Prof. M. Pollefeyes

Filippo Ficarra: Lab 2 - Feature extraction and Optical flow

fficarra@student.ethz.ch, 22–938–062.

06/10/2023 - 16:37h

1 Introduction

In this lab we aim to extract features from images, such as corners, using Harris detection and match descriptors between two different photos of the same thing.

2 Harris corner detection

The idea of Harris corner detection is to analyze the change of intensity of pixels in a window. Using this idea we can basically identify three different regions:

- flat: there is no change of intensity in all the directions
- edge: there is no change of intensity in the direction of the edge
- corner: large change of intensity in all the directions

To perform this analysis we define a window $\boldsymbol{W}$ and we define the SSD error as:

$$E(u,v) = \sum_{(x,y)\in\boldsymbol{W}} \left[I(x+u,y+v) - I(x,y)\right]^2$$

using Taylor approximation we can approximate the error with:

$$E(\Delta x, \Delta v) \approx [\Delta x, \Delta y]\, M \begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix},\ M = \sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2\end{bmatrix}$$

Note that I_y and I_y are respectively the partial derivative of the image with respect to x and y.

2.1 Image derivatives

Since images are discrete we need to compute some approximation of the derivative. The approximation used in this case is the following:

$$I_x = \frac{I(x+1,y)-I(x-1,y)}{2},\ I_y = \frac{I(x,y+1)-I(x,y-1)}{2}$$

We can exploit the convolution operations to compute these derivatives, using the following filters:

$$\mathrm{filter_x} = \left[\begin{bmatrix}-0.5 & 0 & 0.5\end{bmatrix}\right],\ \mathrm{filter_y} = \left[\begin{bmatrix}-0.5\end{bmatrix}, \begin{bmatrix}0\end{bmatrix}, \begin{bmatrix}0.5\end{bmatrix}\right]$$

This convolutions have been done in python in the following way:

Listing 1: Image gradients

```
filter_x = np.array([[1/2, 0, -1/2]])
filter_y = np.array([[1/2],[0],[-1/2]])

I_x = signal.convolve2d(img, filter_x, mode=’same’)
I_y = signal.convolve2d(img, filter_y, mode=’same’)
```

2.2 title