Tutorial: Harris corner detector

The aim of this tutorial is to develop a simple Harris corner detector. This is the first step in pattern matching, generally followed by a feature descriptor construction, and a matching process.

1 Corner detector and cornerness measure

Use imgradientxy and imgaussfilt with a scale parameter σ that will constrain the size of the window W.

Use the sobel and gaussian_filter from the scipy.ndimage module.

1.1 Gradient evaluation

The Harris corner detector is based on the gradients of the image, I_x and I_y in x and y directions, respectively.

Apply a Sobel gradient in both directions in order to compute I_x and I_y.

1.2 Structure tensor

The structure tensor is defined by the following matrix. The coefficients ω follow a gaussian law, and each summation represents a gaussian filtering process. W is an operating window.

$$M = \begin{bmatrix} \sum\limits_{(u,v)\in W} \omega(u,v) I_x(u,v)^2 & \sum\limits_{(u,v)\in W} \omega(u,v) I_x(u,v) I_y(u,v) \\ \sum\limits_{(u,v)\in W} \omega(u,v) I_x(u,v) I_y(u,v) & \sum\limits_{(u,v)\in W} \omega(u,v) I_y(u,v)^2 \end{bmatrix} = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$$

- Evaluate M_1 to M_4 for each pixel of the image.

1.3 Cornerness measure

The cornerness measure C, as proposed by Harris and Stephens, is defined as follows for every pixel of coordinates (x, y):

$$C(x, y) = \det(M) - K\mathrm{trace}(M)^2$$

with K between 0.04 and 0.15.

?

Compute C for all pixels and display it for several scales σ.

2 Corners detection

A so-called Harris corner is the result of keeping only local maxima above a certain threshold value. You can use the checkerboard image for testing, or load the sweden road sign image Fig.1.

```
I = imread('sweden_road.png');
```

Use the following function to generate a checkerboard pattern.

```
def checkerboard(nb_x=2, nb_y=2, s=10):
    """
    checkerboard generation
    a grid of size 2*nb_x X 2*nb_y is generated
    each square has s pixels.
    """
    C = 255*np.kron([[1, 0] * nb_x, [0, 1] * nb_x] * nb_y, np.ones((s, s)
        ↪ ))
    return C
```

?

- Evaluate the extended maxima of the image.
- Only the strongest values of the cornerness measure should be kept. Two strategies can be employed in conjonction:
    - Use a threshold value t on C: the choice of this value is not trivial, and it strongly depends on the considered image. An adaptive method would be preferred.

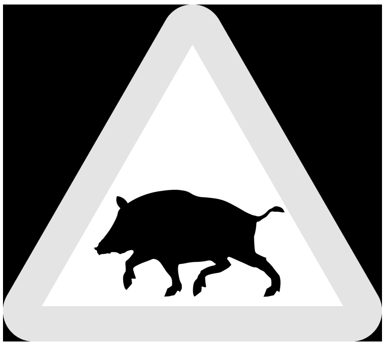

Figure 1: Sweden road sign to be used for corner detection.

  - Keep only the n strongest values.

- The previous operations are affected by the borders of the image. Thus, eliminate the corner points near the borders.
- The detected corners may contain several pixels. Keep only the centroid of each cluster.