Some of the theoretical contents are taken from the instructor manual of the book „Digital Image Processing (4th Edition)“ by Rafael C. Gonzalez and Richard E. Woods. Therefore, the confidentiality class of this document is „internally extended“. The data can be used by OST members as of September 1, 2020, but must not be passed on to third parties.

Image Processing and Computer Vision 1

Chapter 4 – 2-D Convolution using the DFT – week 9

Martin Weisenhorn

HS 2023

1 Book

1.1 Book by Gonzalez and Woods, 4.40

You know from Table 4.3 that the DC-term, $F(0,0)$, of a DFT is proportional to the average value of its corresponding spatial image. Assume that the image is of size $M \times N$. Suppose that you pad the image with zeros to size $P \times Q$, where P and Q are given in Eqs. (4.102) and (4.103). Let $F_p(0,\,0)$ denote the DC-term of the DFT of the padded function $F_p(u,\,v)$.

(a) What is the ratio of the average values of the original and padded images?

(b) Is $F_p(0,0) = F(0,0)$? Support your answer mathematically.

(a) The average value of the original image is

$$\overline{f}(x,y) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

and, for the padded image,

$$\begin{aligned}\overline{f_p} &= \frac{1}{PQ}\sum_{x=0}^{P-1}\sum_{y=0}^{Q-1} f_p(x,y)\\ &= \frac{1}{PQ}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\\ &= \frac{MN}{PQ}\overline{f}(x,y)\end{aligned}$$

where the second step is result of the fact that the image is padded with 0s. Thus, the ratio of the average values is

$$R = \frac{PQ}{MN}$$

The ratio increases as a function of PQ, indicating that the average value of the padded image decreases as a function of PQ. This is as expected; padding an image with zeros decreases its average value.

(b) Yes, they are equal. We know that $F(0,0) = MN\overline{f}(x,y)$ and that $F_p(0,0) = PQ\overline{f_p}(x,y)$. From part (a), $\overline{f_p}(x,y) = MN\overline{f}/PQ$. Then,

$$\frac{F_p(0,0)}{PQ} = \frac{MN}{PQ}\frac{F(0,0)}{MN}$$

from which it follows that $F_p(0,0) = F(0,0)$.

1.2 Filter mask interpration in the frequency domain

Consider a 3×3 spatial Mask that averages the four closest neighbors of a point (x, y), but excludes the point itself from the average.

(a) Find the equivalent filter, $H(u, v)$, in the frequency domain.

(b) Show that the $H(u, v)$ is a lowpass filter transfer function for intensity oscillations along the x axis and along the y axis.

(a) The spatial average (excluding the center term) is

$$\begin{aligned} g(x,y) &= \frac{1}{4}[f(x,y+1) + f(x+1,y) + f(x-1,y) + f(x,y-1)] \\ &= \frac{1}{4}[\delta(x,y+1) + \delta(x+1,y) + \delta(x-1,y) + \delta(x,y-1)] \star f(x,y) \end{aligned}$$

From property 3 in Table 4.4,

$$\begin{aligned} G(u,v) &= \frac{1}{4}\left[e^{j2\pi v/N} + e^{j2\pi u/M} + e^{-j2\pi u/M} + e^{-j2\pi v/N}\right] F(u,v) \\ &= H(u,v)F(u,v) \end{aligned}$$

where

$$H(u,v) = \frac{1}{2}\left[\cos(2\pi u/M) + cos(2\pi v/N)\right]$$

is the filter transfer function in the frequency domain.

(b) To see that this is a lowpass filter transfer function, consider its values u *in the range* $[-M/2, M/2]$, *while keeping* $v = 0$. *The function assumes its highest value at the origin, i.e.* $u = 0$ *and decreases on either side of it, so it passes the DC term and low frequencies, and suppresses higher frequencies. Thus, it acts as a lowpass filter transfer function for intensity oscillations along the vertical image axis* x. *It behaves the same way for image intensity oscillations along the horizontal* y*-axis. However, for diagonal oscillations it passes low frequencies e.g.* $x = y = 0$, *stops intermediate frequencies* $x = M/4,\ y = N/4$ *and passes high frequencies* $x = M/2,\ y = N/2$. *We conclude that this filter has no general rotation symmetry, in other words, it is not isotropic. The same conclusion can be drawn by inspecting the filter in the spatial domain.*

1.3 Book by Gonzalez and Woods, 4.56

Do the following:

(a) Show that the Laplacian of a continuous function $f(t, z)$ of two continuous variables t and z satisfies the following Fourier transform pair:

$$\nabla^2 f(t,z) = \frac{\partial^2 f(t,z)}{\partial t^2} + \frac{\partial^2 f(t,z)}{\partial z^2} \Leftrightarrow -4\pi^2 \cdot \left(\mu^2 + \nu^2\right) \cdot F(\mu,\nu)$$

Hint: See Eq.(3-59) and study entry 12 in Table 4.4.

(b) The result in (a) is valid only for continuous variables. How would you implement the continuous frequency domain transfer function $H(\mu,\nu) = -4\pi^2 \cdot \left(\mu^2 + \nu^2\right)$ for discrete frequency variables u and v?

(c) As you saw in Example 4.21, the Laplacian result in the frequency domain was similar to the result of using a spatial mask with a center coefficient equal to -8. Explain the reason why the frequency domain result was not similar instead to the result of using a spatial mask with a center coefficient of -4.

(a) From Eq. (3-50) and entry 12 in Table 4.4 it follows

$$\nabla^2 f(t,z) \Leftrightarrow -4\pi^2(\mu^2+\nu^2)F(\mu,\nu)$$

(b) Generate a $P \times Q$ array centered on $[P/2, Q/2]$:

$$H(u,v) = -4\pi^2\left([u-P/2]^2 + [v-Q/2]^2\right)$$

for $u = 0,1,2,\ldots,P-1$ and $v = 0,1,2,\ldots,Q-1$ where P and Q are sizes to which the input image is padded prior to filtering. Then apply $H(u,v)$ like any other filter transfer function.

Note by the lecturer: $\mu \neq u$, $\nu \neq v$ but instead $\mu = \frac{u}{P\Delta T}$ and $\nu = \frac{v}{Q\Delta T}$. Hence, with $\Delta T = 1\,\text{pixel}$ the correct result is

$$\begin{aligned} H(u,v) &= -4\pi^2\left[\left(\frac{u-P/2}{P}\right)^2 + \left(\frac{v-Q/2}{Q}\right)^2\right] \\ &= -4\pi^2\left([u/P-1/2]^2 + [v/Q-1/2]^2\right). \end{aligned}$$

(c) The Laplacian operator is isotropic, a property that is captured well by the way the Laplacian transfer function is generated in the frequency domain. The Laplacian kernel with a -8 in the center and 1's surrounding it is a closer approximation to an isotropic kernel than the kernel with a -4 in the center and only four 1's surrounding it. Thus, we would expect the kernel with the -8 to give a result that is closer to the result obtained using the Laplacian transfer function in the frequency domain.

1.4 Book by Gonzalez and Woods, 4.57

Can you think of a way to use the Fourier transform to compute (or partially compute) the magnitude of the gradient [Eq. (3-58)] for use in image differentiation? If your answer is yes, give a method to do it. If your answer is no, explain why.

The answer is no. The Fourier transform is a linear process, while the square and square roots involved in computing the gradient are nonlinear operation. The Fourier transform could be used to compute the derivatives as differences (as in Problem 4.54), but the squares and square root values, must be computed directly in the spatial domain.