Practical works – n$^o$6–7
*Recursive filtering*

We consider the derivative filters defined by:

$$f_n(x) = -\mathrm{sign}(x)^{n+1} \frac{s_j^{n+1}}{n!} x^n e^{-s|x|} \tag{1}$$

where $n$ is the filter index and $s$ the scale factor. Each filter is normalized in amplitude:

$$\left| \int_{-\infty}^{0} f_n(\tau) d\tau \right| = \left| \int_{0}^{+\infty} f_n(\tau) d\tau \right| = 1 \tag{2}$$

A smoothing filter is defined by integrating the corresponding derivative filter (same index $n$):

$$h_n(x) = e^{-s|x|} \sum_{i=0}^{n} \frac{s^{n-i}}{(n-i)!} |x|^{n-i} \tag{3}$$

These filters, being defined by continuous functions, have an infinite support. Filtering operations will imply a truncation of their impulse response, the width of this impulse response varying according to the scale factor. The Z-transform permits to convert these filters in a efficient recursive form. We will focus in this practical work on the second filter ($n = 1$, Canny-Deriche filter):

$$f_1(x) = -s^2 x e^{-s|x|} \tag{4}$$
$$h_1(x) = (1 + s|x|) e^{-s|x|} \tag{5}$$

The recursive form is obtained in three steps: sampling, Z transformation to design the transfer function, Z transformation$^{-1}$ to obtain the difference (recursive) equation.

**Sampling** :

$$f_1[k] = -s^2 k T_s e^{-sT_s|k|} \tag{6}$$
$$h_1[k] = (1 + sT_s|k|) e^{-sT_s|k|} \tag{7}$$

Noting $\alpha = sT_s$ and $a = e^{-\alpha}$:

$$f_1[k] = -s\alpha k a^{|k|} \tag{8}$$
$$h_1[k] = (1 + \alpha|k|) a^{|k|} \tag{9}$$

**Transfer function** : We now apply the unilateral Z-transform on the causal parts of these filters:

$$f_1^+[k] = -s\alpha k a^k u[k] \tag{10}$$
$$h_1^+[k] = (1 + \alpha k) a^k u[k] \tag{11}$$

where $u[k]$ is the Heaviside (step) function. The Z-tranform are as follows:

$$F_1^+(z) = -s\alpha \frac{az^{-1}}{(1 - az^{-1})^2} \tag{12}$$

$$H_1^+(k) = \frac{1}{1 - az^{-1}} + \alpha \frac{az^{-1}}{(1 - az^{-1})^2} \tag{13}$$

The anti-causal parts are given by:

$$F_1^-(z) = -F_1^+(z^{-1}) - f_1[0] \quad f_1 \text{ is an odd function} \tag{14}$$
$$H_1^-(z) = H_1^+(z^{-1}) - h_1[0] \quad h_1 \text{ is an even function} \tag{15}$$

$f_1(0) = 0$ and $h_1(0) = 1$ are subtracted because already included in the causal parts. The causal transfer functions become:

$$F_1^+(z) = \frac{Y_F^+(z)}{X^+(z)} = -s\alpha \frac{az^{-1}}{(1 - az^{-1})^2} \tag{16}$$

$$H_1^+(z) = \frac{Y_H^+(z)}{X^+(z)} = \frac{1 + az^{-1}(\alpha - 1)}{(1 - az^{-1})^2} \tag{17}$$

or :

$$
\begin{aligned}
(1-az^{-1})^2 Y_F^+(z) &= -s\alpha a z^{-1} X^+(z) & (18)\\
(1-az^{-1})^2 Y_H^+(z) &= (1+az^{-1}(\alpha-1))X^+(z) & (19)
\end{aligned}
$$

Concerning the anti-causal transfer functions:

$$
\begin{aligned}
F_1^-(z) &= s\alpha \frac{az^1}{(1-az^1)^2} & (20)\\
H_1^-(z) &= \frac{a(\alpha+1)z - a^2z^2}{(1-az^{-1})^2} & (21)
\end{aligned}
$$

**Recursive form** : Applying now the Z-transform$^{-1}$, the causal recursive forms are:

$$
\begin{aligned}
y_k^{+F} - 2ay_{k-1}^{+F} + a^2 y_{k-2}^{+F} &= -s\alpha a x_{k-1} & (22)\\
y_k^{+H} - 2ay_{k-1}^{+H} + a^2 y_{k-2}^{+H} &= x_k + a(\alpha-1)x_{k-1} & (23)
\end{aligned}
$$

For the anti-causal parts:

$$
\begin{aligned}
y_k^{-F} - 2ay_{k+1}^{-F} + a^2 y_{k+2}^{-F} &= s\alpha a x_{k+1} & (24)\\
y_k^{-H} - 2ay_{k+1}^{-H} + a^2 y_{k+2}^{-H} &= a(\alpha+1)x_{k+1} - a^2 x_{k+2} & (25)
\end{aligned}
$$

Finally, the expected results $y$ of the filtering of $x$ are given by:

$$
\begin{aligned}
y_k^{t,F} &= y_k^{+F} + y_k^{-F} & (26)\\
y_k^{t,H} &= y_k^{+H} + y_k^{-H} & (27)\\
& & (28)
\end{aligned}
$$

• **Exercice 1** – *1D filtering*

**1.1** Construct a signal $x[k]_{k\in[1,40]} = \delta(k-20)$.

**1.2** Apply the causal and the anti-causal parts of the smoothing filter on the signal. Analyze the result.

**1.3** Construct a signal $x[k]_{k\in[1,40]} = u[k-10] - u[k-30]$.

**1.4** Apply the causal and the anti-causal parts of the derivative filter on the signal. Analyze the result.

• **Exercice 2** – *Canny-Deriche filtering*

**2.1** Load a gray image $I_m$ and display this image.

**2.2** Apply the smoothing (derivative) filter along the columns (rows) of the images to obtain the component of the gradient on the horizontal direction.

**2.3** Apply the smoothing (derivative) filter along the rows (columns) of the images to obtain the component of the gradient on the vertical direction.

**2.4** Write an edge detection function (no maxima extraction). Display the modulus and the phase of the results.

**2.5** Compare with convolution.

**Ref.:**

R. Deriche, Using Canny's criteria to derive a recursively implemented optimal edge detector, Int. J. Computer Vision, Vol. 1, pp. 167–187, April 1987.

O. Laligant, F. Truchetet. Generalization of Shen-Castan and Canny-Deriche filters, Proc. SPIE 3522, Intelligent Robots and Computer Vision XVII: Algorithms, Techniques, and Active Vision, 54 (October 6, 1998)