The noise estimator NOLSE

Olivier Laligant, Frederic Truchetet, Eric Fauvet

Le2i Lab. CNRS UMR 6306
Université de Bourgogne

Supplemental report for the paper "Noise estimation from digital step-model signal" published in IEEE Image Processing: DOI 10.1109/TIP.2013.2282123

The noise estimator NOLSE

Authors
Olivier Laligant, Fred Truchetet and Eric Fauvet

November 13, 2013

The noise estimator NOLSE

Olivier LALIGANT, Frédéric TRUCHETET, Eric FAUVET
Le2i Lab., CNRS UMR 6306, Université de Bourgogne
12 rue de la Fonderie, 71200 Le Creusot, France
olivier.laligant@u-bourgogne.fr

November 13, 2013

This report is not self-content and presents complementary definitions and demonstrations for the paper "Noise estimation from digital step-model signal".

1 The polarized derivatives

1.1 The derivative

First, let us recall the fundamental definition of the derivative. Let $f(x)$ be a function of a real variable and suppose that a is in the domain $\mathcal{D}$ of f. The derivative of f at $x = a$ is the limit:

$$f'(a) \quad = \quad \lim_{u \to 0} \frac{f(a+u) - f(a)}{u} \tag{1}$$

provided this limit exists. If this limit exists, then we say that f is differentiable at $x = a$. In this case, f belongs to the class $\mathcal{C}^1$ of differentiable signals. We emphasize that u can be a negative or a positive value. It is easy to find simple signals presenting non differentiable points (discontinuities for example) and mathematicians have found some continuous signals that are non differentiable everywhere. For discrete signals, the derivative does not exist, and the problem is reduced to defining an approximation of this derivative. The Taylor series permits one to define several approximations of this derivative, for example:

$$f'(a) \quad \simeq \quad \frac{f(a+1) - f(a)}{1} \tag{2}$$

1.2 Polarized and oriented components of the derivative

Let now introduce the real variables $v > 0$ and $w < 0$. In the continuous, we define case the oriented (**L**eft/**R**ight) and polarized (+/-) components of the polarized derivatives as:

$$f'_{R+}(a) = \begin{cases} \lim\limits_{v\to 0} \frac{f(a+v)-f(a)}{v} & \text{if } \frac{f(a+v)-f(a)}{v} \geq 0 \\ 0 & \text{elsewhere} \end{cases} \tag{3}$$

$$f'_{R-}(a) = \begin{cases} 0 & \text{if } \frac{f(a+v)-f(a)}{v} \geq 0 \\ \lim\limits_{v\to 0} \frac{f(a+v)-f(a)}{v} & \text{elsewhere} \end{cases} \tag{4}$$

$$f'_{L+}(a) = \begin{cases} \lim\limits_{w\to 0} \frac{f(a+w)-f(a)}{w} & \text{if } \frac{f(a+w)-f(a)}{w} \geq 0 \\ 0 & \text{elsewhere} \end{cases} \tag{5}$$

$$f'_{L-}(a) = \begin{cases} 0 & \text{if } \frac{f(a+w)-f(a)}{w} \geq 0 \\ \lim\limits_{w\to 0} \frac{f(a+w)-f(a)}{w} & \text{elsewhere} \end{cases} \tag{6}$$

In the discrete domain and introducing the nonlinear threshold operator T ($T(u) = u$ if $u \geq 0$, 0 elsewhere) the polarized, oriented discrete derivatives y_k^{R+}, y_k^{R-}, y_k^{L+} and y_k^{L-} of a signal x can be defined as:

$$\begin{cases} y_k^{R+} = T(x_{k+1} - x_k) \\ y_k^{R-} = -T[-(x_{k+1} - x_k)] \\ y_k^{L+} = T(x_k - x_{k-1}) \\ y_k^{L-} = -T[-(x_k - x_{k-1})] \end{cases} \tag{7}$$

Defining the polarized and oriented derivative operators $D^{R+}, D^{L+} : \mathbb{R} \to \mathbb{R}^+$, D^{R-} , $D^{L-} : \mathbb{R} \to \mathbb{R}^-$, the above equations are now written as:

$$\begin{cases} y^{R+} = D^{R+}(x) \\ y^{R-} = D^{R-}(x) \\ y^{L+} = D^{L+}(x) \\ y^{L-} = D^{L-}(x) \end{cases} \tag{8}$$

Note that in a 3-neighborhood, combined operators having interesting properties for edge detection [2]can be defined.

The combined operators D^- and D^+ have interesting properties in edge detection [2]. Considering the same local neighborhood they lead to improvements in localization and signal to noise ratio compared to the classical derivative operator $(1 - z^{-1})$.

2 The noise estimator

2.1 Noise signal components

From the signals in the above section, we propose to define the noise components:

$$\begin{cases} y^+ = \min\left(y^{L+}, -y^{R-}\right) \\ y^- = -\min\left(-y^{L-}, y^{R+}\right) \end{cases} \tag{9}$$

Considering y^+, the components are:

$$y_k^{L+} = \begin{cases} x_k - x_{k-1} & \text{if } x_k \geq x_{k-1} \\ 0 & \text{elsewhere} \end{cases} \tag{10}$$

$$y_k^{R-} = \begin{cases} 0 & \text{if } x_{k+1} > x_k \\ x_{k+1} - x_k & \text{elsewhere} \end{cases} \tag{11}$$

and we have the following:

$$y_k^+ = \begin{cases} 0 & \text{if } x_{k-1} \leq x_k < x_{k+1} \\ \min\left(x_k - x_{k-1}, -x_{k+1} + x_k\right) & \text{if } x_{k-1} \leq x_k \geq x_{k+1} \\ 0 & \text{if } x_{k-1} > x_k < x_{k+1} \\ 0 & \text{if } x_{k-1} > x_k \geq x_{k+1} \end{cases} \tag{12}$$

y_k^+ is always positive or null ($y_k^+ \geq 0$). Introducing the noisy step model to show that the y_k^+ component can be used to measure the noise values:

$$x_k = A.H_k + n_k \tag{13}$$

where H_k is the Heaviside function: $H_k = 1$ if $k \geq 0$, 0 elsewhere, A the amplitude and n_k the noise signal. From this, we developp y^+.

2.1.1 For $k \notin \{-1, 0, 1\}$:

$$y_k^+ = \begin{cases} \min\left(n_k - n_{k-1}, -n_{k+1} + n_k\right) & \text{if } n_{k-1} \leq n_k \geq n_{k+1} \\ 0 & \text{elsewhere} \end{cases} \tag{14}$$

or:

$$y_k^+ = \begin{cases} n_k + \min\left(-n_{k-1}, -n_{k+1}\right) & \text{if } n_{k-1} \leq n_k \geq n_{k+1} \\ 0 & \text{elsewhere} \end{cases} \tag{15}$$

Finally:

$$y_k^+ = \begin{cases} n_k - \max\left(n_{k-1}, n_{k+1}\right) & \text{if } n_{k-1} \leq n_k \geq n_{k+1} \\ 0 & \text{elsewhere} \end{cases} \tag{16}$$

2.1.2 For $k \in \{-1, 0, 1\}$:

$$y_0^+ = \begin{cases} \min\left(A + n_0 - n_{-1}, n_0 - n_1\right) & \text{if } -A + n_{-1} \leq n_0 \geq n_1 \\ 0 & \text{elsewhere} \end{cases} \tag{17}$$

The desired equation for y_0^+ to estimate the noise should be $y_0^+ = \min\left(n_0 - n_{-1}, n_0 - n_1\right)$, meaning A can influence the result of Eq. 17:

- If $A \leq 0$ (and $y_0^+ \geq 0$)

– if $(n_0 - n_{-1}) \geq (n_0 - n_1)$ or $(n_{-1} - n_1) \leq 0$, A does not influence the result

– if $(n_0 - n_{-1}) < (n_0 - n_1)$ or $(n_{-1} - n_1) > 0$,

  * if $(A + n_0 - n_{-1}) < (n - n_1)$ or $A < (n_{-1} - n_1)$, the result of the min function is biased by A: $(A + n_0 - n_{-1})$
  * else the min function does not give the correct term: $(n_0 - n_1)$ instead of $(n_0 - n_{-1})$

- if $A < 0$ (and $y_0^+ \geq 0$), we examine $\min\left(-|A| + n_0 - n_{-1}, n_0 - n_1\right)$

  – if $(n_0 - n_{-1}) \leq (n_0 - n_1)$ or $(n_{-1} - n_1) \geq 0$, the result is biased by A

  – if $(n_0 - n_{-1}) > (n_0 - n_1)$ or $(n_{-1} - n_1) < 0$,

    * if $(-|A| + n_0 - n_{-1}) > (n - n_1)$ or $A > (n_{-1} - n_1)$, A does not influence the result $(n_0 - n_1)$
    * else the min function yields the wrong and biased term: $(A + n_0 - n_{-1})$

It is important to note that the influence of $A < 0$ is limited by the condition $y_0^+ \geq 0$. For the two other points, we have:

$$y_{-1}^+ = \begin{cases} \min\left(n_{-1} - n_{-2}, n_{-1} - n_0 - A\right) & \text{if } n_{-2} \leq n_{-1} \geq A + n_0 \\ 0 & \text{elsewhere} \end{cases} \tag{18}$$

$$y_1^+ = \begin{cases} \min\left(A + n_1 - n_0, n_1 - n_2\right) & \text{if } -A + n_0 \leq n_1 \geq n_2 \\ 0 & \text{elsewhere} \end{cases} \tag{19}$$

These equations show that the effect of the amplitude A is identical. The influence of A on the estimator will be quantified in Section 2.2.

2.2 Influence of edge density and edge model on the estimator

In this section, we quantify the edge influence on the final noise estimate.

2.2.1 Step edge density

Let us consider the step edge model and the signal in Fig. 1(a) presenting the highest density of edge model profiles. The first equations of the noisy version are as follows:

$$\begin{cases} y_1^+ = \min\left(n_1 - n_0, n_1 - n_2 - A\right), & y_1^+ \geq 0 \\ y_2^+ = \min\left(A + n_2 - n_1, n_2 - n_3\right), & y_2^+ \geq 0 \\ y_3^+ = \min\left(n_3 - n_2, A + n_3 - n_4\right), & y_3^+ \geq 0 \\ y_4^+ = \min\left(n_4 - n_3 - A, n_4 - n_5\right), & y_4^+ \geq 0 \\ y_5^+ = ... \end{cases}$$

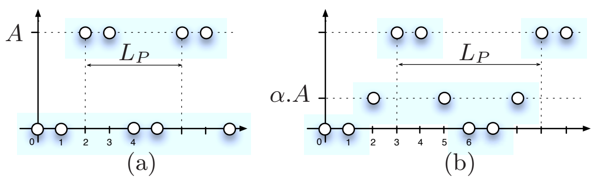

Figure 1: Highest edge density ($D = 1$ or 100%). (a) step edge model, (b) a more realistic edge model. L_P edge pattern length.

The edge amplitude A and its opposite $-A$ appear in these equations. We will estimate the influence of edges on the noise estimator by calculating the *pdf* of $y_1^+$and y_2^+. Rewritting $y_1^+$and y_2^+ as

$$\begin{cases} y_1^+ = n_1 - \max\left(n_0, n_2 + A\right), & y_1^+ \geq 0 \\ y_2^+ = n_2 - \max\left(-A + n_1, n_3\right), & y_2^+ \geq 0 \end{cases}$$

we have for the random variable M associated with the maximum of y_1^+:

$$P_M(v) = P_{N_0}(v).P_{N_2+A}(N_2 + A < v) + P_{N_0}(N_0 < v).P_{N_2+A}(v)$$

seeing:

$$P_{N_2+A}(u) = P_{N_2}(u + A)$$

we deduce:

$$\begin{aligned} P_M(v) \quad &= \quad P_{N_0}(v).\int\limits_{-\infty}^{v} P_{N_2}(x + A).dx \\ &\quad + P_{N_2}(v + A).\int\limits_{-\infty}^{v} P_{N_0}(x).dx \end{aligned}$$

and then:

$$\begin{aligned} P_{Y_1}(y \geq 0) \quad &= \quad \int_{-\infty}^{\infty} P_{N_1}(n).P_{N_0}(n + A - y).dn \\ &\quad \times \int_{-\infty}^{n-y} P_{N_2}(x).dx \\ &\quad + \int_{-\infty}^{\infty} P_{N_1}(n).P_{N_2}(n - y).dn \\ &\quad \times \int_{-\infty}^{n-y} P_{N_0}(x + A).dx \end{aligned}$$

Replacing A by $-A$ in the previous equation leads to P_{Y_2}. Finally, the second moment on the biased noise variable Y_A (modeling the measurement under the influence of the edge) is given by the following

$$E\left[Y_A^2\right] = \int_0^{\infty} y^2.\frac{P_{Y_1}(y) + P_{Y_2}(y)}{2}.dy$$

For Gaussian white noise, the biased random variable of the estimator is

$$\mathcal{S}_{\mathrm{A}}{}^2 = \frac{8}{\pi}.\frac{1}{N}.\sum_{k=1}^{N} Y_A^2$$

and its expected value is

$$E\left[\mathcal{S}_{\mathrm{A}}{}^2\right] = \frac{8}{\pi}.\frac{1}{N}.\sum_{k=1}^{N} E\left[Y_A^2\right]$$

The bias is a maximal if $A \gg \sigma$ and then:

$$E\left[\mathcal{S}_{\mathrm{A_M}}{}^2\right] = \frac{8}{\pi}.\frac{\sigma^2}{2}$$

It follows that in the presence of the highest density of edges, the maximum effect of the edge influence corresponds to the following maximum value of the biased estimator:

$$E\left[\mathcal{S}_{\mathrm{A_M}}{}^2\right] \simeq 1.27\sigma^2$$

Fig. 2 presents the bias on the estimated variance with respect to the amplitude

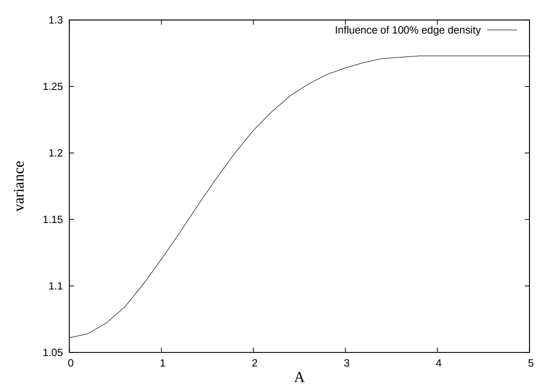

Figure 2: Influence of edge amplitude on the variance estimate for a 100% edge density (Gaussian white noise $\sigma^2 = 1$). The maximum value ($1.27\sigma^2$) for the biased variance is reached for high amplitudes; it decreases as the edge density decreases. $E\left[\mathcal{S}_{\mathrm{A_m}}{}^2\right] = \sigma^2\left(1 + D\left(\frac{4}{\pi} - 1\right)\right)$ where D is the edge density ($[0, 1]$).

of the edges. Defining the edge density as follows:

$$D = \frac{4}{L_P}$$

where L_P (see Fig.. 1) is the length of the (repeated) edge profile, and 4 stands for the minimum number of pixels to define an edge profile, we obtain the influence of the edge density on the variance

$$E\left[\mathcal{S}_{\mathrm{A_m}}{}^2\right] = \sigma^2\left(1 + D\left(\frac{4}{\pi} - 1\right)\right)$$

The estimate tends toward the true variance as the edge density decreases.

2.2.2 Realistic edge model

In real images, the projection of a step edge on the sensor does not truly correspond to a Heaviside function [4, 1]. Among the different effects of the acquisition system [3], the integration pixel surface induces an edge profile C_r that we will model as follows:

$$C_r(k) = \begin{cases} 0 & \text{if } k < 0 \\ \alpha A & \text{if } k = 0 \\ A & \text{if } k > 0 \end{cases}$$

with A representing the edge amplitude and α ($0 \leq \alpha \leq 1$) the partial integration of the intermediate pixel value (see Fig. 1b). Fig. 3 presents the effect of

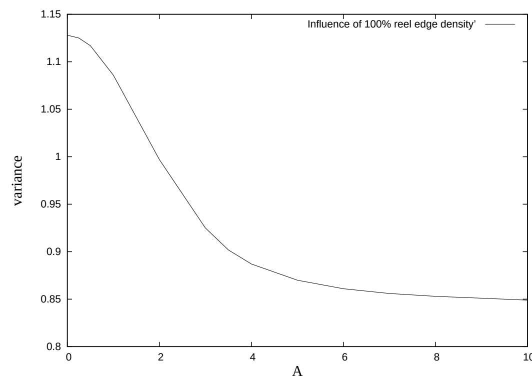

Figure 3: Influence of realistic edge amplitude on the variance estimate for an edge density of 100% edge density (Gaussian white noise $\sigma^2 = 1$). This density corresponds to a periodicity of 6 pixels. The variance decreases as the edge amplitude increases; the noise cannot be separated from the signal.

a more realistic edge model on the variance estimate.Considering Fig. 1b, three equations are sufficient to obtain the biased variance estimate for 100% edge density:

$$\begin{cases} y_1^+ = \min\left(n_1 - n_0, n_1 - n_2 - \alpha A\right), & y_1^+ \geq 0 \\ y_2^+ = \min\left(n_2 + \alpha A - n_1, n_2 + \alpha A - n_3 - A\right), & y_2^+ \geq 0 \\ y_3^+ = \min\left(n_3 + A - n_2 - \alpha A, n_3 + A - n_4 - A\right), & y_3^+ \geq 0 \\ y_4^+ = ... \end{cases}$$

we deduce the general equation:

$$\begin{aligned} P_{Y_k}(y \geq 0) \quad &= \quad \int_{-\infty}^{\infty} P_{N_k}(n).P_{N_{k-1}}(n+B-y).dn \\ &\times \int_{-\infty}^{n-y} P_{N_{k+1}}(x+C).dx \\ &+ \int_{-\infty}^{\infty} P_{N_k}(n).P_{N_{k+1}}(n+C-y).dn \\ &\times \int_{-\infty}^{n-y} P_{N_{k-1}}(x+B).dx \end{aligned}$$

where:

$$\begin{cases} B=0,\ C=-\alpha A & \text{for } \mathrm{P}_{\mathrm{Y}_1} \\ B=\alpha A,\ C=-(1-\alpha)A & \text{for } P_{\mathrm{Y}_2} \\ B=(1-\alpha)A,\ C=0 & \text{for } P_{\mathrm{Y}_3} \end{cases}$$

It follows the biased noise variance of Y_A is given by:

$$E\left[Y_A^2\right] = \int_0^{\infty} y^2.\frac{P_{Y_1}(y)+P_{Y_2}(y)+P_{Y_3}(y)}{3}.dy$$

For gaussian white noise, the biased random variable of the estimator is:

$$\mathcal{S}_{\mathrm{A}}{}^2 = \frac{8}{\pi}.\frac{1}{N}.\sum_{k=1}^{N} Y_A^2$$

and its variance :

$$E\left[\mathcal{S}_{\mathrm{A}}{}^2\right] = \frac{8}{\pi}.\frac{1}{N}.\sum_{k=1}^{N} E\left[Y_A^2\right]$$

The bias is maximum if $A \to \infty$ and then:

$$E\left[\mathcal{S}_{\mathrm{A_m}}{}^2\right] = \frac{8}{\pi}.\frac{\sigma^2}{3}$$

In presence of the highest density of edges, the maximum effect of edge influence correspond mostly to this value of the biased estimator:

$$E\left[\mathcal{S}_{\mathrm{A_m}}{}^2\right] \simeq 0.849\sigma^2$$

Considering the intermediate pixel (αA), for high edge amplitude (compared to the noise level), the effect of the noise is equivalent to a new value of α: α'. If $0 \leq \alpha' \leq 1$, the algorithm cannot separate the noise from the signal because the profile does not, in this case, contain any impulse part. If we suppose α represents a random variable characterized by a uniform distribution, for $A \gg \sigma$ we have $E[S_A^2] \simeq 0.87\sigma^2$ (the variance estimate varies between 0.85

and 1.22 according to α like a U profile). Obviously, the error decreases as the edge density decreases (here $D = 6/L_p$). For more extended edge profiles (several pixels), the effect on the variance estimate decreases, because the slope corresponding to the extending edge profile is smaller and the noise is better detected.

2.2.3 Discussion

We showed in this section that while the influence of the edge is limited, it is not negligible for a high density of edges. However, in most real images, the edge density is not extremely high, and extended edges counterbalance this influence. Moreover, due to the fact that the strongest effect is obtained for high amplitudes ($A > \sigma$), the estimate could be corrected by eliminating measures with high edge amplitudes (not addressed in this work).

2.3 Variance of the estimator

Let us consider the random variable S^2 associated with the estimator s^2: $S^2 = K_2.\frac{1}{N}.\sum_{k=1}^{N} Y_k^2$. The variance is:

$$\begin{aligned} Var\left(S^2\right) &= E\left[S^4\right] - E\left[S^2\right]^2 \\ &= E\left[\left(.\frac{K_2}{N}.\sum_{k=1}^{N} Y_k^2\right)^2\right] - E\left[.\frac{K_2}{N}.\sum_{k=1}^{N} Y_k^2\right]^2 \end{aligned}$$

or: $\frac{N^2}{K_2^2}.Var\left(S^2\right) = E\left[Y_1^4 + Y_1^2Y_2^2 + Y_1^2Y_3^2 + ... + Y_1^2Y_N^2 + Y_2^2Y_1^2 + Y_2^4 + Y_2^2Y_3^2 + ... + Y_N^2Y_N^2\right] - \left(N.E\left[Y^2\right]\right)^2$. Because the noise is assumed to be location-independent (random variable Y), we only use Y and the first terms to express the variance: $\frac{N^2}{K_2^2}.Var\left(S^2\right) = E\left[N.Y^4 + (2N-2).Y_1^2Y_2^2 + (2N-4).Y_1^2Y_3^2 + (N(N-5)+6).\,Y_1^2Y_4^2\right] - N^2.E\left[Y^2\right]^2$. We have $E\left[Y_1^2Y_2^2\right] = 0$ because there is no correlation between two consecutive measures:

$$\begin{cases} y_k = \begin{cases} n_k - \max(n_{k-1}, n_{k+1}) & \text{if } n_{k-1} \leq n_k \geq n_{k+1} \\ 0 & 0 \text{ elsewhere} \end{cases} \\ y_{k-1} = \begin{cases} n_{k-1} - \max(n_{k-2}, n_k) & \text{if } n_{k-2} \leq n_{k-1} \geq n_k \\ 0 & 0 \text{ elsewhere} \end{cases} \end{cases}$$

It follows: $\frac{N^2}{K_2^2}.Var(S^2) = N.E\left[Y^4\right] + (2N-4).E\left[Y_1^2Y_3^2\right] + (-5N+6).E\left[Y^2\right]^2$

where $E\left[Y^2\right] = \int_{-\infty}^{\infty} y^2 P_Y(y).dy$ and $E\left[Y^4\right] = \int_{-\infty}^{\infty} y^4 P_Y(y).dy$. The term $E\left[Y_1^2Y_3^2\right]$ needs to be evaluated because there is a correlation between two measures separated by one value. To simplify the notation, we consider again the first terms to develop the equations (we suppose here that n_0 exists):

$$\begin{cases} y_1 = n_1 - \max(n_0, n_2) & \text{if } n_0 \leq n_1 \geq n_2,\ 0 \text{ elsewhere} \\ y_3 = n_3 - \max(n_2, n_4) & \text{if } n_2 \leq n_3 \geq n_4,\ 0 \text{ elsewhere} \end{cases}$$

We first determine the joint probability $P_{Y_1Y_3}(y_1,y_3)$ in order to deduce $E\left[Y_1^2Y_3^2\right]=\int_{-\infty}^{+\infty}y_1^2y_3^2P_{Y_1Y_3}(y_1,y_3)dy_1dy_3$. As Y_1 and Y_3 are not independant, we write $P_{Y_1Y_3}(y_1,y_3)$, with the conditional probabilities as follows:

$$P_{Y_1Y_3}(y_1,y_3)=\int\limits_{-\infty}^{+\infty}P_{Y_1/U}(y_1,u).P_{Y_3/U}(y_3,u).P_U(u).du$$

where $U=N_2$ is the shared random variable between Y_1 and Y_3 ($u=n_2$ is the common sample between y_1 and y_2). Let us examine the conditional probability $P_{Y_1/U}(y_1,u)$ where two cases need to be detailed depending on the variable N_0 and the value u :

if $N_0\geq u$

we have $Y_1=N_1-N_0|Y_1\geq 0$, we deduce ($y_1=n_1-n_0$):

$$P_{Y_1/U}(y_1,u)=\int\limits_{u}^{+\infty}P_{N_1}(y_1+n_0).P_{N_0}(n_0)dn_0$$

if $N_0<u$

we obtain $Y_1=N_1-u|Y_1\geq 0$, we deduce ($y_1=n_1-u$)

$$P_{Y_1/U}(y_1,u)=P_{N_1}(y_1+u)\int\limits_{-\infty}^{u}P_{N_0}(n_0)dn_0$$

and then:

$$\begin{aligned}P_{Y_1/U}(y_1,u)&=P_{N_1}(y_1+u)\int\limits_{-\infty}^{u}P_{N_0}(n_0)dn_0\\&+\int\limits_{u}^{+\infty}P_{N_1}(y_1+n_0).P_{N_0}(n_0)dn_0\end{aligned}$$

The conditional probability for $P_{Y_3/U}(y_3,u)$ is identical and finally:

$$\begin{aligned}Var(S^2)&=K_2^2.\frac{1}{N^2}.\left\{N.E\left[Y^4\right]+(2N-4).E\left[Y_1^2Y_3^2\right]\right\}\\&+K_2^2.\frac{1}{N^2}.\left\{(-5N+6).E\left[Y^2\right]^2\right\}\end{aligned}$$

For a gaussian white noise of variance σ^2 ($E\left[Y^4\right]\simeq\frac{2\pi^2}{13}\sigma^4$, $E\left[Y_1^2Y_3^2\right]\simeq\frac{\pi^2}{41}\sigma^4$, $E\left[Y^2\right]^2=\frac{\pi^2}{8^2}\sigma^4$) and then: $\frac{\pi^2N^2}{8^2}Var\left(S^2\right)\simeq N\frac{2}{13}\pi^2\sigma^4+(2N-4)\frac{1}{41}\pi^2\sigma^4+(-5N+6)\frac{1}{8^2}\pi^2\sigma^4$. For a long signal:

$$Var\left(S^2\right)\approx\frac{8^2}{N}.\left(\frac{2}{13}+\frac{2}{41}-\frac{5}{8^2}\right)\sigma^4\approx\frac{8}{N}\sigma^4$$

3 Code of $NOLSE_2$

```
% -- fnolse(Im, type_noise)
%    Inputs: -Im the noisy image
%            -type_noise in { 'gaussian', 'salt & pepper', 'exponential', 'poisson', 'speckle' }
%    Ouputs: v1D v2D (variance estimation in 1D and 2D)
%            comment2, comment1 : results with comments
%            image_v : image containing the norm of the four measures
%            y1, y2, y3, y4 : images containing the noise maasures
%
%    Noise estimator in images
%    C. Olivier LALIGANT, 2009-10
%
%    No optimization !
%
%    Tested on octave (+package image) and Matlab V7R14
%
function [v2D, v1D, comment2, comment1, image_v, y1, y2, y3, y4] = fnolse(Im, type_noise)

%size(Im)
if(nargin == 2)
else
comment = sprintf('nolse(Image, ''type_noise''); \n');
%disp(comment);
comment = printf(' ''type_noise'' = {''gaussian'', ''salt & pepper'', ...
    ''exponential'', ''poisson'', ''speckle''} \n');
%disp(comment);
return;
end
border = 1; % border preservation
% verif. dim. for 1D estimators
[ni, nj] = size(Im);
a12=0; a34=0;
if((nj-2*border) > 1) a12=1;
end
if((ni-2*border) > 1) a34=1;
end
if( (a12==0) & (a34==0) ) disp('pb dimensions');
return;
end

% dim. verif. for 2D estimators
if((a12 == 0) | (a34 == 0))
disp('1D Signal => nonvalid 2D operators !');
end

% --- y1 estimator ---
yjp = min(DjLp(Im), -DjRn(Im) );
y1 = min(DiLp(yjp), -DiRn(yjp));
```

```
v1 = mse(y1, border) * 4;
% 1D estimator
s2_1 = mse(yjp, border) / (pi/8);

% --- y2 estimator ---
yjn = -min(DjRp(Im), -DjLn(Im));
y2 = -min( DiRp(yjn), -DiLn(yjn) );
v2 = mse(y2, border) * 4;
% 1D estimator
s2_2 = mse(yjn, border) / (pi/8);

% --- y3 estimator ---
yip = min( DiLp(Im), -DiRn(Im) );
y3 = min( DjLp(yip), -DjRn(yip) );
v3 = mse(y3, border) * 4;
% 1D estimator
s2_3 = mse(yip, border) / (pi/8);

% --- y4 estimator ---
yin = -min( DiRp(Im), -DiLn(Im) );
y4 = -min(DjRp(yin), -DjLn(yin));
v4 = mse(y4, border) *4;
% 1D estimator
s2_4 = mse(yin, border) / (pi/8);

% mean of the estimators
s2 = (a12*s2_1 + a12*s2_2 + a34*s2_3 + a34*s2_4)/(2*a12+2*a34);
v = (v1 + v2 + v3 + v4) / 4;
image_v = sqrt(y1.^2+y2.^2+y3.^2+y4.^2);
vp = (v1 + v3) / 4;
vn = (v2 + v4) / 4;

comment = sprintf('v1 2 3 4 : %f %f %f %f\n', v1, v2, v3, v4);
comment = sprintf('s1 2 3 4 : %f %f %f %f\n', s2_1, s2_2, s2_3, s2_4);

switch(type_noise)
case {'salt & pepper'}
   v1D = 0; % results reported in the strings comment1 and comment2
   v2D = 0;
   e=1/0.82269;    % slope correction
   K =  1.2036;    % model correction
   vi = K * v^e;
   vip = K*(v^e/2*(1-e) + e*v^(e-1)*vp);
   vin = K*(v^e/2*(1-e) + e*v^(e-1)*vn);
   comment1 = sprintf('Salt & pepper (2D) nolse estimator %f\n', s2);
   comment2 = sprintf('pepper %f   salt %f\n', vin, vip);
case {'speckle'}
   comment1 = sprintf('Speckle noise\n');
   comment2 = sprintf('2D nolse estimator: variance = %f\n', v);
   v1D = -1;
```

```
   v2D = v;
case {'poisson'}
   comment1 = sprintf('Poisson noise\n');
   comment2 = sprintf('2D nolse estimator: lambda = %f\n', v);
   v1D = -1;
   v2D = v;
case {'gaussian'}
   comment = sprintf('Gaussian noise\n');
   comment1 = sprintf('1D nolse estimator: variance = %f\n', s2);
   comment2 = sprintf('2D nolse estimator: variance = %f\n', v);
   v1D = s2;
   v2D = v;
case {'exponential'}
   comment = sprintf('Exponential noise : p(x)=1/2B.exp(-|x|/B)\n');
   m = mean(mean((yjp-yjn)*a12+(yip-yin)*a34))/(2*a12+2*a34);
   v1D = 8/3*m;
   comment1 = sprintf('1D nolse estimator: B = %f\n', v1D);
   m = mean(mean(y1-y2+y3-y4))/4;
   v2D = sqrt(2)*8/3*m;
   comment2 = sprintf('2D nolse estimator: B = %f\n', v2D);
otherwise
   disp('Noise not handled');
   comment1 = sprintf('other case: gaussian 1D nolse estimator %f\n', s2);
   comment2 = sprintf('other case: gaussian 2D nolse estimator %f\n', v);
   v1D = s2;
   v2D = v;
end

end%function %nolse

% operators D+, D-
function yjLp = DjLp(Im)
yjLp = thresh0(conv2(Im, [0 1 -1], 'same'));
end%function

function yjRn = DjRn(Im)
yjRn = -thresh0(-conv2(Im, [1 -1 0], 'same'));
end%function

function yiLp = DiLp(Im)
yiLp = thresh0(conv2(Im, [0; 1; -1], 'same'));
end%function

function yiRn = DiRn(Im)
yiRn = -thresh0(-conv2(Im, [1;  -1; 0], 'same'));
end%function

function yjRp = DjRp(Im)
yjRp = thresh0(conv2(Im, [1 -1 0], 'same'));
end%function
```

```
function yjLn = DjLn(Im)
yjLn = -thresh0(-conv2(Im, [0 1 -1], 'same'));
end%function

function yiRp = DiRp(Im)
yiRp = thresh0(conv2(Im, [1; -1; 0], 'same'));
end%function

function yiLn = DiLn(Im)
yiLn = -thresh0(-conv2(Im, [0;  1; -1], 'same'));
end%function
% end operators D+, D-

function st = thresh0(s)
st = s.*(sign(s)+1)/2;
end%function
```

References

[1] E. Bourennane, P. Gouton, M. Paindavoine, and F. Truchetet. Generalization of canny-deriche filter for detection of noisy exponential edge. *Signal Processing*, pp. 1317–1328, vol. 12, no. 10, Oct. 2002.

[2] O. Laligant and F. Truchetet. "A nonlinear derivative scheme applied to edge detection". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 242-257, Feb. 2010.

[3] S.K. Nayar and M. Watanabe. "Real time Focus Range sensor". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no 2, pp. 1186–1198, 1996.

[4] M. Petrou and J. Kitler. Optimal edge detectors for ramp edges. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 5, pp. 483–491, May 1991.