Assignment 2

- Gradient of Gaussian filtering (GoG)
- Förstner interest point operator

You are free to use the provided image (ampelmaennchen.png) or own photos. Always use **grayscale images** and scale your input image to double with values $[0, \ldots, 1]$ (`mat2gray`).

A) Gradient of Gaussian (GoG) filtering

a. Compute two GoG-filter masks for filtering in x- and y-direction (see slides of lectures and assignment session). Example: for $\sigma = 0.5$ the two masks are

$$G_x = \begin{bmatrix} 0.0000 & 0.0001 & 0.0000 & -0.0001 & -0.0000 \\ 0.0002 & 0.0466 & 0.0000 & -0.0466 & -0.0002 \\ 0.0017 & 0.3446 & 0.0000 & -0.3446 & -0.0017 \\ 0.0002 & 0.0466 & 0.0000 & -0.0466 & -0.0002 \\ 0.0000 & 0.0001 & 0.0000 & -0.0001 & -0.0000 \end{bmatrix}, G_y = G_x^T$$

b. Apply the two filters to your input image in order to derive two gradient images (one in x- and one in y-direction, I_x and I_y). Write a function for the filtering (i.e., don't use `imfilter`). Ignore the edges of the image (no padding needed).

c. Compute and plot the gradient magnitude image G (just a by-product and not used in further steps) with

$$G = \sqrt{(I_x)^2 + (I_y)^2},$$

where I_x and I_y are the filter outputs from step b.

B) Förstner operator: The goal is to detect points of interest in our input image

a. Compute the autocorrelation Matrix M for each pixel using a moving window of 5x5 pixels (use I_x and I_y, ignore the edges of the images).

b. Compute the cornerness w and roundness q for each pixel from M and store the values in two matrices (W and Q). Plot W and Q (`imshow, colormap(jet)`).

c. Derive a binary mask of potential interest points by simultaneously applying the thresholds $t_w = 0.004$ and $t_q = 0.5$ on W and Q, respectively. The result is a mask M_C with pixel values = 1, if ($w > t_w$ *and* $q > t_q$), and 0 otherwise.

d. Since we are only interested in pixels where $M_C = 1$, multiply W and Q with M_C (e.g. $\bar{Q} = Q \cdot M_C$) and use the function `imregionalmax` to derive the final interest points from $\bar{Q} \cdot \bar{W}$.

e. Plot an overlay of the initial input image and the detected points (`find, plot`).