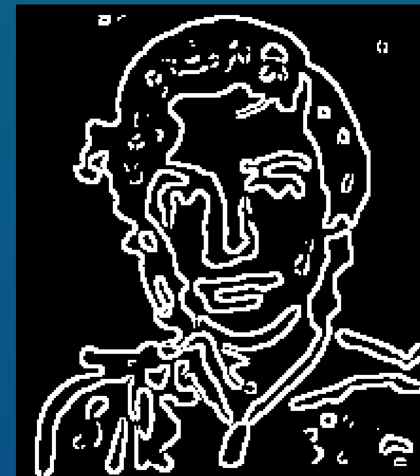
- The continuous zero-crossing edge detector always gives closed contours.
  - Reason: Cross-sections of continuous surface at zero level
  - In principle, this may help in contour following
  - In practice, many spurious loops appear
- Controlled operator size $\sigma \Rightarrow$ Natural edge hierarchy within a scale-space.
  - Edges may only merge or disappear at rougher scales (larger σ)
  - This tree-like data structure facilitates structural analysis of image
- Does not provide edge orientation.
  - Non-maxima suppression and hysteresis threshoding are not applicable
  - Other ways of post-processing to remove unrealiable edges can be used



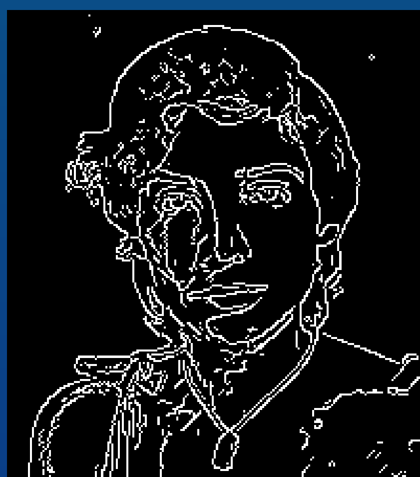
Prewitt 3×3



Mérő-Vassy 7×7



LoG 21×21



Canny 3×3

Canny 7×7

Canny 25×25

Examples of edge detection by different operators. The LoG result was obtained with removal of weak edges. Mérő-Vassy is a non-gradient edge detector.

Summary of edge detection

- 3×3 gradient operators (Prewitt, Sobel) are simple and fast. Used when
  - Fine edges are only needed
  - Noise level is low
- By varying the σ parameter, the Canny operator can be used
  - to detect fine as well as rough edges
  - at different noise levels
- All gradient operators
  - Provide edge orientation
  - Need localisation: non-maxima suppression, hysteresis threshoding
- The zero-crossing edge detector
  - Is supported by neurophysiological experiments
  - Was popular in the 1980's
  - Today, less frequently used in practice

Corner detection in greyscale images

A reminder:

- Corners are used in shape analysis and motion analysis
  - Motion is ambiguous at an edge, unambiguous at a corner
  - Shapes can be approximately reconstructed from their corners
- Two different operations although related operations are called corner detection:
  - Detection of corners in greyscale images
    - does not assume extracted contours
  - Detection of corners in digital curves
    - assumes extracted contours

This lecture deals with corner detection in greyscale images. Corner detection in contours will be discussed later.

Corners, edges, and derivatives of intensity function

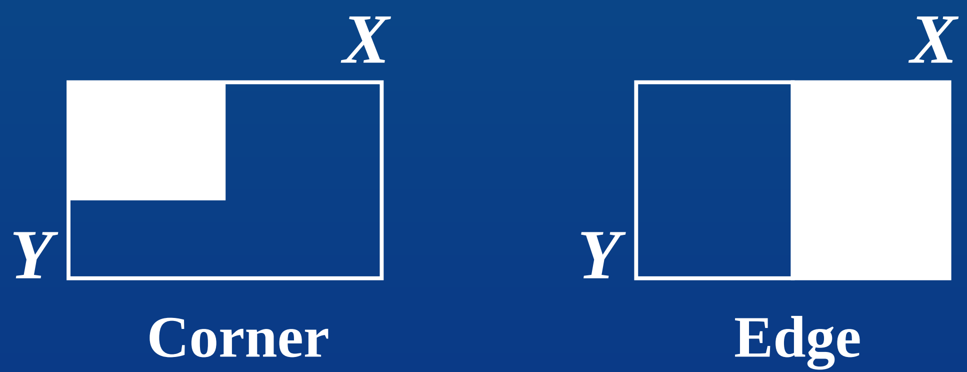
Diference between greyscale corners and edges:

- Corners are local image features characterised by locations where variations of intensity function $f(x, y)$ in both X and Y directions are high.

$\Rightarrow$ Both partial derivatives f_x and f_y are large

- Edges are locations where the variation of $f(x, y)$ in a certain direction is high, while the variation in the orthogonal direction is low.

$\Rightarrow$ In an edge oriented along the Y axis, f_x is large, while is f_y small



A corner and an edge.

Two selected corner detectors

Different corner detectors exist, but we will only consider two of them:

- The Kanade-Lucas-Tomasi (KLT) operator
- The Harris operator

Reasons:

- Most frequently used: Harris in Europe, KLT in US.
- Can select corners and other interest points.
- Have many application areas, for example:
  - motion tracking, stereo matching, image database retrieval
- Are relatively simple but still efficient and reliable.

The two operators are closely related and based on the local structure matrix.

The local structure matrix C_{str}

Definition of the local structure matrix (tensor):

$$C_{str} = w_G(r;\sigma) * \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix} \tag{1}$$

Explanation of the definition:

- The derivatives of the intensity function $f(x, y)$ are first calculated in each point.
  - If necessary, the image is smoothed before taking the derivatives
- Then, the entries of the matrix (f_x^2, etc.) are obtained.
- Finally, each of the entries is smoothed (integrated) by Gaussian filter $w_G(r;\sigma)$ of selected size σ.
  - Often, a simple box (averaging) filter is used instead of the Gaussian.

Properties of the local structure matrix

Denoting in (1) the smoothing by $\widehat{ff}$, we have

$$C_{str} = \begin{bmatrix} \widehat{f_x^2} & \widehat{f_x f_y} \\ \widehat{f_x f_y} & \widehat{f_y^2} \end{bmatrix}$$

The local structure matrix C_{str} is

- Symmetric

⇒ It can be diagonalised by rotation of the coordinate axes. The diagonal entries will be the two eigenvalues λ_1 and λ_2:

$$C_{str} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

- Positive definite

⇒ The eigenvalues are nonnegative. Assume $\lambda_1 \geq \lambda_2 \geq 0$.

The meaning of the eigenvalues of C_{str}

The geometric interpretation of λ_1 and λ_2:

- For a perfectly uniform image: $C_{str} = 0$ and $\lambda_1 = \lambda_2 = 0$.
- For a perfectly black-and-white step edge: $\lambda_1 > 0$, $\lambda_2 = 0$, where the eigenvector associated with λ_1 is orthogonal to the edge.
- For a corner of black square against a white background: $\lambda_1 \geq \lambda_2 > 0$.
  - The higher the contrast in that direction, the larger the eigenvalue

Basic observations:

- The eigenvectors encode edge directions, the eigenvectors edge magnitudes.
- A corner is identified by two strong edges $\Rightarrow$ A corner is a location where the smaller eigenvalue, λ_2, is large enough.

The KLT corner detector has two parameters: the threshold on λ_2, denoted by λ_{thr}, and the linear size of a square window (neighbourhood) D.

Algorithm 1: The KLT Corner Detector

---

1. Compute f_x and f_y over the entire image $f(x, y)$.

2. For each image point p:

   (a) form the matrix C_{str} over a $D \times D$ neighbourhood of p;
   (b) compute λ_2, the smaller eigenvalue of C_{str};
   (c) if $\lambda_2 > \lambda_{thr}$, save the p into a list, L.

3. Sort L in decreasing order of λ_2.

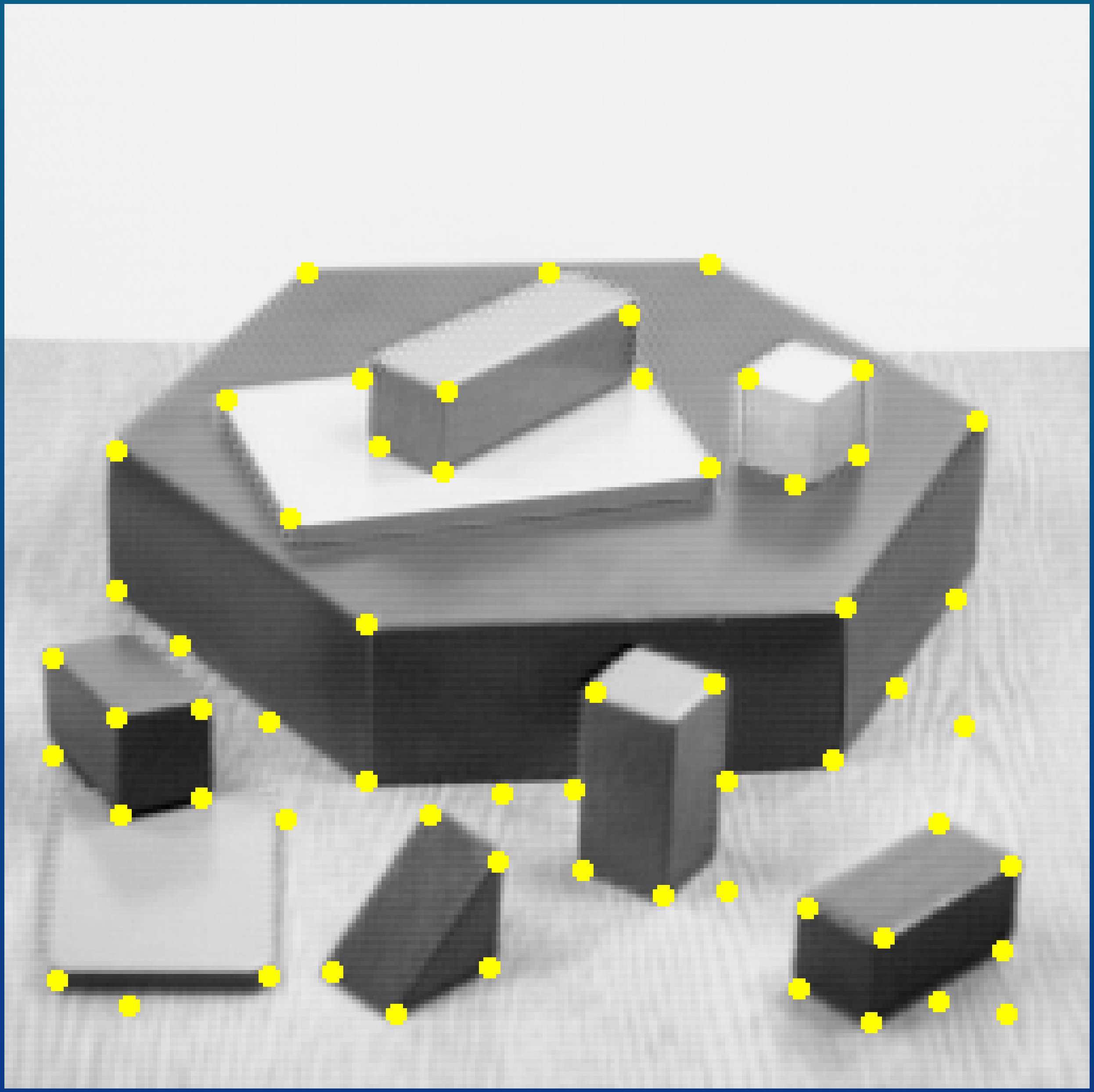
4. Scan the sorted list from top to bottom. For each current point, p, delete all points apperaring further in the list which belong to the neighbourhood of p.

The output is a list of feature points with the following properties:

- In these points, $\lambda_2 > \lambda_{thr}$.
- The D-neighbourhoods of these points do not overlap.

Selection of the parameters λ_{thr} and D:

- The threshold λ_{thr} can be estimated from the histogram of λ_2: usually, there is an obvious valley near zero.
  - Unfortunately, such valley is not always present
- There is no simple criterion for the window size D. Values between 2 and 10 are adequate in most practical cases.
  - For large D, the detected corner tends to move away from its actual position
  - Some corners which are close to each other may be lost



Example of corner detection by the KLT operator.

The Harris corner detector

The Harris corner detector (1988) appeared earlier than KLT. KLT is a different interpretation of the original Harris idea.

Harris defined a measure of corner strength:

$$H(x, y) = \det C_{str} - \alpha \left(\operatorname{trace} C_{str}\right)^2 ,$$

where α is a parameter and $H \geq 0$ if $0 \leq \alpha \leq 0.25$.

A corner is detected when

$$H(x, y) > H_{thr},$$

where H_{thr} is another parameter, a threshold on corner strength.

Similar to the KLT, the Harris corner detector uses D-neighbourhoods to discard weak corners in the neighbourhood of a strong corner.

Parameter of Harris operator and relation to KLT

Assume as before that $\lambda_1 \geq \lambda_2 \geq 0$. Introduce $\lambda_1 = \lambda$, $\lambda_2 = \kappa\lambda$, $0 \leq \kappa \leq 1$.

Using the relations between eigenvalues, determinant and trace of a matrix A

$$\det A = \prod_i \lambda_i$$

$$\text{trace}\, A = \sum_i \lambda_i,$$

we obtain that

$$H = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2 = \lambda^2 \left(\kappa - \alpha(1+\kappa)^2\right)$$

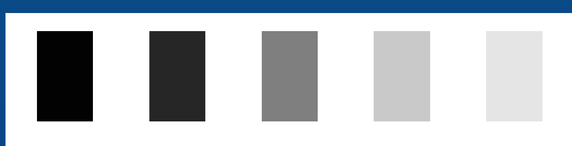
Assuming that $H \geq 0$, we have

$$0 \leq \alpha \leq \frac{k}{(1+\kappa)^2} \leq 0.25 \quad \text{and, for small } \kappa,\ H \approx \lambda^2 \left(\kappa - \alpha\right),\ \alpha \lesssim \kappa$$

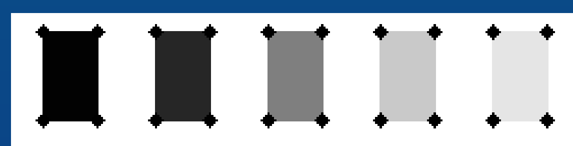
In the Harris operator, α plays a role similar to that of λ_{thr} in the KLT operator.

- Larger $\alpha \Rightarrow$ smaller $H \Rightarrow$ less sensitive detector: less corners detected.
- Smaller $\alpha \Rightarrow$ larger $H \Rightarrow$ more sensitive detector: more corners detected.

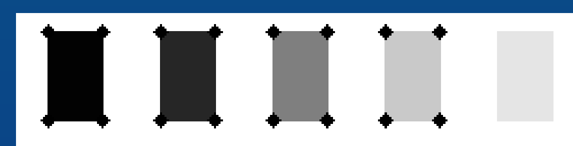
Usually, H_{thr} is set close to zero and fixed, while α is a variable parameter.



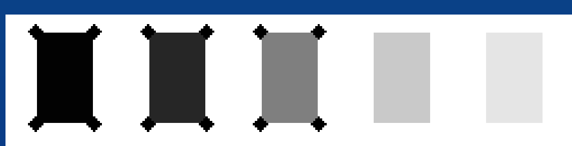
original image



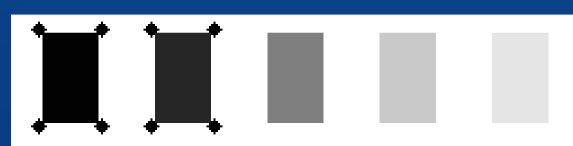
$\alpha = 0.05$



$\alpha = 0.10$



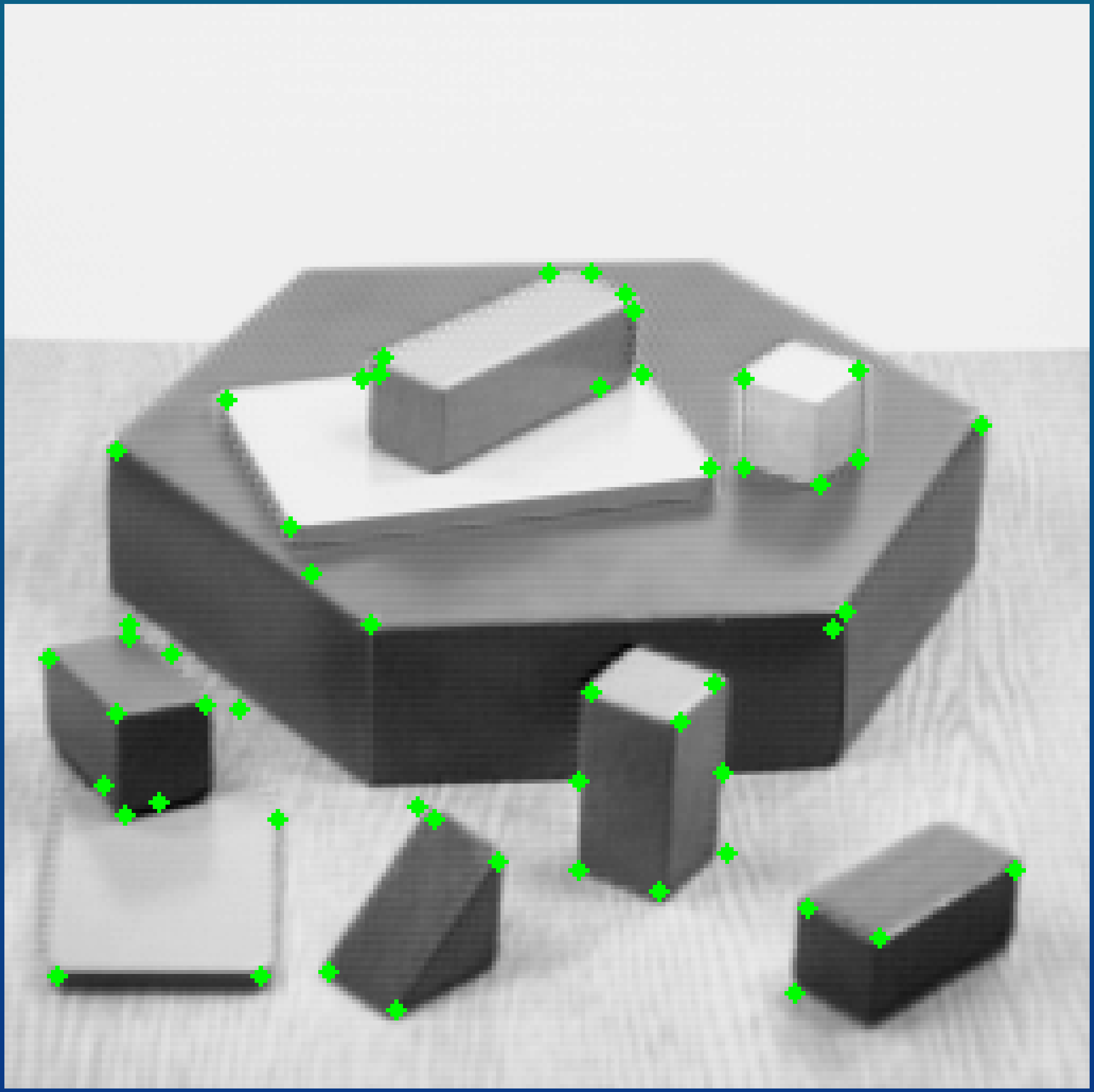
$\alpha = 0.20$



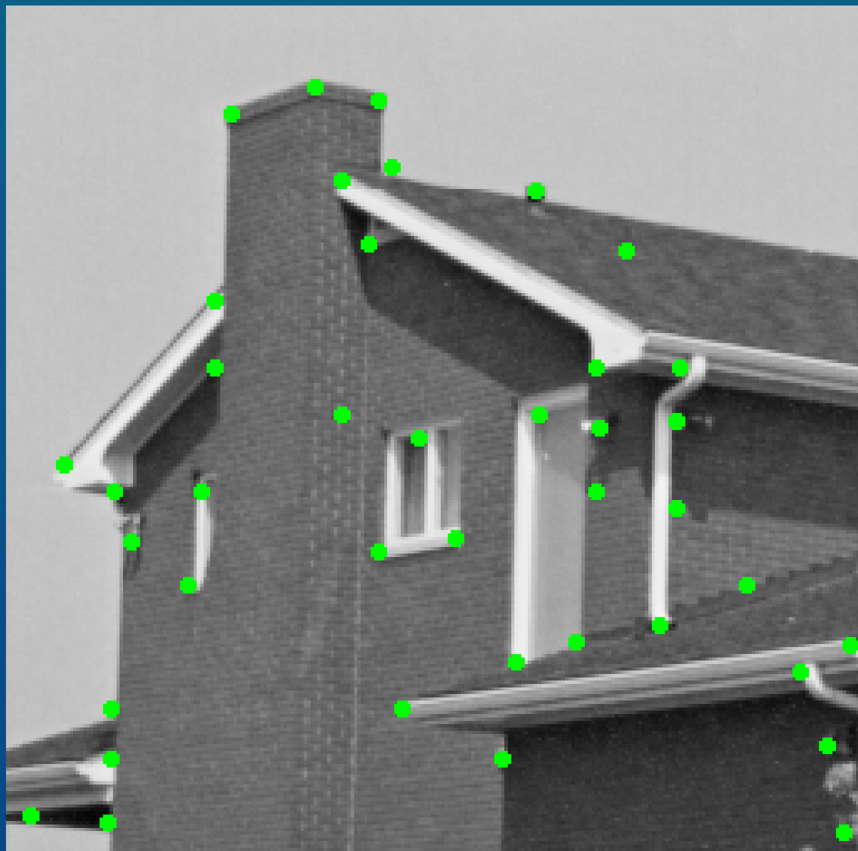
$\alpha = 0.22$

$\alpha = 0.24$

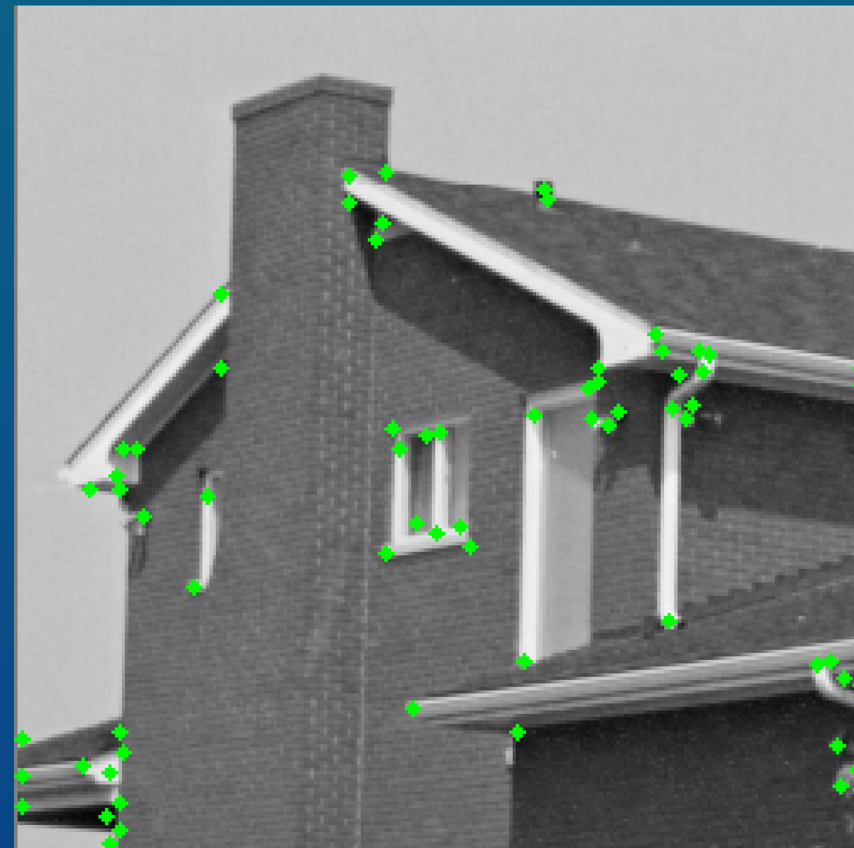
Corner detection by Harris operator: influence of α. ($H_{thr} = 0$.)



Example of corner detection by the Harris operator.



KLT 40 corners



Harris $\alpha = 0.2$

Comparison of the two operators.

Summary of corner detection

- The KLT and the Harris corner detectors are conceptually related.
  - Based on local structure matrix C_{str}
  - Search for points where variations in two orthogonal directions are large
- Difference between the two detectors:
  - KLT sets explicit threshold on the diagonalised C_{str}
  - Harris sets implicit threshold via corner magnitude $H(x, y)$
- The KLT detector
  - usually gives results which are closer to human perception of corners;
  - is often used for motion tracking in the wide-spread KLT Tracker.
- The Harris detector
  - provides good repeatability under varying rotation and illumination;
  - if often used in stereo matching and image database retrieval.
- Both operators may detect interest points other than corners.