Beeldverwerken Assignment week 3

Stephan Grzelkowski (10342931)
Joeri Bes (10358234

April 30, 2017

Chapter 1

Introduction

All code exercises can be run from the script files in the runScripts subfolder. The figures for all of the exercises can be found in the figures subfolder.

Chapter 1

Convolution

1.1 1D convolution

We have to perform convolution of the signals:

$$g = \{0000\underline{1}1111\}, f = \{1\underline{2}1\}$$

The answer is:

$$(f * g) = \{0001\underline{3}4443\}$$

In cases when it's out of range we take the size equal to g.

1.2 1D convolution

We have to perform convolution of the signals:

$$f = \{0000\underline{1}1111\}, g = \{1\underline{2}1\}$$

The answer is:

$$(f * g) = \{0001\underline{3}4443\}$$

In cases when it's out of range we take a constant of zero.

1.3 1D convolution

We have to perform convolution of the signals:

$$f = \{0000\underline{1}1111\}, g = \{-1\underline{1}\}$$

The answer is:

$$(f * g) = \{0000\underline{1}0000\}$$

In cases when it's out of range we take a constant of zero.

1.4 Prove that convolution is commutative

The proof that convolution is commutative by showing that $(f * g) = (g * f)$:

$$(f * g)(x) = \sum_{i=-\infty}^{\infty} f(x-i)g(i)$$

$$(g * f)(x) = \sum_{j=-N}^{N} g(x-j)f(j)$$

$$(g * f)(x) = \sum_{j=-N}^{N} f(j)g(x-j)$$

Substitution of x-j= j and i = x-j

$$(g * f)(x) = \sum_{j=-N}^{N} f(x-i)g(i)$$

$$(f * g) = (g * f)$$

1.5 Prove that convolution is associative

The proof that convolution is associative by showing that $((f * g) * h) = (f * (g * h))$:

$$((f * g) * h)(x) = \sum_{i=-\infty}^{\infty} (f * g)(x)h(x-i)$$

replace $(f * g)(x)$

$$((f * g) * h)(x) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f(j)g(i-j)h(x-i)$$

verder...

1.6 Identity

The identity operation is $\{\underline{1}\}$

1.7 Intensity

Multiplying the image intensity is $\{\underline{3}\}$

1.8 Translate

Translating the image over vector $[-3, 1]^T$ equals to:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \underline{0} \end{bmatrix}$$

1.9 Rotate around an arbitrary angle around the origin

It's not possible to rotate an image by the use of convolution, convolution is based on its surrounding neighbors in a local point, in order to rotate an image you would need the whole image and not a local point.

1.10 Average 3x3 neighborhood

When taking the average, we take the average of all neighboring pixels:

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

1.11 Median 3x3 neighborhood

It is impossible to create a convolution kernel for the median. Because the median of neighbouring pixels is based on local pixels and not the whole image, thus the convolution kernel would fail.

1.12 Minimum value in a 5x5 neighborhood

It's impossible to compute the minimum value in a 5x5 neighborhood, the same rule as in exercise 1.11 applies to this.

1.13 Motion blur

The motion blur moved horizontally to the right over 5 pixels equals to:

$$\frac{1}{5}\begin{bmatrix} \underline{1} & 1 & 1 & 1 & 1 \end{bmatrix}$$

1.14 Optical blur

The optical blur spread out like a Gaussian with a standard deviation of 3 pixels:

$$\begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix}$$

1.15 Unsharp masking

Unsharp masking is achieved by substracting a blurred version from the original image:

1.16 Approximation of the derivative

An approximation of the derivative in the x-direction:

$$\frac{1}{2}[1\underline{0}-1]$$

1.17 Second derivative

In order to get the second derivative we have to multiply the convolution kernel with itself:

$$\frac{1}{4}[10\underline{-2}01]$$

1.18 Bilinear interpolating on a zoom

This is impossible because a convolution kernel cannot perform both a zoom and bilinear interpolation by using a single convolution kernel.

1.19 Threshold on an image

It's impossible to create a convolution kernel that's able to perform tresholding. Because a convolution kernel is not capable of discriminating neighboring values.

1.2 Gaussian convolution

1.20 Smooth derivatives of arbitrary order

The Gaussian always has a smooth derivative of arbitrary order because the Gaussian will always return in the derivatives. This is due to the exponential e.

1.21 Taking a derivative

It should be a convolution because the derivative can be calculated easily by looking at the neighboring pixels.

1.22 Associativity and commutativity

The derivatives of the convolution of the Gaussian can be described as:

$$((D * f) * G)$$

This can be written with the use of associativity as:

$$(D * (f * G))$$

With the use of commutativity it can be rewriten as:

$$D * (G * f))$$

And by using associativity again we get to the solution:

$$((D * G) * f)$$

1.23 First order derivative

The formula of the Gaussian convolution scheme in 2D equals to:

$$G(x1, x2) = \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1+x_2||^2}{2\sigma^2}}$$

The derivative for x_1:

$$\begin{aligned}\frac{\partial G_\sigma(x1, x2)}{\partial x_1} &= \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1+x_2||^2}{2\sigma^2}} * \frac{-x_1}{\sigma^2} \\ &= \frac{-x_1}{\sigma^2} G_\sigma(x1, x2)\end{aligned}$$

The derivative for x_2:

$$\begin{aligned}\frac{\partial G_\sigma(x1, x2)}{\partial x_2} &= \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1+x_2||^2}{2\sigma^2}} * \frac{-x_2}{\sigma^2} \\ &= \frac{-x_2}{\sigma^2} G_\sigma(x1, x2)\end{aligned}$$

These derivative are both in the form given to us, so the formula for the derivative holds and can be written as:

$$\begin{aligned}\frac{\partial G_\sigma}{\partial x} &= \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1+x_2||^2}{2\sigma^2}} * \frac{-x}{\sigma^2} \\ &= \frac{-x}{\sigma^2} G_\sigma(x)\end{aligned}$$

1.24 Second order derivative

For the second order derivative in 2D, we first calculate the 2nd order derivative in the x1 direction:

$$\frac{\partial^2 G_\sigma}{\partial x_1^2}(x_1, x_2) = \frac{-1}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} + \frac{-x_1}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} * \frac{-x_1}{\sigma^2}$$

$$= e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{2\pi\sigma^4} + \frac{x_1^2}{2\pi\sigma^6})$$

$$= \frac{1}{2\pi\sigma^2}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{\sigma^2} + \frac{x_1^2}{\sigma^4})$$

$$= (\frac{x_1^2}{\sigma^4} - \frac{1}{\sigma^2})G_\sigma(x1, x2)$$

After that we calculate the x2 direction:

$$\frac{\partial^2 G_\sigma}{\partial x_2^2}(x_1, x_2) = \frac{-1}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} + \frac{-x_2}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} * \frac{-x_2}{\sigma^2}$$

$$= e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{2\pi\sigma^4} + \frac{x_2^2}{2\pi\sigma^6})$$

$$= \frac{1}{2\pi\sigma^2}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{\sigma^2} + \frac{x_2^2}{\sigma^4})$$

$$= (\frac{x_2^2}{\sigma^4} - \frac{1}{\sigma^2})G_\sigma(x1, x2)$$

These derivative are both in the form given to us, so the formula for the derivative holds and can be written as:

$$\frac{\partial^2 G_\sigma}{\partial x^2}(x) = \frac{-1}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} + \frac{-x}{2\pi\sigma^4}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}} * \frac{-x}{\sigma^2}$$

$$= e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{2\pi\sigma^4} + \frac{x^2}{2\pi\sigma^6})$$

$$= \frac{1}{2\pi\sigma^2}e^{\frac{-||x_1+x_2)||^2}{2\sigma^2}}(\frac{-1}{\sigma^2} + \frac{x^2}{\sigma^4})$$

$$= (\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2})G_\sigma(x)$$

1.25 Smoothed image

1.26 Separable

We can show that the 2D Gaussian is seperable by going into the x_1-direction and x_2-direction:

$$G_\sigma(x_1) = \frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-||x_1||^2}{2\sigma^2}}$$

$$G_\sigma(x_2) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-||x_2||^2}{2\sigma^2}}$$

But if we take them together:

$$G_\sigma(x_1)G\sigma(x_2) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-||x_1||^2}{2\sigma^2}} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-||x_2||^2}{2\sigma^2}}$$

$$\frac{1}{2\pi\sigma^2} e^{\frac{-||x-y||^2}{2\sigma^2}}$$

$$= G_\sigma$$

As you can see from above, the scale d is equal to 1.

1.27 Separability of all derivatives

The 2D Gaussian derivative in the x_1-direction and x_2-direction is equal to:

$$\partial G_\sigma(x1) = \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1||^2}{2\sigma^2}} * \frac{-x_1}{\sigma^2}$$

$$\partial G_\sigma(x2) = \frac{1}{2\pi\sigma^2} e^{\frac{-||x_2||^2}{2\sigma^2}} * \frac{-x_2}{\sigma^2}$$

Let's take them together again:

$$\partial G_\sigma(x_1)\partial G\sigma(x_2) = \frac{1}{2\pi\sigma^2} e^{\frac{-||x_1+x_2||^2}{2\sigma^2}} (\frac{-x_1}{\sigma^2} + \frac{-x_1}{\sigma^2})$$

$$= \frac{-x_1 - x_2}{\sigma^4} G_\sigma(x)$$

$$= \partial G_\sigma$$