Print ISSN:1312 – 2622; Online ISSN: 2367-5357
DOI:10.2478/itc-2013-0002

# Pre-Processing of Hyperspectral Images Using Nonlinear Filters

**V. Behar, V. Bogdanova**

***Key Words:*** *Hyperspectral imaging; random field simulation; edge-preserving filtration.*

***Abstract.*** *In this paper the use of a set of nonlinear edge-preserving filters is proposed as a pre-processing stage with the purpose to improve the quality of hyperspectral images before object detection. The capability of each nonlinear filter to improve images, corrupted by spatially and spectrally correlated Gaussian noise, is evaluated in terms of the average Improvement factor in the Peak Signal to Noise Ratio (IPSNR), estimated at the filter output. The simulation results demonstrate that this pre-processing procedure is efficient only in case the spatial and spectral correlation coefficients of noise do not exceed the value of 0.6.*

## 1. Introduction

A significant recent achievement in remote sensing is the development of hyperspectral sensors and the corresponding software used to analyze and interpret the resulting image data [1]. During the past decade hyperspectral image analysis has become one of the most powerful and fastest growing technologies in the field of remote sensing. The hyperspectral sensors collect information as a set of 2D images. Each 2D image represents a range of the electromagnetic spectrum known as a spectral band. These 2D images are then combined, forming a three-dimensional data file (a cube of data), which includes spatial information (2D images) of the object, added with spectral information (1D) of each spatial coordinate. On the other hand, the spectrum received at this point of the object corresponds to each point of the image [2]. Hyperspectral imaging allows by finding objects with their physical and chemical structure to identify a specific structure or a condition of the vegetable cover, to define the geological structure, to reveal the chemical composition of water. The hyperspectral sensors are used in the following areas:

- In agriculture: spatial distribution of different types of land, assessment of the condition of crops, areas of plants distribution (including weed and narcotic-containing), violations of the vegetable and soil cover, definition of types of the damp and salted soils, etc. [3,4].
- In forestry: assessment of the condition of a forest stand and areas of distribution of wreckers, drawing maps of distribution of breeds of trees, determination of the quantitative parameters of stocks of wood and reservoirs condition, identification of minerals, detection of temperature anomalies [5].
- In ecology: identification and localization of the defects of oil and gas pipelines, detection of an oil film, etc. [6].
- In other spheres: mapping of underground pipelines, tunnels, control of deformations in oil and gas pipelines, search and rescue works on sea and land, studying of undercurrents, protection of wild animals (definition of types and monitoring of the number of a livestock), assessment of the heat losses in the cities and certain buildings, detection and localization of the underground and land mines and unexploded shells [7].

In Hyperspectral Imaging the great problem of the existing satellite sensors is the low Signal to Noise Ratio (SNR) of the acquired images, particularly in low reflectance areas. The quality of the collected spectral signatures is usually degraded by the sensor noise that can be correlated. In object detection applications the noise reduces the efficiency of the detection techniques in a manner that depends both on its intensity and on its correlation properties. For this reason the pre-processing is a necessary step in object detection using hyperspectral images. It should be noted, however, that it takes relatively large computational resources to implement the filters in real-time environment.

In this paper we study the possibility to use nonlinear filters at the pre-processing stage in order to improve SNR of hyperspectral images before object detection. Several well known nonlinear edge preserving filters, such as SNN, Kuwahara (three versions), Cubic, Rational, Sticks, Anisotropic Diffusion, Nonlinear Gaussian are studied for the aim of pre-processing of hyperspectral images [9-15]. The efficiency of nonlinear filtration is evaluated in terms of the average Improvement in the Peak SNR (PSNR) estimated at the filter output.

## 2. Signal Model

In accordance with [8] we assume that the noise in a recorded 3D hyperspectral image is additive and signal independent. We assume that a 3D hyperspectral image can be considered as a set of 2D images recorded in *K* spectral band. The process of formation of a 3D hyperspectral image can be written as

(1) $P(k) = |Y(k)| = |F(k) + N(k)|$, $k=1,\dots,K$,

where $|Y(k)|$ is the intensity of the 2D signal received in the *k*-th spectral band (spectral channel), $F(k)$ is the reflectance of the object in the *k*-th spectral band, $N(k)$ is a zero-mean 2D Gaussian process independent of *F,* which is stationary and correlated along both *X* and *Y* dimensions. Under this assumption, the intensity of a hyperspectral image can be modeled as a 3D random process with Rice distribution:

(2) $P(i,j,k)=|y(i,j,k)|=\sqrt{[f(i,j,k)+n_1(i,j,k)]^2+[f(i,j,k)+n_2(i,j,k)]^2}$

In (2) $P\,(i, j, k)$ is the intensity of a 2D image at pixel position $(i,\ j)$ recorded in the $k$-th spectral band and $f\,(i,\ j,\ k)$ is the $k$-th component of the spectral reflectance at pixel position $(i,\ j)$. The terms $n_1$ and $n_2$ are zero-mean Gaussian processes independent of $f$. The two processes $n_1\,(k)$ and $n_2\,(k)$ in (2) are spatially correlated and spatially stationary 2D zero-mean Gaussian processes with variance $\sigma^2_X\ (k)$ along $j$ and $\sigma^2_Y\ (k)$ along $i$. The correlation coefficients of both processes are $\rho_X$ along $j$ , $\rho_Y$ – along $i$ and $\rho_K$ – along $k$.

## 3. Image Processing

As shown in *figure 1*, the image processing investigated in this study includes the following stages: noisy image formation according to (1) and (2); image filtration using a set of nonlinear filters and PSNR estimation.

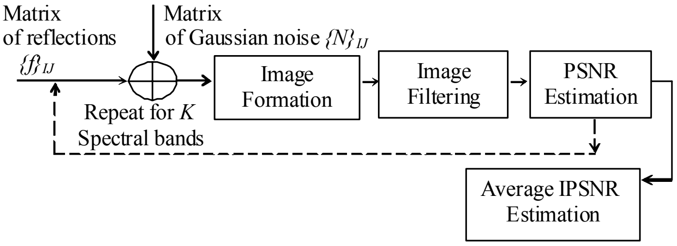


**Figure 1.** Block-scheme of image processing

The Peak Signal-to-Noise Ratio (PSNR) is used to measure the image quality before and after filtering. This quality measure is evaluated as the ratio between the maximum possible pixel power and the power of the corrupting noise. Usually PSNR is expressed in terms of the logarithmic decibel scale

(3) $PSNR = 10\log_{10}(MAX_I^{\ 2}) - 10\log_{10}(MSE)$.

In (3) $MAX_I$ is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per a sample, this is 255. According to (1), given a noise-free $I\times J$ monochrome image $F$ and its noisy version $P$, MSE (Mean Square Error) is defined as

(4) $MSE = 10\log_{10}(\frac{1}{I\cdot J}\sum_{i=1}^{I}\sum_{j=1}^{J}[P(i,j)-F(i,j)]^2$.

Since a higher PSNR at the filter output generally indicates filtration of higher quality, the quality of image filtration can be evaluated in terms of the Improvement in PSNR at the filter output (IPSNR)

(5) $IPSNR = PSNR_{out} - PSNR_{in}$.

Here $PSNR_{in}$ and $PSNR_{out}$ are PSNR values evaluated before and after the image filtering. If the filtering improves the image quality, IPSNR accepts positive values.

## 4. Filter Description

Unlike linear filtering, nonlinear filtering may increase the SNR of the images while preserving important image elements. The present study attempts to achieve this aim by testing several variants of nonlinear edge-preserving filters, which have been designated in literature as the most appropriate ones for image noise reduction.

### 4.1. Symmetric-Nearest-Neighbourhood (SNN) Filter

The kernel of this filter is divided into sub-windows [8]. All sub-windows must include the central pixel of the kernel, which, however, is not accounted in processing. The pixel intensity, which is closest to the intensity of the central pixel, is extracted from each sub-window. The intensity of the central pixel of the output image is determined as the average of the intensities of the pixels extracted from each sub-window. The algorithm of the filter is given in *table 1*.

The values of C1, C2, C3 and C4 are determined as follows:

$$C1 = \begin{cases} P1, \mathit{if}\ |P5-P1| < |P5-P9| \\ P9, otherwise \end{cases},$$

$$C2 = \begin{cases} P2, \mathit{if}\ |P5-P2| < |P5-P8| \\ P8, otherwise \end{cases}$$

(6)

$$C3 = \begin{cases} P3, \mathit{if}\ |P5-P3| < |P5-P7| \\ P7, otherwise \end{cases},$$

$$C4 = \begin{cases} P4, \mathit{if}\ |P5-P4| < |P5-P6| \\ P6, otherwise \end{cases}$$

### 4.2. Kuwahara-Nagao Filters

The kernel of this filter is divided into four sub-windows ($S_1$, $S_2$, $S_3$ and $S_4$), each of them including a central pixel of the filter kernel [8]. Each sub-window is used to estimate the variance of the intensity ($\sigma_1^2$, $\sigma_2^2$, $\sigma_3^2$ and $\sigma_4^2$), and next to find a sub-window with the minimal variance ($S_{min}$), where the average intensity $m\ (S_{min})$ is determined. The following three modifications of Kuwahara filter are used in practice:

- **Kuvahara-1.** The intensity of the central pixel in the window of the output image is replaced by $m\ (S_{min})$;
- **Kuvahara-2.** The intensity of the central pixel in the window of the output image is replaced by the pixel intensity from the sub-window $S_{min}$ with the smallest difference from $m\ (S_{min})$;
- **Kuvahara-3.** The intensity of the central pixel in the window of the output image is replaced by the pixel intensity

**Table 1.** SNN-filter with a kernel of size 3×3

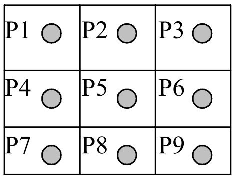
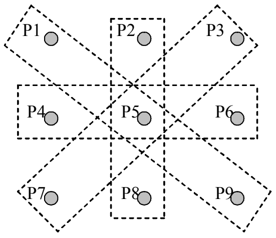
| Filter kernel | Sub-kernels | Filter algorithm |
|---|---|---|
|  |  | $(P5)_F = (C1+ C2+ C3+ C4)/4$ |

**Table 2.** Kuwahara filters with a kernel of size 3×3

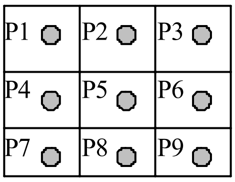
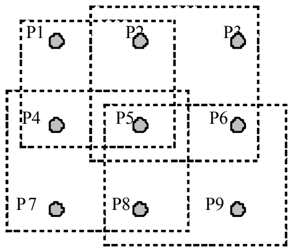
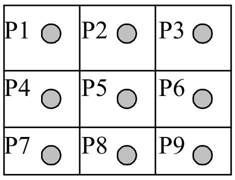
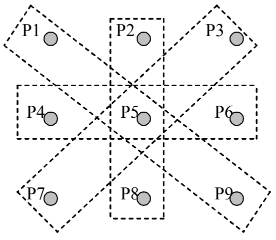
| Filter | Kernel window | Sub-windows | Filter algorithm |
|---|---|---|---|
| Kuw-1 |  |  | $(P5)_F = (P5+P6+P8+P9)/4$ |
| Kuw-2 | | | $(P5)_F = P9$, because $P9-(P5+P6+P8+P9)/4 = \min$ |
| Kuw-3 | | | $(P5)_F = P6$, because $P9-(P5+P6+P8+P9)/4 = \max$ |

**Table 3.** Rational filter with a kernel of size 3×3

| Filter kernel | Sub-windows | Filter algorithm |
|---|---|---|
|  |  | $(P5)_F = \alpha \cdot P5 + Z$ |

from the sub-window $S_{min}$ with the largest difference from *m* ($S_{\min}$).

In *table 2* three Kuwahara filters are presented with a kernel window of size 3×3. Four sub-windows are determined in the kernel window of the filter: $S_1$={P1, P2, P4, P5}; $S_2$={P2, P3, P5, P6}; $S_3$={P4, P5, P7, P8} and $S_4$={P5, P6, P8, P9}. Assuming that the sub-window $S_4$ has the smallest variance of the pixel intensity, i.e., $S_4 = S_{\min}$, the intensity of the central pixel in the window of the output image accepts values according to the algorithms in *table 2*.

## 4.3. Rational Filter

According to [9], the kernel window of this filter is divided into sub-windows in a manner shown in *table 3*.

The intensity of the central pixel in the window of the output image is formed as

$$(P5)_F = \alpha \cdot P5 + Z\,, \tag{7}$$

where

$$Z = \frac{w(P2+P8)}{w \cdot k(P2-P8)^2+1} + \frac{w(P4+P6)}{w \cdot k(P4-P6)^2+1} + \frac{w(P1+P9)}{w \cdot k(P1-P9)^2+\sqrt{2}} + \frac{w(P3+P7)}{w \cdot k(P3-P7)^2+\sqrt{2}}$$

$$\alpha = 1 - \frac{2w}{w \cdot k(P2-P8)^2+1} - \frac{2w}{w \cdot k(P4-P6)^2+1} - \frac{2w}{w \cdot k(P1-P9)^2+\sqrt{2}} - \frac{2w}{w \cdot k(P3-P7)^2+\sqrt{2}}$$

and $w$ = 0.16; $k$ = 0.01

## 4.4. Cubic Filter

According to [10], in the kernel window of this filter a sub-window is defined as shown in *table 4*.

The kernel sub-window includes four pixels P2, P4, P6, P8 and P5. The intensity of the central pixel in the window of the output image is formed as

$$\begin{aligned}(P5)_F = P5 + \lambda(&-2r_1/3 + r_2/6 - r_3/12 + r_4/3 + r_5/12 - r_6/12 - \\ &- r7/12 - r_8/3 + r_9/6),\end{aligned} \tag{8}$$

where

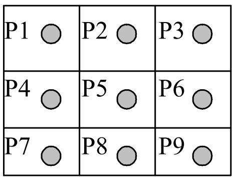
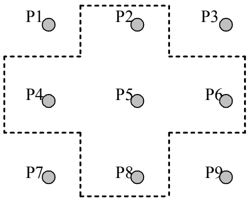
$r_1 = P5$; $r_2 = P2$; $r_3 = (P2)^2$; $r_4 = P5(P2)^2$; $r_5 = P8(P2)^2$; $r_6 = P6(P2)^2$; $r_7 = P4(P2)^2$; $r_8 = P2P5P8$; $r_9 = P2P4P8$.

The parameter λ in (8) controls the level of smoothing in the filtered image.

## 4.5. Nonlinear Gaussian Filter

According to [11] the algorithm for nonlinear Gaussian filtration of the image *P* (*p*) can be described as:

**Table 4.** Cubic filter with a kernel of size 3×3

| Filter kernel | Sub-window | Filter algorithm |
|---|---|---|
|  |  | $(P5)_F$= P5+λ·(-2r1/3 + r2/6 - r3/12 + r4/3 + r5/12 - r6/12 - r7/12 - r8/3 + r9/6) |

(9) $P_F(\sigma_X,\sigma_Z,p)=P(p)+\frac{1}{N_p}\sum_{q\in P} g_X(\|p-q\|)\cdot g_Z(P(q)-P(p))\cdot(P(p)-P(q))$.

In (9), $P_F(p)$ is the filtered image, $\sigma_X$ and $\sigma_Y$ are filter parameters, $g_X(t)$ and $g_Z(t)$ are the functions of Gaussian distribution, and $N_p$ is the normalizing factor

(10) $g_X(t)=\exp(\frac{-t^2}{2\sigma_X^2})$, $g_Z(t)=\exp(\frac{-t^2}{2\sigma_Z^2})$, $N_p=\sum_{q\in P} g_X(\|q-p\|)g_Z(P(q)-P(p))$.

The weight function $g_X(t)$ is responsible for image smoothing while the function $g_Z(t)$ preserves edges within the image. In order to enhance the noise reduction, the parameter $\eta$ can be added to the nonlinear Gaussian filter

(11) $P_F(\sigma_X,\sigma_Z,p)=P(p)+\frac{\eta}{N_p}\sum_{q\in P} g_X(\|p-q\|)\cdot g_Z(P(q)-P(p))\cdot(P(p)-P(q))$.

According to [11] the parameter $\eta$ must be within the range [1, 1.5]. In general, the smoothing effect of a single nonlinear Gaussian filter (11) may not be satisfactory. Therefore, a filter chain can be used, which comprises several filters in series, with different parameters $\sigma_X$ and $\sigma_Z$. The first filter in the filter chain serves mainly for reducing the contrast of the fine details in the images. The next stages perform additional contrast reduction, but at the same time sharpen the edges of the coarser structures, which have been blurred by the first step. Formally, the filter chain may be written as

(12) $P_{F,N}(p)=G_N(\sigma_{X,N},\sigma_{Z,N})G_{N-1}(\sigma_{X,N-1},\sigma_{Z,N-1})\ldots G_1(\sigma_{X,1},\sigma_{Z,1})P(p)$

with $\sigma_{X,n} = 2\sigma_{X,n-1}$ and $\sigma_{Z,n} = \sigma_{Z,n-1}/2$.

## 4.6. Adaptive Nonlinear Anisotropic Diffusion (AAD) Filter

As shown in [12,13], an Adaptive nonlinear Anisotropic Diffusion filter offers greater opportunities for improving the relationship “signal/noise” in the halftone image (gray palette). Theoretical results show that the filter smoothes the image without being deformed and without blurring the boundaries of objects of different intensity, which is particularly important for the identification of different objects in the images obtained. The iterative algorithm of the AAD filter can be written in the form

(13) $P_{i,j}^{n+1}=P_{i,j}^n+\Delta t\cdot d_{i,j}^n/4$,

where $P_{i,j}^{\;n}$ is the filtered image after the $n$-th iteration, $\Delta t$ is the filter parameter and $d_{i,j}^n$ are the divergence coefficients, which are calculated as follows:

(14) $d_{i,j}^n=c_{i+1,j}^n(P_{i+1,j}^n-P_{i,j}^n)+c_{i,j}^n(P_{i-1,j}^n-P_{i,j}^n)+c_{i,j+1}^n(P_{i,j+1}^n-P_{i,j}^n)+c_{i,j}^n(P_{i,j-1}^n-P_{i,j}^n)$

with symmetric boundary conditions

$$d_{-1,j}^n=d_{0,j}^n;\, d_{M,j}^n=d_{M-1,j}^n;\; j=0,1,1,\ldots,N-1$$

$$d_{i,-1}^n=d_{i,0}^n;\; d_{i,N}^n=d_{i,N-1}^n;\quad i=0,1,2,\ldots M-1$$

The diffusion coefficients $c^n_{\;i,j}$ are given by

(15) $c_{i,j}^n=c\left[q\left(\frac{1}{P_{i,j}^n}\sqrt{\left|\nabla_R P_{i,j}^n\right|^2+\left|\nabla_L P_{i,j}^n\right|^2},\frac{1}{P_{i,j}^n}\nabla^2 P_{i,j}^n\right)\right]$,

where $\nabla_R P_{i,j}^n=\left[(P_{i+1,j}^n-P_{i,j}^n),(P_{i,j+1}^n-P_{i,j}^n)\right]$

$$\nabla_L P_{i,j}^n=\left[(P_{i,j}^n-P_{i-1,j}^n),(P_{i,j}^n-P_{i,j-1}^n)\right]$$

$$\nabla^2 P_{i,j}^n=P_{i+1,j}^n+P_{i-1,j}^n+P_{i,j+1}^n+P_{i,j-1}^n-4P_{i,j}^n,$$

with symmetric conditions:

$$P_{-1,j}^n=P_{0,j}^n;\, P_{M,j}^n=P_{M-1,j}^n;\; j=0,1,1,\ldots,N-1$$

$$P_{i,-1}^n=P_{i,0}^n;\; P_{i,N}^n=P_{i,N-1}^n;\quad i=0,1,2,\ldots M-1$$

The diffusion coefficients $c^n_{\;i,j}$ in (12) are obtained as a result of discretization of the diffusion function $c(q)$, which can be chosen as

(16) $c(q)=\frac{1}{1+[q^2(x,y,t)-q_0^2(t)]/[q_0^2(t)(1+q_0^2(t))]}$,

(17) $c(q)=\exp\{-\frac{q^2(n,m,t)-q_0^2(t)}{q_0^2(t)(1+q_0^2(t))}\}$.

In (16) and (17) $q_0(t)$ is the scale function and $q(x, y, t)$ is the instantaneous coefficient of variation determined by

(18) $q(n,m,t)=\sqrt{\frac{(1/2)(|\nabla P|/P)^2-(1/4^2)(\nabla^2 P/P)^2}{[1+(1/4)(\nabla^2 P/P)]^2}}$,

where $P(n,m)$ is the intensity at pixel ($n$, $m$). The instantaneous coefficient of variation $q$ ($n$, $m$, $t$) serves as an edge detector in the noisy image. This function exhibits high values at edges or on high-contrast features and produces low values in homogeneous regions. The scale function $q_0(t)$ efficiently controls the amount of smoothing applied to the image. It is estimated as:

(19) $q_0(t)=\sqrt{\mathrm{var}[z(t)]}\big/\bar{z}(t)$,

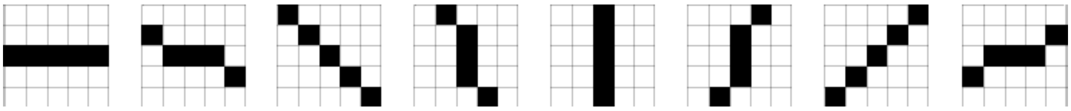


**Figure 2.** A set of the kernel windows of sub-filters

where var[$z(t)$] and $\bar{z}(t)$ are the intensity variance and mean value over a homogeneous area at *t*, respectively. The scale function $q_0(t)$ can be approximated by

$$q_0(t) \approx q_0 \exp(-\rho t) \tag{20}$$

where $\rho$ is a constant and $q_0$ is the coefficient of noise variation in the observed image.

## 4.7. Sticks Filter

The pixel intensity of the filtered image is formed as the maximum value of "2*n* – 2" pre-filtered images. The kernel of these sub-filters $s_{èi}$ has the form of a stick, which includes *n* pixels with angular orientation $\theta_i$ [14]. A set of these sub-filter kernels is $S = \{s_{èi} \mid i = 1, \ldots, 2n - 2\}$. In this study we used a filter with 8 sub-*2*filters, the kernels of which include 5 pixels, and the stick thickness is $k = 1$ (*figure 2*).

After pre-filtering of the input image the following (2*n* – 2) images are formed:

(21) .

The final filtered image is formed as

$$P_F(x, y) = \max\{F_i(x, y)\},\ i = 1 \ldots 2n - 2\,. \tag{22}$$

# 5. Numerical Results

In this study we tested the possibility of the usage of nonlinear filtration as pre-processing of hyperspectral images. For this aim, seven 2D images (obtained at seven different spectral bands) of the same object recorded by a spectrometer of high quality are used as reflections matrices $F_1, F_2, \ldots, F_7$. For each reflections matrix $F_k$ ($k$=1…7), the noise matrix $N_k$ is simulated as a stationary and correlated 2D Gaussian process, as a result of which seven noisy images $P_1, P_2, \ldots, P_7$ are formed in accordance with (2). In simulation of noise, the noise correlation coefficients $\rho_X$ (along X-axis), $\rho_Y$ (along Y-axis) and $\rho_k$ (along numbers of spectral bands) vary within the range from 0 up to 0.8. For the sake of simplicity, we consider the variant when $\rho_X = \rho_{Y,} = \rho_k = \rho$. In noise simulation, the noise variances $\sigma_X^2$ (along X-axis), $\sigma_Y^2$ (along Y-axis) and $\sigma_k^2$ (along numbers of spectral bands) are determined as a function of the SNR value. For simplicity we have assumed that $\sigma_X^2 = \sigma_Y^2 = \sigma_K^2 = \sigma^2$, which are determined according to a given SNR value in decibels:

$$\sigma = \frac{\sum_i \sum_j \sum_k F_{i,j,k}}{SNR \cdot I \cdot J \cdot K}, \text{ where } SNR = 10^{SNR\_dB/20}\,. \tag{23}$$

In (23) ($I$×$J$) is the image size in the *k*-th spectral band and *K* is the number of spectral bands. All noisy images are simulated by algorithm (2) and then they are filtered by a set of different nonlinear filters (SNN, Kuwahara-1,2,3, Rational, Cubic, Gaussian, AAD in two variants and Sticks). In *tables 5-7* the values of the average improvement factor *IPSNR* are presented, evaluated with respect to the filter type and noise parameters (SNR in dB and ñ).

**Table 5.** Average improvement factor depending on the correlation coefficient ρ (SNR=0dB)

| **Average improvement factor in dB: SNR db = 0** | | | | | |
|---|---|---|---|---|---|
| **ρ**<br>**Filter** | 0 | 0.2 | 0.4 | 0.6 | 0.8 |
| Gaussian | 1.8837 | 1.3950 | 0.8950 | 0.4886 | 0.2169 |
| AADm | 1.7450 | 1.2760 | 0.7956 | 0.4037 | 0.1462 |
| AADY | 1.7719 | 1.3498 | 0.9006 | 0.5063 | 0.2137 |
| Rat | 1.7283 | 1.3540 | 0.9583 | 0.5947 | 0.2863 |
| Sticks | -1.5390 | -1.5475 | -1.3953 | -1,0803 | -0.6579 |
| **CUB** | **5.0235** | **5.2973** | **4.6684** | **2.7153** | -1.9313 |
| Kuw1 | 2.2021 | 1.6357 | 0.9972 | 0.4478 | 0.1388 |
| Kuw2 | 2.1521 | 1,6011 | 0.9971 | 0.4390 | 0.1364 |
| Kuw3 | 1.3613 | 0.9937 | 0.5795 | 0.2517 | 0.0774 |
| SNN | 2.3329 | 1,6753 | 0.9763 | 0.4337 | 0.1231 |

**Table 6.** Average improvement factor depending on ρ and SNR

| **Average improvement factor in dB: SNR db = 5** | | | | | |
|---|---|---|---|---|---|
| **ρ**<br>**Filter** | 0 | 0.2 | 0.4 | 0.6 | 0.8 |
| Gaussian | 1.3113 | 0.8313 | 0.5140 | 0.3089 | 0.1724 |
| AADm | 1.3248 | 0.8038 | 0.4599 | 0.2428 | 0.1070 |
| AADY | 1.0995 | 0.7660 | 0.5185 | 0.3237 | 0.1671 |
| Rat | 0.7932 | 0.6676 | 0.5487 | 0.4033 | 0.2363 |
| Sticks | -2.333 | -1.9849 | -1.4913 | -1.0024 | -0.5828 |
| **CUB** | **3.9786** | **6,1132** | **6.1735** | **4.2189** | -0.5962 |
| Kuw1 | 1.4019 | 0.8997 | 0.5137 | 0.2410 | 0.1036 |
| Kuw2 | 1.3403 | 0.8661 | 0.4977 | 0.2376 | 0.1027 |
| Kuw3 | 0.8856 | 0.5429 | 0.3052 | 0.1473 | 0.0657 |
| SNN | 1.5504 | 0.9320 | 0.5183 | 0.2372 | 0.0860 |

**Table 7.** Average improvement factor depending on ρ and SNR

| **Average improvement factor in dB: SNR db = 10** | | | | | |
|---|---|---|---|---|---|
| **ρ**<br>**Filter** | 0 | 0.2 | 0.4 | 0.6 | 0.8 |
| Gaussian | 0.4638 | 0.3695 | 0.3057 | 0.2353 | 0.1568 |
| AADm | 0.5865 | 0.3972 | 0.2749 | 0.1779 | 0.0933 |
| AADY | 0.1242 | 0.2837 | 0.3064 | 0.2486 | 0.1506 |
| Rat | -0.3966 | 0.1278 | 0.3274 | 0.3252 | 0.2186 |
| Sticks | -3.2519 | -2.2991 | -1.5181 | -0.9609 | -0.5511 |
| **CUB** | **2.4416** | **6.3081** | **6.8855** | **4.8139** | -0.0017 |
| Kuw1 | 0.2713 | 0.3402 | 0.2799 | 0.1858 | 0.0973 |
| Kuw2 | 0.2017 | 0.3111 | 0.2674 | 0.1811 | 0.0964 |
| Kuw3 | 0.2230 | 0.2315 | 0.1821 | 0.1200 | 0.0644 |
| SNN | 0.4796 | 0.4028 | 0.3043 | 0.1778 | 0.078 |

The value of PSNR is evaluated at each filter output for each spectral band. The average improvement factor is

$$IPSNR_{ave} = \frac{1}{K}\sum_{k=1}^{K} PSNR_{out}(k) - \frac{1}{K}\sum_{k=1}^{K} PSNR_{in}(k). \quad (24)$$

Analysis of the numerical results shows that in general the efficiency of the nonlinear filtration as a pre-processing procedure of hyperspectral images depends strongly on the noise parameters – the noise intensity and correlation properties. It can be seen that the average improvement factor of all nonlinear filters decreases with increase of both SNR and ñ. The numerical results show that all filters considered, except for Sticks filter, improve the quality of hyperspectral images. However, the Cubic filter is the most efficient among them when the noise correlation coefficients do not exceed the value of 0.6 (given in bold fonts in the tables). In this case the usage of the Cubic filter allows improvement of the PSNR of hyperspectral images by 3-6 dB.

## Conclusions

It is shown that nonlinear filtration can be used as a pre-processing step which can improve the SNR of hyperspectral images before their further processing. This preprocessing procedure is efficient only in case the spatial and spectral correlation coefficients of noise do not exceed the value of 0.6. The next step of implementation is verification of the explored algorithms over a data set received by a laboratory hyperspectral camera installation.

## Acknowledgment

The research work reported in the paper is partly supported by Project AComIn "Advanced Computing for Innovation", Grant 316087, funded by FP7 Capacity Programme (Research Potential of Convergence Regions) and by Project No DFNI – I01/8, funded by the Bulgarian Science Fund.

## References

1. AVIRIS: Airborne Visible Infrared Imaging Spectrometer. http://aviris.jpl.nasa.gov.

2. Manolakis, D., D. Marden, G. Shaw. Hyperspectral Image Processing for Automatic Target Detection Applications. – *Lincoln Laboratory Journal,* 14, 2003, No. 1,79-116.

3. Ferwerda, J. G. Charting the Quality of Forage: Measuring and Mapping the Variation of Chemical Components in Foliage with Hyperspectral Remote Sensing. – *ITC Dissertation,* Wageningen University, 126, 2005, 166.

4. Tilling, A. K., et al. Remote Sensing to Detect Nitrogen and Water Stress in Wheat. The Australian Society of Agronomy, 2006.

5. Kamaruzaman, J. Precision Forestry Using Airborne Hyperspectral Imaging Sensor. – *Journal of Agricultural Science*, 1, 2009, No. 1, 142-147.

6. Ustin, S., D. Roberts, J. Gamon, G. Asner, R. Green. Using Imaging Spectroscopy for Study Ecosystem Processes and Properties. – *BioScience*, 54, 2004, No. 6, 523-534.

7. Funk, C., J. Theiler, D. Roberts, C. Borel. Clustering to Improve Matched Filter Detection of Weak Gas Plumes in Hyperspectral Thermal Imagery. – *IEEE Trans. on Geoscience and Remote Sensing*, 39, 2001, No. 7, 1410-1420.

8. Nagao, M., T. Matsuyama. Edge-preserving Smoothing Filters. – *Computer Graphics and Image Processing*, 9, 1979, No. 4, 394-407.

9. Ramponi, G. The Rational Filter for Image Processing. – *IEEE Signal Processing Letters*, 3, 1996, No. 3, 63-65.

10. Kroner, S., G. Ramponi, Edge Preserving Noise Smoothing with an Optimized Cubic Filter. Proc. COST-254 Workshop, Ljubljana, 1998, 19-21.

11. Aurich, V., J. Weule. Non-Linear Gaussian Filters Performing Edge Preserving Diffusion. 17th DAGM Symposium, Bielefeld, 1995, 538–545.

12. Yu, Y. Speckle Reducing Anisotropic Diffusion. – *IEEE Trans. Imag. Proc.,* 11, 2002, 1260–1270.

13. Sun, O., A. Hossack, J. Tang, S. Acton. Speckle Reducing Anisotropic Diffusion for 3D Ultrasound Images. – *Computerized Medical Imaging and Graphics*, 28, 2004, 461-470.

14. Stefan, D. Prostate Ultrasound Image Processing. – *Spring*, 13, 2007, No. 3, 20-23.

15. Aiazzi, B., L. Alparone, A. Barducci, S. Baronti, P. Marcoionni, I. Pippi, M. Selva. Noise Modelling and Estimation of Hyperspectral Data from Airborne Imaging Spectrometers. – *Annals of Geophysics*, 49, 2006, No. 1, 1-9.

**Manuscript received on 15.10.2013**

***Vera Petrovna Behar*** *was born in Bologoe, Russia. She received the M.S. degree in Applied Mathematics in the Sanct-Peterburg State Institute of Fine Mechanics and Optics (LITMO), and the Ph.D degree in radar signal processing in the Central Laboratory for Parallel Data Processing, Sofia, Bulgaria. Up to date she is an Assoc. Prof. in radar signal processing and works in the Institute of Information & Communication Technologies, Bulgarian Academy of Sciences, Sofia, Bulgaria. Her research experience is in signal & image processing,* ***CFAR*** *detection, adaptive array processing, Forward Scattering radar imaging, medical ultrasound imaging; signal transform (Hough, Wavelet , SFT) and Monte Carlo simulation. At the moment she works in the field of research related to adaptive array processing for navigation and sonar receivers, GPS signal processing and hyperspectral image processing.*

*Contacts:*
*Institute of Information and Communication Technologies, BAS*
*25A Acad. G. Bonchev St., 1113 Sofia, Bulgaria*
*e-mail: behar@bas.bg*

***Violeta Bogdanova*** *is born in Sofia. Her University Degree is Engineer of Automation, High Chemistry and Technology Institute, Sofia, Bulgaria, 1976. She an Assistant Professor at the Bulgarian Academy of Sciences, IICT, Sofia, Bulgaria. Her current research interests are in the area of signal and image processing.*

*Contacts:*
*Institute of Information and Communication Technologies, BAS*
*25A Acad. G. Bonchev St., 1113 Sofia, Bulgaria*
*e-mail: vbogdan@bas.bg*