Image Processing and Computer Vision 1

Chapter 3 – Spatial Filtering – week 5

Martin Weisenhorn

HS 2023

1 Book

1.1 Book by Gonzalez and Woods, 3.31 (a)

Show that the Gaussian kernel, $G(s,t)$, in Eq. (3-45) is separable.
Hint: read the paragraph *separable filter kernels* on Page 161 in Section 3.4. Alternatively, and in alignment with the definition (2-57) use the commutativity of filtering to swap image $f(x, y)$ and kernel $w(x, y)$ in Eq. (3-31) so that the filtering i.e. correlation is expressed in the general form (2-55). Finally, show that the kernel satisfies Eq. (2-57).

Book section 3.5 *Smoothing Spatial Filters*

$$w(s,t) = G(s,t) = Ke^{-\frac{s^2+t^2}{2\sigma^2}} \tag{3-45}$$

Book section3.4 *The Mechanics of Linear Spatial Filtering*

$$g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t)\, f(x+s, y+t) \tag{3-31}$$

Book section 2.6 *Image Transforms*
The kernel $r(x, y, u, v)$ of the image transformation

$$T(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\, r(x, y, u, v) \tag{2-55}$$

is separable if

$$r(x, y, u, v) = r_1(x, u)\, r_2(y, v). \tag{2-57}$$

1.2 Book by Gonzalez and Woods, 3.39*

An image is filtered with a kernel whose coefficients sum to 1. Show that the sum of the pixel values in the original and filtered images is the same.

1.3 Book by Gonzalez and Woods, 3.49

Show that the Laplacian defined in Eq. (3-59) is isotropic (invariant to rotation of the directions of derivation). Assume continuous quantities. From Table 2.3, coordinate rotation by an angle θ is given by

$$x = x' \cos\theta - y' \sin\theta \text{ and } y = x' \sin\theta + y' \cos\theta$$

where (x, y) are the unrotated and (x', y') are the rotated coordinates, respectively.

Book section 3.6 *Sharpening Spatial Filters*

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{3-59}$$

Hints.

1. Proof the following equation.

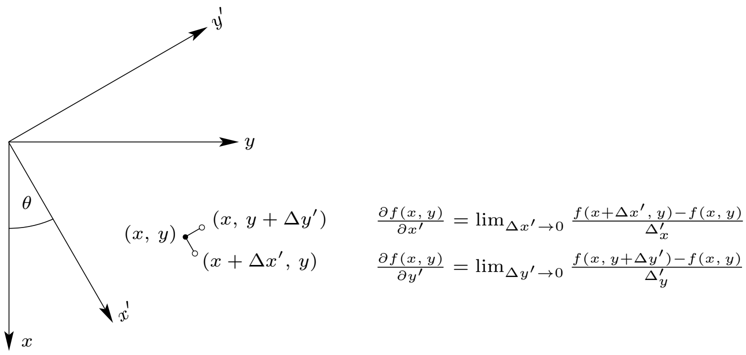
$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} = \frac{\partial^2 f(x,y)}{\partial x'^2} + \frac{\partial^2 f(x,y)}{\partial y'^2}$$

2. You will need the chain rule for the partial derivative. In this case it is:

$$\frac{\partial f(x,y)}{\partial x'} = \frac{\partial f(x,y)}{\partial x} \cdot \frac{\partial x}{\partial x'} + \frac{\partial f(x,y)}{\partial y} \cdot \frac{\partial y}{\partial x'}$$

3. There will be much to write, choose abbreviations like c instead of $\cos(\theta)$, f instead of $f(x,y)$, f_x instead of $\frac{\partial f(x,y)}{\partial x}$ and f_{xy} instead of $\frac{\partial f(x,y)}{\partial x \partial y}$.

Partial derivatives along transformed axes x' and y' are visualized in the figure below.



Note. Following the same idea, the magnitude of gradients can be shown to be non isotropic.

1.4 Book by Gonzalez and Woods, 3.53

Show that subtracting the Laplacian from an image gives a result that is proportional to the unsharp mask in Eq. (3-64). Use the definition for the Laplacian given in Eq. (3-62).
Hint: Rearrange Eq. (3.62) so that part of it has the form of an average filter.

Book section 3.6 *Sharpening Spatial Filters*

$$\nabla^2 f = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y) \tag{3-62}$$

$$g_{\text{mask}}(x,y) = f(x,y) - \bar{f}(x,y) \tag{3-64}$$

2 Practical Exercise

Write a program which implements a lowpass filter, where each pixel is replaced by the average of a $(2m+1) \times (2m+1)$ neighborhood. Observe the effects for different m. Use this lowpass filter to highpass filter an image using the unsharp masking technique. Compare the speed of a direct implementation with the predefined command `filter2` (MATLAB) or `scipy.signal.correlate2d` / `cv2.filter2D` (Python)

Additional Exercise

Use the script `webcam.*` to read images from the webcam. Convert the color images to grayscale. Apply your solution from above to the images of the camera using predefined commands. Try to make your implementation running in real time.