Feature Detection and Scale-space Theory
Signal and Image Processing

March 7, 2019

Feature detectors

1. Use the `skimage.feature.canny` function on the `hand.tiff` image. Try different settings for the parameters `sigma`, `low_threshold`, and `high_threshold` of the `canny` function. Create an illustration showing the results of the different settings and explain what the effect is of each of the parameters based on these results.

2. Use the `skimage.feature.corner_harris` function on the `modelhouses.png` image to compute a Harris corner response image (a.k.a. feature map). Try different settings for the parameters `sigma` and `k` for `method='k'` of the `corner_harris` function. Also try to fix `sigma` and change method to `method='eps'` and try different values of the `eps` parameter. Create an illustration showing the results of the different settings and explain what the effect is of each of the parameters based on these results.

3. Write a Python function that finds local maxima in the feature map generated by the `skimage.feature.corner_harris` function by using the function `skimage.feature.corner_peaks`. Apply this function to the `modelhouses.png` image and create a figure of the resulting corner points overlaid on the `modelhouses.png` image. Remember to indicate your choice of parameter settings in the caption of the figure.

Scale-space operators

1. Consider a Gaussian kernel

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \ . \tag{1}$$

   The convolution of a Gaussian with itself is also a Gaussian, i.e.,

$$G(x,y,\sigma) * G(x,y,\tau) = G(x,y,\sqrt{\sigma^2+\tau^2}) \ . \tag{2}$$

   Consider the function made from this analytical expression,

$$I(x,y) = G(x,y,\sigma) \ . \tag{3}$$

   We will use this function as a synthetic model of a blob in an image.

   Using Python, make a discrete image from the function in eq. (3) for some fixed σ by sampling the x, y axes appropriately. Visually illustrate your solution and eq. (2) by calculating images from its scale-space,

$$I(x,y,\tau) = I(x,y) * G(x,y,\tau) \ . \tag{4}$$

   Hint: You can use the `scale` function written previously in Assignment 4.

2. Consider the 2-dimensional scale normalized partial derivatives of order $i+j$ at scale τ,

$$I_{x^i y^j}(x,y,\tau) = \tau^{\gamma(i+j)} \frac{\partial^{i+j} I(x,y,\tau)}{\partial x^i \partial y^j}, \tag{5}$$

where $\gamma \in \mathbb{R}$ is a parameter of the scale normalization and $I(x, y, \tau)$ is the scale space of an image. Now consider the scale normalized Laplacian of the blob image defined in eq. (3),

$$H(x, y, \tau) = I_{xx}(x, y, \tau) + I_{yy}(x, y, \tau), \tag{6}$$

Using $\gamma = 1$, solve the following:

(a) Write the closed form expression for $H(x, y, \tau)$.

(b) Consider the point $(x, y) = (0, 0)$ and derive analytically the scale(s), τ, for which $H(0, 0, \tau)$ is extremal. Maple (or Mathematica) may be helpful, or simply solve it by hand. Characterize these extremal point(s) in terms of maximum, saddle, and minimum in (x, y, τ).

(c) Confirm your result in Python.

(d) Locating the maxima and minima (x, y, τ) in the scale-space of eq. (6) applied to images $I(x, y)$ in general is called blob detection with scale selection. Write a Python script that detects the 20 maxima and minima in the scale-space of the `sunflower.tiff` image with the largest absolute value of eq. (6). Indicate each detected point and corresponding detection scale τ with a dot and a circle centred on the point of detection and with a radius proportional to τ. Choose different colors for the circle and point so you can distinguish maxima from minima. What image structure does maxima of eq. (6) represent, and what image structure does minima represent?

3. Consider this model of a soft edge,

$$J(x, y) = \int_{-\infty}^{x} G(x', 0, \sigma)\, dx' \tag{7}$$

for some constant σ and Gaussian function G defined as in eq. (1). Consider also its scale-space,

$$J(x, y, \tau) = J(x, y) * G(x, y, \tau), \tag{8}$$

and the scale-normalized spatial squared gradient magnitude operator

$$\|\nabla J(x, y, \tau)\|^2 = J_x^2(x, y, \tau) + J_y^2(x, y, \tau)\ . \tag{9}$$

Using Python, make a discrete image from the function in eq. (7) for some fixed σ by sampling the x, y axes appropriately. Visualize the scale-space eq. (8) of this soft edge model.

4. Using the scale normalized partial derivatives of eq. (5) and $\gamma = \frac{1}{2}$, solve the following:

(a) Write the closed form expression for $\|\nabla J\|^2$ for the soft edge model eq. (7).

(b) Derive analytically the scale, τ, for which $\|\nabla J\|^2$ is maximal in the point $(x, y) = (0, 0)$ (hint: You might need the fundamental theorem of calculus). Is this a maximum in (x, y, τ)?

(c) Confirm your result in Python.

(d) The maxima in (x, y, τ) of (9) is edge detection with scale-selection. In the `hand.tiff` image, detect the 100 maxima with the largest value of eq. (9) and indicate the point of detection and scale by circles with a radius proportional to the detection scale τ.