174 3 Image processing

**Ex 3.11: Sharpening, blur, and noise removal** Implement some softening, sharpening, and non-linear diffusion (selective sharpening or noise removal) ﬁlters, such as Gaussian, median, and bilateral (Section 3.3.1), as discussed in Section 3.4.4.

Take blurry or noisy images (shooting in low light is a good way to get both) and try to improve their appearance and legibility.

**Ex 3.12: Steerable ﬁlters** Implement Freeman and Adelson’s (1991) steerable ﬁlter algo- rithm. The input should be a grayscale or color image and the output should be a multi-banded image consisting of *G*0*◦* and *G*90*◦* . The coefﬁcients for the ﬁlters can be found in the paper

1 1

by Freeman and Adelson (1991).

Test the various order ﬁlters on a number of images of your choice and see if you can reliably ﬁnd corner and intersection features. These ﬁlters will be quite useful later to detect elongated structures, such as lines (Section 4.3).

**Ex 3.13: Distance transform** Implement some (raster-scan) algorithms for city block and Euclidean distance transforms. Can you do it without peeking at the literature (Danielsson 1980; Borgefors 1986)? If so, what problems did you come across and resolve?

Later on, you can use the distance functions you compute to perform *feathering* during image stitching (Section 9.3.2).

**Ex 3.14: Connected components** Implement one of the connected component algorithms from Section 3.3.4 or Section 2.3 from Haralick and Shapiro’s book (1992) and discuss its computational complexity.

* Threshold or quantize an image to obtain a variety of input labels and then compute the area statistics for the regions that you ﬁnd.
* Use the connected components that you have found to track or match regions in differ- ent images or video frames.

**Ex 3.15: Fourier transform** Prove the properties of the Fourier transform listed in Ta- ble 3.1 and derive the formulas for the Fourier transforms listed in Tables 3.2 and 3.3. These exercises are very useful if you want to become comfortable working with Fourier transforms, which is a very useful skill when analyzing and designing the behavior and efﬁciency of many computer vision algorithms.

**Ex 3.16: Wiener ﬁltering** Estimate the frequency spectrum of your personal photo collec- tion and use it to perform Wiener ﬁltering on a few images with varying degrees of noise.

1. Collect a few hundred of your images by re-scaling them to ﬁt within a 512 *×* 512

window and cropping them.

1. Take their Fourier transforms, throw away the phase information, and average together all of the spectra.
2. Pick two of your favorite images and add varying amounts of Gaussian noise, *σn ∈*

*{*1*,* 2*,* 5*,* 10*,* 20*}* gray levels.

1. For each combination of image and noise, determine by eye which width of a Gaussian blurring ﬁlter *σs* gives the best denoised result. You will have to make a subjective decision between sharpness and noise.