The Neural Code

Summer term 2022, Christian Leibold, Yuk-Hoi Yiu

Problem Set 3 **—Convolution—** **25.04.2019**

12.05.2022

---

1. (Convolution as Filter) Let us numerically explore convolutions

$$y(t) = (\kappa * x)(t) = \int_{-\infty}^{\infty} \mathrm{d}s\, \kappa(s)\, x(t-s)$$

(a) Load the fluoresence trace `exampleCAtrace.npy` from a calcium imaging experiment and plot it with the appropriate time axis (sampling frequency is 20 Hz).

(b) Let's now focus on a small snippet from sample 1000 to 2000, which contains three visible transients.

(c) Convolve the snippet with a Gaussian kernel with $\sigma = 250$ ms and interpret its action.

(d) Change σ and explore the effect.

(e) Next generate a new kernel by multiplying the Gaussian kernel ($\sigma = 250$ ms) with $-\sin(2\pi t)$ and plot the convolution. Why is the convolution approximating a differentiation?

(f) Construct a kernel that approximates a second derivative and verify.

2. (Image filters) Instead of convolving in time, one can also convolve in space:

$$Q(x,y) = (\kappa * P)(x,y) = \int_{-\infty}^{\infty} \mathrm{d}x' \int_{-\infty}^{\infty} \mathrm{d}y'\, \kappa(x',y')\, P(x-x', y-y')$$

If we consider $P(x,y)$ to reflect the intensity of an image at pixel (x,y), the output Q is a filtered image.

(a) Load the image `freiburg.jpeg` with `img=matplotlib.image.imread(....)`, average over the three color channels (`img = np.mean(np.array(img),axis=2)`) and plot (`matplotlib.pyplot.imshow`).

(b) Generate a 2-d Gaussian kernel $\sigma = 5$ pixels (check out `numpy.mgrid`) and convolve the image (use `from scipy.signal import convolve2d`) to obtain a blurred image.

(c) Next, write a function that implements a Gabor kernel

$$\kappa(\vec{x}) = \exp[-(\vec{x})^2/(2\sigma^2)]\, \cos(\vec{k}\cdot\vec{x} - \phi)$$

with $\vec{x} = (x,y)$ and plot it for $\sigma = 5$ pixels, varying values of ϕ and various vectors $\vec{k}$ with $|\vec{k}| = 0.5$. You can obtain vector $\vec{k}$ with constant length L via $\vec{k} = L\,(\cos\alpha, \sin\alpha)$.

(d) Which value of ϕ approximates a spatial derivative (edge detector)?

(e) Obtain spatial derivatives for varying directions $\vec{k}$ with $|\vec{k}| = 0.5$.

(f) Now change the length of the wave vector $|\vec{k}|$. What is the effect of $|\vec{k}|$ on the convolutions?