Supervised Constrained Optimization of Bayesian Non-Local Means Filter with Sigma Preselection for Despeckling SAR Images

Luis Gomez, Cristian Munteanu

Electronic Engineering and Automatic Department, University of Las Palmas G.C., 35017, EITE, Campus Tafira, Spain

Maria Buemi, Julio Jacobo Berlles, Marta Mejail

Computation and Image Processing Group, University of Buenos Aires, Argentina

Abstract

Speckle reduction is an important problem in SAR image analysis. Recent years have seen how Bayesian filters emerge as the natural extension of the Non-local means (NL-means) filters, providing a general framework to deal with multiplicative (speckle) noise. In this work, we present an easy-to-use software tool applying an evolutionary algorithm to optimize a Bayesian Non-local means filter with sigma preselection for denoising SAR images. The desired result is a filtered image having a significative reduction in its variance but preserving the original mean value of the noisy image. A mixed-integer constrained optimization problem is stated and solved with the human intervention, where the user assists the evolutionary algorithm to reduce the noisy image variance under the restriction of keeping the mean value of the noisy SAR image within a predetermined interval of acceptance. We apply the methodology to a set of synthetic and real SAR speckle corrupted images. The results through the evaluation of objective global and local quality criteria, show the excellent potential of the proposal.

1 Introduction

It is well-known that Synthetic Aperture Radar (SAR) systems generate the images through coherent processing of the scattered signals and therefore, are susceptible to speckle. Speckle is different depending on the image adquisition and processing system, but its appeareance resembles a characteristic granular pattern.

Speckle in SAR images is assumed to be modeled as a multiplicative noise [1], although there is a small additive component that it is usually ignored. SAR systems show a limited bandwidth and, filtering is needed to obtain fine detail images even for images having low speckle. Although speckle can be useful to extract information from SAR images (it is not truly a noise in the signal processing sense), effort has been constantly put to suppress it because it reduces the efficiency of post-processing steps such as image segmentation or classification. Moreover, speckle makes SAR images more difficult to interpret.

The goal for a despeckeling filter consists of suppressing the speckle while preserving all the scene features such as texture, point-type targetss and especially edges. A large body of literature exists on speckle filtering for SAR images. Several conventional speckle filtering algorithms, such as, low pass and averaging filters [2], edge preserving [3] and filters based on underlying image statistics [4], [5] and diffusion-like filters [6] have been applied during last years.

Other filtering methods are based on the wavelet transform [7], or work in the curvelet domain [8] or even some existing methods combine both domains [9]. Authors in [10] apply the multiscale mirror-extended curvelet transform and the global search particle swarm optimization method to reduce speckle. This method also enhances edge features and notably improves the constrast of the SAR images. However, as happens with most curvelet methods, despeckled images show the typical pseudo-Gibbs ripple-like artifacts.

Regarding the speckle reduction in SAR images, several methods using regularization schemes have

been proposed in the recent years [11],[12],[13]. These methods deal directly with raw complex-valued SAR data and they are intended to provide high-resolution and default-free SAR images. These methods rely closely to initial data SAR acquisition and signal processing domain by including SAR specificities in the models. Consequently, one obtains high quality SAR images with excellent edge and feature preservation and showing low speckle. These image restoration methods typically model the problem by using quadratic functions and embedding them in the well-known Tikhnovov regularization approach [11], or variants [14], [15]. To establish a robust constrained optimization problem a set of constraints and prior information related to features is needed. Some post-enhancement algorithms are also applied to these SAR image formation methods.

It is important to note that filters based on a partial differential scheme (this also applies to diffusion-like filters) are iterative and, if not well designed, they produce over-filtered images showing notably edge degradation and removing structural details. The correct setting of the several regularization parameters involved is also a notably issue as well as their high computational cost required to get optimal solutions. It is interesting to note that our proposal, as any despeckling filtering proposal, will directly benefit from the improvement in SAR image formation through regularization based methods.

New approaches based on the non-local means filter to remove white Gausssian noise have appeared recently [16]. These NL-means filters update pixel value through a weighted average of surroundings pixels using slide patch windows to account for the non-local statistical estimation. Hence, NL-filters update the information to restore the noisy pixel by considering global (*non-local*) statistical information which can be estimated from a neighborhood (search space). This search space can be even the whole image, but, for practical reasons, it is reduced to a convenient neighborhood. Note that as a main difference with *local-based* filters, instead of comparing the intensity of the pixels, the non-local approach analyzes the patterns (*patches*) around pixels.

NL-filters have shown excellent performances when compared to other state-of-the art denoising filters [17], [18], [19], [20]. The main drawback of NL-filters is their computational cost. Optimal

setting of the patches size (local and search patches) is a problem not yet solved. These parameters are usually set from previous empirical testing on a set of images. Different implementations of these NL-means filters accounting for speckle removal in SAR images can be found in the recent literature. In [21] a NL-means filter is designed to reduce speckle in PolSAR (Polarimetric SAR) images by adapting the original NL-filter [16], suited for additive noise, to multiplicative noise. In [22] a new non-local filter for despeckling SAR images is addressed by extending the BM3-D (Block-matching 3D [23]) approach into a wavelet shrinkage approach to account for the peculiarities of SAR images content. This new filter, as authors mentioned, reduces significantly the artifacts related to the Gibbs phenomena.

Several NL-local filters for speckle removal in ultrasound images have been also published since the irruption of the non-local paradigm into the image processing community. In [24] by modelling the speckle in ultrasonic images by the Rayleigh distribution, and considering the maximunm likelihood estimmator, a NL-means filter to reduce speckle is designed. The performances of this new filter is compared to the conventional NL-means filter, the Lee filter, the Med-wavelet filter, and the anisotropic diffusion filter among other state-of-the-art despeckling methods. For all the cases considered, both on synthetic and real ultrasound images, the new NL-means filter performed better speckle removal and better edge and feature preservation.

In [25] authors deal with a NL-means filter specifically designed to overcome the degradation of borders and edges show by the conventional NL-means filter due to the use of square patches with fixed sizes. This problem is caused by an abrupt lack of redundancy of the image in these features. Therefore, less valuable information is available to reduce noise and visually appreciable residual noise around edges remains unfiltered. In [26] a family of different patches with different shapes (rectangular, square, disk, half-circle, quarter-circle) , different sizes and different orientation is proposed and embedded in the conventional NL-means filter to reduce additive White Gaussian noise in noisy images. This multi-shape patch approach is combined with a Gaussianization preprocess to reduce speckle in ultrasound images in [25]. In our case we take the multipatch approach, in that we com-

bined different patch sizes for the search space and for the local neighborhoods, when performing the random selection of variables, as explained in Section 3.

Recent years have seen how Bayesian filters emerge as the natural extension of the NL-means filters, providing a general framework to deal with multiplicative (speckle) noise. The Bayesian Non-Local means (BNL) filter ([27] [28]), aims to reduce the Bayesian risk under the assumption of estimating the prior probability on the noisy image patch. Bayesian-like filters are recognized as excellent filters in terms of mean, edge and, detail preservation but they demand high computational cost. Bayesian filters outperform other efficient despeckling filters such as the Speckle Reduction Anisotropic Diffusion (SRAD) filter [6] as it is discussed in [29]. The better performance of BNL filter over the improved sigma filter and the PPB (Probabilistic Patched-Based) filter is shown in [30].

Some efforts have been focused on reducing this cost. In [29] a new proposal to adapt the conventional NL-means filter to the statistical particularities of ultrasound medical images is discussed. In this case, the OBNLM (Optimized Bayesian Nonlocal Means) filter is proposed, having two main contributions. On one hand, the replace of the initial patches by the named blockwise approach, which contributes to low computational cost. Note that the conventional NL-means filter can be seen as a *pixel-wise* based filter, and this new approach can be regarded as a *blockwise* NL-means filter. On the other hand, a new statistical distance for patch comparison is derived: the Pearson distance. Results show better performances when comparing with the set of usual despeckling filters.

The enhanced version of the BNL filter is the Bayesian Non-local means filter with sigma preselection (EBNL), which combines the well-known sigma preselection mechanism from the improved sigma filter [5] with the PPB filter [21]. The PPB filter corrects the estimation of the prior probability done by the BNL filter and estimates this probability in a data-driven manner, providing a better estimate for the Bayesian risk. The sigma preselection assures to benefit from the original improved sigma filter, that is, to correct the bias problem related to the issue that the N-looks probability distribution is far from being symmetric.

For all the despeckling filtering methods the major problem is to reach a trade-off between the output image resolution and the speckle removal. The desired result is a filtered image showing a miminum standard deviation while keeping the original mean value of the noisy image. It is known that such a filtered image is an enhanced despeckled version of the original noisy image.

To optimize a filter, there are several numerical methods such as gradient-like and quasi-Newton methods but, although providing acceptable solutions, such solutions are susceptible of showing undesirable artifacts or deformed borders or simply of lossing relevant image details. Additionally, as direct numerical methods tend to get trapped into local minima, in this work, we propose to apply a supervised methodology [31] based on an interactive genetic algorithm (IGA) to guide the efficient design of the Enhanced Bayesian Non-Local filter (EBNL filter) [30], under the supervision of a user (*the filter designer*). The desired result is a filtered image showing a miminum standard deviation but keeping the original mean value of the noisy image. A constrained optimization problem is stated and solved by the human intervention, where the user assists the evolutionary algorithm under the restriction of keeping the mean value of the noisy SAR image within a predetermined interval of acceptance.

From a user's standpoint, running the interactive algorithm that employs less interaction in order to reach a suitable solution is to be preferred [31]. In principle, GAs, as global optimization methods, are less prone to get stuck into local optima, and therefore find good solutions in just one run with no need for restarts. Moreover, in our tests, it was found out that the IGA produced good solutions in one run for all images tested. Designers of interactive methods should strive to reduce tediousness - that is, reduce the number of interactions a user make in order to guide the algorithm towards a suitable solution. Thus, GAs area clearly preferred to local search with random restarts.

Interactive Evolutionary computation has not only been applied successfully when just a subjective assessment of the optimization criterion is available, but also in cases where the end-users are not familiar with processing or optimization techniques leading to the result. One example is the interactive

evolutionary computation applied to image enhancement and filtering of medical images where doctors are not familiar with details of image processing techniques but are interested in the end result: a filtered or enhanced image that offers sufficient information to proceed with the medical analysis [32], [33]. Interactive evolutionary computation has been extensively applied to relevant engineering problems in speech recognition [34], data mining [35] among others applications but we are not aware of its application to optimize the EBNL filter.

The paper is organized as follows: firstly, we present in Section II the Bayesian Non-local means filter, the improved version (the EBNL filter), and its related control parameters. Secondly, in Section III the main aspects regarding the interactive genetic algorithm are discussed. In Section IV some experimental results on synthetic SAR images and on real SAR images are shown and the quality of the achieved results are evaluated through a set of well-known statistical estimators. Finally, some conclusions are drawn.

2 Enhanced Bayesian Non-local Means Filter with Sigma Preselection

Our methodology applies the EBNL filter [30] combined with an interactive genetic algorithm to guide the design of the filter. We call *O*-EBNL filter (Optimized BNL filter) to the optimized BNL filter embedded into the evolutionary algorithm.

In this section, first we summarize the main features of the BNL filter [27] and then, we describe its enhanced version [30]. While describing the enhanced BNL filter, we focus on the optimization of the filter's parameters through an evolutionary algorithm.

2.1 Bayesian Non-local Means Filter

The BNL means filter is the extension of the NL-means filter based on Euclidean-distance criteria [36] and on a probabilistic approach which can be also applied to multiplicative noise (speckle-like noise).

BNL filters, in general, proceed by minimizing the Bayesian risk under the assumption that the statistical estimations from an image patch are valid, that is, good aproximations for the true intervening statistical parameters. Besides, BNL filters better preserve edges while reducing the standard deviation and keeping the original mean almost unchanged [30], as compared to other well-established despeckling filters [5, 6].

For a noise-free image u, and a noisy image v, the BNL filter [27] proceeds updating the noisy data at pixel $v(x)$ as,

$$\widehat{u}(x) = \frac{\sum_{y\in\varepsilon(x)} p(\mathbf{v}(x)|\mathbf{u}(y))p(\mathbf{u}(y))u(y)}{\sum_{y\in\varepsilon(x)} p(\mathbf{v}(x)|\mathbf{u}(y))p(\mathbf{u}(y))}, \tag{1}$$

where $\widehat{u}(x)$ is estimated as the weighted average of all gray values $u(y)$ in the local neighborhood $\varepsilon(x)$ of x. $\mathbf{v}(x)$ and $\mathbf{u}(x)$ are the vectorized $M\times M$ image patches centered at pixel x. Under the assumption of fully developed and non-correlated speckle samples, the conditional distribution $p(\mathbf{v}(x)|\mathbf{u}(y))$ can be estimated by means of a Gamma distribution (see [28]) as,

$$p(\mathbf{v}(x)|\mathbf{u}(y)) \propto \exp\left(-\frac{1}{\rho^2}\sum_{m=1}^{M\times M}\left(\frac{v_m(x)}{u_m(y)} + \ln\frac{u_m(y)}{v_m^{L'}(x)}\right)\right), \tag{2}$$

where ρ is the smoothing parameter (a *decay* filter parameter), and $L' = (L-1)/L$, with L the number of looks for an intensity SAR image (a similar expression can be obtained for the amplitude case). The smoothing parameter ρ can be related to the number of looks, L, through $\rho = k/\sqrt{L}$, where k is an empirical factor. A good choice for the k value is $k \approx 2$, although in our proposal, k is regarded as a filter parameter to be optimized. The description of the BNL filter is completed by establishing the prior distribution $p(\mathbf{u}(y))$, which is inversely proportional to the size of the neighborhood $\varepsilon(x)$ of x, $|\varepsilon(x)|$, i.e., $p(\mathbf{u}(y)) = 1/|\varepsilon(x)|$.

2.2 Enhanced Bayesian Non-local Means Filter

Authors in [30] propose a variation of the BNL filter described above in order to reduce the bias problem due to the use of $\mathbf{u}(y)$ and $u(y)$ - which are not available in a real case - instead of $\mathbf{v}(y)$ and $v(y)$. The EBNL filter is:

$$\widehat{u}(x) = \frac{\sum_{y \in N(x)} p(\mathbf{v}(x)|\mathbf{u}'(y)) p(\mathbf{u}'(y)) u'(y)}{\sum_{y \in N(x)} p(\mathbf{v}(x)|\mathbf{u}'(y)) p(\mathbf{u}'(y))}, \tag{3}$$

where the *a priori* mean estimation, $u'(y)$ is done using a 3×3 mean window, which shows better performances in retaining details than using a larger window, and it is also faster. In our proposal, we apply this criterium instead of the iterative method proposed in [27]. However, to avoid the blurring of the strong reflective scatterers, we apply the same criterium used in the improved Lee sigma filter [5]. This criterium consists of the following: the 98th percentile Z_{98} of the SAR data within the image is first estimated. Next, for a x-neighborhood (3×3), if more than T_k pixels (including the center pixel) have values greater that Z_{98}, pixels in the neighborhood will not be filtered.

A typical value for the threshold Z_{98} is between 5 and 7, but in the present work we employ it as a filter parameter to control the visual appeareance of the image and the Z_{98} percentile we also employ it as a filter parameter (namely, *Threshold*, T_H).

2.3 Sigma Range Preselection

Note that equation (3) also differs from (1) in the neighborhood $N(x)$, which is a refined subset based on the sigma range, to account for the pixel preselection, as detailed in the following:

$$N(x) = \varepsilon(x) \cap N_1(x) \cap N_2(x), \tag{4}$$

where the set $N_1(x)$ is a set of pixels obtained by patch preselection to eliminate the uncorrelated pixels within the $\varepsilon(x)$ patch, and the set $N_2(x)$ accounts for pixel preselection. The subset $N_1(x)$ is calculated as,

$$y \in N_1(x), \text{ only if } \gamma < \overline{\mathbf{v}}(y)/\overline{\mathbf{v}}(x) < 1/\gamma, \tag{5}$$

where $\overline{\mathbf{v}}(\cdot)$ stands for the mean value of the corresponding patch $\mathbf{v}(\cdot)$ and $\gamma < 1$ is a threshold defined by the user. In our proposal, we consider this parameter as a filter variable to be optimized. Note that this parameter strongly affects the computational cost involved when filtering an image. Moreover,

this parameter is relevant as a bad choice for its values reduces the efficiency of the overall filtering process.

The other set appearing in (4), namely $N_2(x)$, is the the set of pixels obtained by pixel preselection. Pixel preselection based on the sigma range [5] is a useful technique for retaining details and for preservation of strong reflective scatterers, which are usually of high interest when filtering SAR images.

It is also useful for excluding outliers during the calculations. The sigma range preselection reduces the speckle -smoothes the image noise- by means of averaging *only* those neighborhood pixels showing intensities within a fixed sigma range of the center pixel. Note that the sigma range preselection takes into consideration the asymmetrical distribution of speckle probability density function, which removes the well-known bias problem related to the original sigma filter.

For SAR images, the sigma value, ξ usually ranges from ξ = 0.8 to ξ = 0.9 as indicated in [5] for terrain and crop classification, although, for some applications, a lower value is preferred. In our proposal, we consider ξ as a filter parameter spanning *continuously* a user defined interval. Section 3.4, provides a simple numerical scheme to obtain the sigma range for a given continuous ξ value.

To account for dark areas and isolated black dots, we follow the approach from [30], which consists of separating the pixels with a threshold, $T = v_{max}/2$; pixels with intensity values higher than T are preselected through the sigma range mechanism,

$$y \in N_2(x), \text{ only if } v(y) \in (u'(x)I_1, u'(x)I_2), \tag{6}$$

where I_1 and I_2 are the ranges satisfying the condition $\xi = \int_{I_1}^{I_2} p_N(I)dI$, for a N-looks intensity pdf $p_N(I)$.

2.4 EBNL Filter Parameters

In the previous section we have discussed the BNL filter and its enhanced version, the EBNL filter as well as the refinement of the EBNL by means of considering the sigma preselection mechanismn.

The filter parameters chosen for filter optimization are: the k value (related to the smoothing parameter), the γ parameter (related to the patch preselection), and the ξ parameter associated to pixel preselection. Due to its high computational cost when using large search windows patches (15×15, 21×21), the BNL and the EBNL filter perfom in one iteration.

However, if one considers smaller search window patches, the filter can be applied iteratively. Therefore, we consider the maximum number of iterations, n_{max}, as another filter parameter. The size of the vectorized patch M, that is SM, and the size of the neighborhood $N(x)$ of x, that is SN, are also considered as filter parameters to be optimized. We gather them in a decision vector $\mathbf{z} = [k, \gamma, \xi, T_k, T_H, n_{max}, SM, SN]$.

3 Optimization and Evolutionary Algorithm

For a given intensity noisy image I_v with support Ω and mean $\mu(I_v)$, and standard deviation $\sigma(I_v)$, we want to optimize the EBNL by solving:

$$\begin{aligned} &\text{minimize } \sigma(I_v(\mathbf{z})) \\ &\text{subject to } \mu(I_v(\mathbf{z})) = \mu(I_v), \end{aligned} \tag{7}$$

where $\mathbf{z}$ is the decision vector.

The desired result is a filtered image showing a miminum standard deviation but keeping the original mean value of the noisy image. It is known that such a filtered image corresponds to the enhanced despeckled version of the original noisy image.

The problem to be solved is a nonlinear contrained optimization problem (*probably*) non-convex. Additionally, as some components of the **z** vector admit real values (k, γ, ξ, T_k) an others admit only integer values (n_{max}, SM, SN), the problem to be solved is a mixed-integer constrained optimization problem (MIP problem, Mixed-Integer Programming problem).

This kind of optimization problems are usually solved by branch and bound techniques and its resolution implies a high computational cost. We are not aware of any attempt to optimize the EBNL filter stated as in equation 7, not even a relaxed formulation of it using only real variables.

There are several direct numerical methods to solve a relaxed (continuous) formulation of the above problem to get an *acceptable* non-optimal solution, such as gradient-like methods, and quasi-Newton methods. Using direct methods can be risky because, although they provide acceptable solutions satisfying the restrictions with low computational costs, those solutions are susceptible of showing undesirable artifacts. Such solutions may also provide deformed borders or simply may lose relevant image details such as the typical extra bright scatterers found in real SAR images. Besides, good filters such as curvelets and diffusion-like filters, tend to overfilter the image, causing a notable blur effect, when a restriction like the one in equation (7) is considered and no upper limit in the number of iterations is taken into account.

To avoid these shortcomings and with the purpose in mind to deal with the natural formulation of the filter optimization in the mixed-integer domain, we propose a supervised methodology to guide the efficient design of the EBNL filter while retaining image details. In addition, as direct numerical methods tend to get trapped in local minima, we prefer to use a heuristic method. Among the several heuristic methods available in the technical literature (for instance: tabu search, simulated annealing, Monte Carlo), we choose an evolutionary methodology which is detailed in the next section.

3.1 Interactive Genetic Algorithm

The basis of genetic algorithms (GAs) can be seen as the intelligent -highly efficient- exploitation of a selective random search inspired by the natural evolution process [37]. GA employs a population P of individuals z_j (*chromosomes*) and evolves this population through the application of random variation and selection operators.

It is clear that the larger the population P, the higher the probability to reach the optimal solution to the given problem, at the expense of increased computational cost. Therefore, it is desirable to establish a tradeoff between these variables (size of P population and computational cost) to make the evolutionary algorithm efficient in terms of providing near-optimal solutions in a reasonable time.

A population, P, with its corresponding chromosomes z_i, (potential solutions to the stated optimization problem:the EBNL filter design), is defined as the set, $P = \{z_i\}$, with, $z_i = (z_{i1}, ..., z_{il})$ as a vector of l genes, $z_{ij} \in \left[vlb_j, vub_j\right] \subset \Re$, and $i \in \{1, 2 \ldots, N\}$, $j \in \{1, 2 \ldots, l\}$; vlb_j and vub_j stand for the lower and upper bound for the values of the genes, respectively.

In our EBNL filter implementation, the chromosome corresponds to the decision vector,

$\mathbf{z} = [k, \gamma, \xi, T_H, T_k, n_{max}, SM, SN]$,

and the lower and upper bounds are given by the set $[(1.5, 30.0), (0.70, 1.0), (0.50, 0.99), (0.70, 0.98), (5, 8), 3]$ for the variables $k, \gamma, \xi, T_H, T_k, n_{max}$, while the size patches may take values in: $[(3 \times 3, 5 \times 5, 7 \times 7)]$ for the search window patch and in $[(7 \times 7, 9 \times 9, 11 \times 11)]$ for the vectorized patch. Both sets have been chosen through a training phase over a set of real SAR images.

Authors in [27] and in [16] suggest to use a 15×15 patch for SN (search window) and 7×7 for the vectorized patch due to its proven detail preservation performance. However, in [30], the patches used are 21×21 for the SM patch and 7×7 for the SN patch.

The maximum number of iterations, n_{max} can be fixed to this small number (3), because similar filter realizations of the non-optimized EBNL filter can be obtained just by conveniently selecting the patch sizes and the other filter parameters (see Section 4).

From (7), the objective function to be optimized is $\sigma(I_v(\mathbf{z}))$, but, using the Interactive Genetic Algorithm (IGA), we solve the problem by minimizing the *fitness function* F, in the GA context. This fitness function at iteration i, is evaluated visually by a human operator for a given decision vector, $\mathbf{z}_i$, and gives a score $z \mapsto F(z) \in [0,10]$. The IGA works like a parameter adaptation algorithm producing and evolving various filter realizations according to a subjective quality criterion. The user evaluates every filtered image attending to its variance reduction and its constant mean value which are shown on screen jointly with the original (noisy image) and the filtered image after each iteration (see Fig. 1).

The equality contraint on equation (7) is relaxed by an acceptable level of approximation (the restriction is considered satisfied by any mean value, $\mu(I_v(\mathbf{z})) \in [\mu_{min}, \mu_{max}]$). The input parameter values μ_{min} and μ_{max} are provided by the user according to his/her preferences and they are evaluated within a set of Regions Of Interest (ROI) selected by the user. The evolutionary algorithm takes into account - *and asks for interactive evaluation*- only for the solutions satisfying the imposed constrains. Note that the rate of convergence of the algorithm depends on the size of the interval $[\mu_{min}, \mu_{max}]$.

As it can be seen in Fig. 1, which corresponds to a running sesion at iteration i, the user can reject a given filter solution $\mathbf{z}_i$ by assigning a low score (or even a null score) to the actual evaluation. Consequently, in later iterations the set of solutions producing such filtered images can be hardly found and, as the algorithms evolves, they are completly eliminated from the population. This is how the IGA evolves to the desired solution demanded by the user through visual evaluation.

It is expected that, by randomly mixing (*combining*) chromosomes ($\mathbf{z}_i$ possible optimal solutions) through the two basic variation operators, crossover and mutation, the algorithm evolves increasing

the average fitness of the population. Although the optimal solution is not guaranteed, it is also expected that the final population contains a near optimal solution, that is, a selection of the best results (individuals) is accomplished in each algorithm iteration.

3.2 Crossover and Mutation Operators

Selection of the best solution (EBNL best filter design for our purpose) is done through a probabilistic mechanism, which chooses -selects - the solution among the current generation and passes it to the next generation. The crossover variation operator is a stochastic process with a probability P_c (a parameter of our GA implementation), that for any pair of chromosomes in P, randomly combines the chromosomes (called *parents*) using a linear combination to yield a new pair of individuals (denoted as *offspring*).

The second operator, the Gaussian mutation [37], applies to all individuals within P with some probability P_m (another parameter of our GA implementation). Each gene z_{kj} in P has a probability P_m to be mutated (modified) according to a normal distribution centered at the z_{kj} value. Therefore, the mutation operator provides random changes in the population, which are required mostly in the first iterations stages of a run, in order to explore, as much as possible, the whole domain set. It also helps to preserve a reasonable level of population diversity which enables the process to escape from sub-optimal regions of the solution space. In the final stages of the run, mutation works also as a fine tunning variation operator.

Note that the choice of the parameters' values related to both operators, the crossover and the mutation, must be carefully suited to the problem to be solved, that is, although genetic algorithms are well-known for their robustness, they demand a subtle and profound knowledge of the problem to deal with.

3.3 Clustering

As mentioned above, a human interpreter (*the user*) visually evaluates the quality of the input image that has been filtered using the actual z_k solution and by taking into account the statistical estimators (see Fig. 1) and next provides a score: $z \mapsto F(z) \in [0,10]$. Obviously, the total number of function evaluations to get a satisfactory solution is certainly big, and the interactive procedure becomes infeasible.

We reduce this interactive design phase by employing a clustering strategy. We focus on reducing the number of images the user has to evaluate subjectively without affecting the user control on the process of image evaluation. Each chromosome z_k to be evaluated is placed in a separate table or *history of past evaluations* Φ, with a zero fitness initial value: $F(z_k) = 0$. The entries in Φ are tuples $(z_i, F(z_i))$. All members of Φ undergo a clustering procedure. Clustering applies to the first elements in the tuples (z_i), represented as a point in the *n*-dimensional space. For instance, if z_k falls within some cluster χ, $F(z_k)$ is estimated (*updated*) as the average fitness of chromosomes in χ excluding z_k itself, as follows:

$$F(z_k) = [1/(|\chi| - 1)] \sum_{j=1, j\neq k}^{|\chi|} F(z_j). \tag{8}$$

If χ is a one element cluster then the evaluation of z_k is deferred to the human evaluator (the *user*).

In the first generation (first iteration) all chromosomes in the pool (*population*) are evaluated subjectively. This task assumes, for instance, 10 evaluations for a setting of 10 predefined maximum clusters, which is not too computationally expensive for the EBNL filter using the parameters disccussed above. In subsequent generations of solutions (*iterations*), chromosomes are evaluated subjectively with some periodicity λ (a parameter of the algorithm suited to the availability of the user). The rest of the chromosomes are evaluated using the clustering strategy described. Note that the evaluation of the fitness function is performed by the user at the first iteration by visually assigning a numerical value according to his/her preferences. In subsequent iterations *some* evaluations are required to update the value of $F(z_k)$, enriching the filtering design with the subjective component.

An agglomerative hierarchical tree clustering methodology with a consistency level, κ, has been applied to form the individual clusters from the dendogram [38]. This means that similar solutions are clustered according to their euclidean distance in the representation space. The parameter $\tau \in [1, n_{max}]$ works as a cutoff for the dendrogram the method employs. Cutting the dendrogram at a smaller cutoff level usually produces many clusters with smaller values of inconsistency [39], where inconsistency is defined over the computed linkage of the tree. Single linkage was used due to its computational simplicity [38]. We took smaller values for τ as this leads to clusters which are consistent.

To fully complete the description of the proposed IGA, an elitist replacement of the worst individual in each generation with the previous best individual has been used to prevent loosing the best enhanced BNL filter realization. This is a common approach taken not only by genetic algorithms, but also by heuristic methods to take advantage of the exploration search and to assure that at least, a better solution (from the initial set) is always guaranteed.

3.4 Numerical Estimation of the Sigma Range

In [5] a numerical method is detailed in order to estimate the sigma range. The method calculates the sigma range $[I_1, I_2]$, for a given ξ value and the N-looks intensity pdf, $p_N(I)$, by solving the pair of integral equations:

$$\xi = \int_{I_1}^{I_2} p_N(I) dI \tag{9}$$

under the constrained of keeping the mean value within the interval $[I_1, I_2]$,

$$\bar{I} = \frac{1}{\xi} \int_{I_1}^{I_2} p_N(I) dI. \tag{10}$$

As it is not possible to find a closed solution for the above pair of equations, an iterative method is applied. For the sake of clarity we will detail the method in [5] as follows. Suppose $\bar{I} = 1$. In the first

interation, I_1^1 is set to 0.5, and the upper range, I_2^1, is obtained from (9). From that, a new value for $\overline{I}^1$ is calculated using (10). If this new value is smaller than the desired value ($\overline{I} = 1$), then, a new value is computed as $\overline{I}^1 = \overline{I}^1 - 0.001$. In case of being greater, the new value is obtained as $\overline{I}^1 = \overline{I}^1 + 0.001$. After several iterarions, it is clear that one gets the result $\overline{I}^{(k)} = \overline{I}$, at iteration k.

From this method, the sigma range, once calculated, is stored in look-up tables according to the sigma value and the number of looks associated to the SAR image for its further use in the improved sigma filter [5] or in the EBNL filter. Some sigma ranges taken from [5] are shown in Table 1 for the *1*-look and *2*-looks intensity cases and for several sigma values. A similar treatment applies to the amplitude case.

In our proposal, as the filter parameter ξ takes continuous values in user-defined interval $[\xi_{inf}, \xi_{sup}]$, there is a need to efficiently calculate the sigma range for each of the ξ values employed in the evolutionary algorithm. Therefore, it is necessary to apply a numerical faster method. In Figure 2 it can be seen that a simple second order polynom can be used to estimate the inferior sigma range I_1 value for the whole range of ξ values in the *2*-looks intensity case.Similar results are obtained for other cases normally found in SAR image processing (*1*-look intensity, *3*-looks intensity and *4*-looks intensity).

Once the inferior sigma range I_1 is estimated from the fitting polynom (the initial searching interval has been notably streched), the bisection method is applied to get I_2 from (9). A similar iterative procedure as in [5] is applied to estimate I_1 from (10). The method converges to the same sigma-ranges values but at lower computational cost, which is relevant for our proposal due to the multiple evaluations accomplished through our evolutionary optimization proposal.

3.5 Implementation Issues

The original BNL filter, the E-BNL filter and the evolutionary algorithm to design the filter have been embedded using a software package with a friendly graphic interface implemented in MATLAB R2008a [40]. The graphic interface is illustrated in Fig. 1. As it can be seen, two images are provided

to the user: the one on the left, image U, is the original (*noisy*) image, and on the right, the actual image obtained by the filter implementation $\mathbf{z}$ at iteration i is shown. Having these two images, the user is asked to evaluate (assign a 0-10 score) the filtered image V.

The original mean and original standard deviation estimated within the whole image or through a ROI are shown below the original image and, the actual corresponding values and their percentage as compared to the original ones appear under the image filtered using the actual filter realization.

The user can evaluate the image V attending to his/her visual preferences while taking into account a quantitative criteria too (the mean preservation and the desired variance reduction). It is important to remark that only the images satisfying the mean restriction (see equation 7) are considered for subjective evaluation. In this sense, this considerably disminishes the user workload when tuning the filter. When the evolutionary algorithm converges to the desired solution, the final filtered image and the final design vector $\mathbf{z}$ appear on the graphic interface. Although not appearing in Fig. 1, a set of statistical estimators (see next section) can be applied to the results directly from the interface.

To design a filter, the user runs the application on an input SAR image, and, after a few iterations running the IGA algorithm (scoring the images according to his/her quality criteria), an output image suited to the desired particular needs is obtained. A typical design sesion takes around 15 minutes on a Pentium-IV 2.3 GHz machine. The application has been checked by different SAR users and they agreed upon its suitability and its ease of training.

4 Experimental Setup

To validate the proposal, we have done some experiments both on synthetic SAR images corrupted with speckle and on real intensity SAR images. We compare the resuls with the EBNL filter using the vectorized patch $SM = 7, (7 \times 7)$ and the search window patch $SN = 21, (21 \times 21)$ patches as indicated in [30], and the parameters $k = 2.0, \gamma = 0.90, \xi = 0.90, T_H = 0.98, T_k = 7, n_{max} = 1$. As previously shown in [30], the EBNL filter shows better performances than other despeckling filters, such as the

Lee sigma improved filter, the BNL filter or the PPB filter.

In this section, we extend the comparison of the *O*-EBNL filter (an by default, the EBNL filter as well) with other state-of-the-art despeckling filters: the SRAD and the wavelet-based MAP-LG (Maximum A Posteriori Laplacian-Gaussian) filter [41]. For the SRAD filter, the number of iterations is set to 5 and the rest of parameters are set to their default values as indicated in [6].
The EBNL filter, the *O*-EBNL filter, and the SRAD filter have been implemented in Matlab. The wavelet-based MAP filter has been executed on-line from [42] which is a website devoted to SAR images providing a set of different despeckling filters.

4.1 Estimators to evaluate the performances of our proposal

For the simulated SAR images a quantitative analysis is performed through well-known statistics estimators: mean, standard deviation and the well-established estimators:

- **Structural Simlarity Index** (*SSIN*): measures the similarity between the original and the despeckled image through a local statistical analysis of the image using the mean, the variance and the covariance between the unfiltered and despeckled pixel values from the sliding window. $SSIN \in (-1,1)$, and a bad similarity between the original and the despeckled image corresponds to $SSIN \mapsto -1$, while a good similarity will be indicated by values $SSIN \mapsto +1$.

- β **estimator**: is useful to compare edge preservation performance [43], as it is defined, for a given image I in amplitude format and for the degraded filtered image $\hat{I}$, by

$$\beta = \frac{\Gamma\left(\Delta I - \overline{\Delta I}, \widehat{\Delta I} - \overline{\widehat{\Delta I}}\right)}{\sqrt{\Gamma\left(\Delta I - \overline{\Delta I}, \Delta I - \overline{\Delta I}\right)} \cdot \sqrt{\Gamma\left(\widehat{\Delta I} - \overline{\widehat{\Delta I}}, \widehat{\Delta I} - \overline{\widehat{\Delta I}}\right)}}, \tag{11}$$

  where ΔI and $\widehat{\Delta I}$ are the high-pass filtered version of the image I and $\hat{I}$, respectively, obtained with a sliding Laplacian pixel kernel window of size 3 x 3 or another edge detector such as the Canny detector [44]. $\overline{\Delta I}$ and $(\overline{\widehat{\Delta I}})$ are the average value of the image I and the average of the high-pass filtered version of the image $\widehat{\Delta I}$, respectively.

The definition is completed with,

$$\Gamma(I_1, I_2) = \sum_{i=1}^{K} I_{1_i} \cdot I_{2_i}. \tag{12}$$

Note that this estimator is evaluating the correlation between the ground-truth edges within the original image and the edges in the denoised image detected by means of the Laplacian filter (or the Canny filter). β ranges between 0 and 1, with unity for ideal edge preservation.

We refer the reader to the references [45] and [46] for a complete description of these estimators. Note that these estimators can be estimated due to for the simulated SAR images, a ground-truth image to compare with (the original SAR phantom image) is available, which does not occur for real SAR images. In the latter case, a filter realization can be evaluated only through the mean preservation and variance reduction (mean preservation and the variance reduction measured within a region -or measured through the entire image- indicate a *possible* successful filter operation).

For real SAR images, edge preservation can be considered as a visual requirement, that is, mere *qualitative*, indicator of good performance of a filter operation. The *Equivalent Number of Looks*, *ENL* is also estimated after filter realization for both synthetic and real SAR images. For a given image I, and a selected homogeneous ROI of I, having a mean value μ and a standard deviation σ, the *ENL* is estimated by $(\mu^2)/(\sigma^2)$.

An excellent test for a despeckling method is provided by the analysis of the ratio images as proposed in [16]. This is also becoming a standard analysis in the SAR community and it consists of inspecting the regularity pattern shown in the $\Pi = U/\hat{V}$ image, where U is the SAR original image and $\hat{V}$ is its denoised version. An ideal filter operation on image U implies that, in areas where speckle is fully developed, this ratio should show the characteristic of pure speckle. Additionally, the presence in Π of geometric structures (hand-made structures) or any detail correlated to the original image U indicates poor filtering operation, that is, not only the speckle has been removed from the original image but also other possible relevant information.

To fully validate our proposal, aspect ratio images have been obtained for both, the synthetic SAR images corrupted with speckle and for the real intensity SAR images.

4.2 Simulated SAR Images

Fig. 3 (first two images on the left) shows the *1*-look 240×160 SAR phantom, and the simulated image and the two ROI's selected for evaluating the mentioned quality estimators. The speckle has been simulated following the Gamma distribution with a mean value of 1 and fitted to provide an *ENL* = 1. As it can be noted in the simulated image, several strong scatterers have been added to appreciate the effectiveness of the filters on preserving them. The phantom consists of two well different homogeneous areas sharing a non-abrupt edge.

The filtered image by the SRAD filter, the MAP-LG filter, the EBNL filter, and by the *O*-EBNL filter are shown in Fig. 4. It is observed that all filters perfom well, preserving edges and details and reducing the image variance. Note also the preservation of strong scatterers which is more evident for the case of applying the *O*-EBNL filter. The border separating the two homogeneous regions appears with some irregularities for the *O*-EBNL filter. However, a closer look to the simulated phantom (Fig. 3, first on the left), reveals that some parts of the edge have vanished just in the points clearly visible in the result from the *O*-EBNL filter. The aspect ratio images clearly show that the best filtering operation has been performed by the *O*-EBNL filter (a more irregular speckle pattern can be noted).

Table 2 shows the quantitative performance evaluation for the filter realizations. The estimators selected to evaluate the performance of the filters were: *SSIN* for the whole image and, for the selected ROIs, the mean, the standard deviation, the *SSIN* and the *ENL* were also calculated. As it can be seen, the *O*-EBNL filter outperforms all the other filters in all the cases execpt for the case of the mean value for ROI 1, but, obtaining the second,*and very close*, best value.

The second results are for an original speckleless image which has been degraded with simulated

speckle in intensity format. Proceeding as the authors in [43], we selected the aerial image from the MATLAB's Image Processing Toolbox due to its similarities in content with real SAR images. From that image (2956×2215 pixels) a subset of 300×300 pixels was selected and it is shown in Fig. 3, (third image from the left). The selected ROI to evaluate the perfomances of the filtering operations appears also in this figure. The speckle has been simulated following the Gamma distribution with a mean value of 1 and fitted to provide an $ENL = 3$. In the same figure (right), the degraded image is illustrated.

In Figure 5 the results after applying the different despeckling filters and the *O*-EBNL using our proposal, are also shown. We can see that, that all filters provide excellent results, but a more detailed analysis indicate that the *O*-EBNL filter results a better visually despeckled image, better preserving edges (see the corresponding edge images illustrated in the same figure) and even providing a better image contrast.

Table 3 provides the quantitative performance evaluation for the whole image and for selected ROI through the estimators β, *SSIN*, and the mean value and the standard deviation value. It is clear from these results the superiority of the *O*-EBNL filter over the original EBNL filter and over the other despeckling filters.

4.3 Real Images Results

To validate our human-into-the loop proposed methodology, a set of several SAR images with speckle has been filtered. The results include a comparison with the non-optimized EBNL filter (using the standard filter realization, $\mathbf{z} = [k, \gamma, \xi, T_H, T_k, n_{max}, SM, SN] = [2.0, 0.90, 0.90, 0.98, 7, 1, 7 \times 7, 21 \times 21]$) as well as the SRAD filter and the MAP-LG filter.

The first SAR noisy image selected, *(SAR Image I)*, is illustrated in Fig. 6 (left). It is a subimage from the *1*-look HH SAR image corresponding to the Weßling area (Oberpfaffenhofen, Germany) revealing the typical speckle pattern of *1*-look intensity SAR data. This image (300×500 pixels) contains

clearcut areas, urban areas and homogeneous areas and there is are also a runway. The chosen ROI for the quantitative test corresponds to the bright area.

In Figure 7 the results after applying the different despeckling filters and the *O*-EBNL using our proposal, are also shown. Once again, we can see that, that all filters provide excellent results, but the result by the *O*-EBNL filter shows a better visually despeckled image with an improved image contrast resulting from the excellent filtering of both homogeneous areas (the bright and the dark ones). The MAP-LG filter provides excellent performances on homogeneous areas but some details are not preserved (see how the long road is blurred). Additionally, the aspect ratio images reveals a better filtering for the *O*-EBNL, although, not all the speckle is removed by any of the filters.

From Table 4, it is observed that *O*-EBNL gets the best variance reduction and an acceptable second best mean value (indeed very close to the original one). Best mean value is for the SRAD filter, but, this filter has not significantly reduced the standard deviation. This is due to SRAD filter needs also to be optimized to suit the image peculiarities. Note that the MAP-LG filter effectively reduces the standard deviation but it is not as good as the other filters to preserving the image mean. This is the reason we have added itn this table, a mark, $*$, to remark this. As a higher -but not acceptable- value of the image mean value implies a higher value for the *ENL* estimator, we have not selected this value but the one for the best *ENL* value.

Another interesting result comes for the results from despeckling the *SAR Image II* depicted in Fig. 6 (right). This image illustrates some strong scatterers that need to be preserved in the denoised image. Other man-made structures are also included in this image. The results after applying the filters can be seen in Fig. 8. In this figure, it can be seen that the MAP-LG filter does not keep these strong bright scatterers and besides, the filtered image shows the typical effects related to wavelet filters. Note this also in the corresponding aspect ratio image. The SRAD filter preserves these strong scatterers but, the best visually result comes from the *O*-EBNL filter. In this case, not only those scatterers are kept, but, the image reveals a high contrast and all the edges are well preserved. The quantitative results

for the selected ROI are depicted in Table 5, where similar conclusions to the case of *SAR Image I* are achieved.

To fully verify the efficacy of our proposal, we apply the methodology to other SAR image shown in Fig. 9. The left image, *SAR Image III*, corresponds to a 300×300 pixels intensity HH-TerraSAR X image, with a ground resolution of 1 m. The scene contains urban areas with buildigs of irregular roofs which brings once, double and multiple complicated reflections (note the strong reflect scatterers). The scenes of homogeneous areas are farms. As a difference from the *SAR Image I and SAR Image II*, it can be appreciated that this image exhibits a low contrast among homogeneous areas and regions with different textures.

The *O*-EBNL filter realization for the this image retains more the strong reflective scatterers as it can be clearly appreciated in the image Fig. 9. Note that the solution applying the EBNL filter and the MAP-LG are also excellent, but edge preserving reveals clearly better for the *O*-EBNL filter.

Table 6 contains the mean, standard deviation and the *ENL* obtained from the ROI shown in the corresponding original image. Mean preservation is very close to the original mean values for both Bayesian filters, but, SRAD shows superiority in this estimation. The MAP-LG strongly reduces the standard deviation but it overestimates the mean. However, the *O*-EBNL filter perfoms better in reducing the standard deviation and well preserving the mean within the constrain imposed by user (a variation less than a 5 % of its original value).

From above results it has been shown that our interactive proposal can be regarded as an interesting methodology to efficiently optimize the EBNL filter.

As indicated along the paper, the main goal of our proposal is to demonstrate that the performances of the standard EBNL filter can be optimized through our supervised interactive optimization approach, and thus, providing better despeckling results for SAR images, as it has been shown. The standard

EBNL filter is not computationally comparable with the SRAD filter, with the MAP-GL filter or with other well-established despeckling filters [30] . For the cases shown, the computational cost of the *O*-EBNL filter results sometimes higher and other times lower than the computational cost for the standard EBNL filter. We remark that the *O*-EBNL filter is optimized in the sense of providing better solutions both visually and quantitatively (i.e. larger *ENL*), with a reasonable computational cost (not necessary much lower).

5 Conclusions

The EBNL filter, which combines the BNL filter and the sigma preselection by Lee is recognized for its superiority as compared to other despeckling filters, preserving image details (edges and specially strong reflective scatterers) while preserving the image mean and reducing the standard deviation. Optimizing the parameters of the EBNL filter can be done in principle using a simple local search method (e.g. gradient-based) with random restarts. Deterministic optimization methods, although providing acceptable solutions, are prone to produce undesirable artifacts or deformed borders or to rub out relevant image details.

In this paper, we propose an interactive easy-to-use software tool, based on an evolutionary algorithm, to optimize the EBNL filter for despeckling SAR images. As a difference from other filter design methodologies, there is a direct implication of a user which provides a subjective validation of the filtered images guiding the filter to his/her requierements. In principle, GAs, as global optimization methods, are less prone to get stuck into local optima, and therefore find good solutions in just one run with no need for restarts.

Results carried out on synthetic images with speckle and using real SAR images show the potential of the methodology for optimizing the EBNL filter. Results obtained for the EBNL filter optimized using our IGA approach clearly show that GAs provide more suitable solutions than the gradient-based approaches using local search with random restarts.

Interactive Evolutionary computation has not only been applied successfully when just a subjective assessment of the optimization criterion is available, but also in cases where the end-users are not familiar with processing or optimization techniques leading to the result. As a remote sensing tool, SAR imagery is usually employed by end-users that are not image processing experts (earth scientists, meteorologists, urban architects, military experts, etc.) [47]. Providing a simple tool such as EBNLO capable of obtaining good filtered SAR images according to particular quality criteria with minimum image processing knowledge is clearly quite useful.

Finally, we believe that our proposal is an alternative method to exploit the potential of the EBNL filter and it is worth noting that this interactive evolutionary methodology can be adapted to other speckle reduction filter to optimize its performance.

References

[1] J. W. Goodman, "Some fundamental properties of speckle," *Journal of the Optical Society of America*, vol. 66, no. 11, pp. 1145-1149, Nov. 1976.

[2] H. C. Andrews and B. R. Hunt, *Digital image restoration,* Engelwood Cliffs, NJ: Prentice Hall, 1977.

[3] W. Hagg and M. Sties, "Efficient speckle filtering of SAR images with the EPPOS algorithm," *IEEE IGARS*, Pasadena, CA, pp. 2140-2142, 1994.

[4] V. S. Frost, J. A. Stiles et al., "A model for radar images and its application to adaptive digital filtering of multiplicative noise," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 157-165, 1982.

[5] J. S. Lee, J. H. Wen, T. L. Ainsworth, K. Chen, and A. J. Chen, "Improved sigma filter for speckle filtering of SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 1, pp. 202-213, 2009.

[6] Y. Yu and S. T. Acton, "Speckle reducing anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 11, no. 11, pp. 1260-1270, 2002.

[7] M. D. Gleich, "Wavelet-based despeckling of SAR images using Gauss-Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4127-4143, Dec. 2007.

[8] A. Schmitt, B. Wessel, and A. Roth, "Curvelet approach for SAR image denoising, structure enhancement, and change detection", in *Proc. ISPRS, CMRT09 (Paris, France)*, vol. 38, part. 3/W4, pp. 151-156, 2009.

[9] Sveinsson, J. R., "Combined wavelet and curvelet denoising of SAR images using TV segmentation," in *Geoscience and Remote Sensing Symposium, IGARSS'07*, pp. 503-506, 2007.

[10] Y. Li, H. Gong, D. Feng, and Y. Zhang, "An Adaptive Method of Speckle Reduction and Feature Enhancement for SAR Images Based on Curvelet Transform and Particle Swarm Optimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 8, pp. 3105-3116, 2011.

[11] M. Cetin and W. C. Karl, "Feature-Enhanced Synthetic Aperture radar image formation based on nonquadratic regularization," *IEEE Trans. Image Process.*, vol. 10, no. 4, pp. 623-631, 2001.

[12] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, " An iterated regularization method for total variation-based image restoration", *Multiscale Model. Simul.*, vol. 4, pp. 460-489, 2005.

[13] M. Soccorsi, D. Gleich, and M.Datcu, "Huber-Markov model for complex SAR image restoration,' *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 63-67, 2010.

[14] T. Goldstein and S. Osher, "The split Bergman method for Lq regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323-343, 2009.

[15] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1-2, pp. 89-97, 2004.

[16] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms with a new one," *SIAM Interdisc. J.: Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490-530, 2005.

[17] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patchbased image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2866-2878, 2006.

[18] Y. Liu, J. Wang, C. Xi, Y. Guo, and Q. Peng, "A robust and fast non-local means algorithm for image denoising," in *Proc. IEEE International Conference on Computer-Aided Design and Computer Graphics*, pp. 30, 2007.

[19] P. Coupé, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot, "An optimized blockwise non local means denoising filter for 3D magnetic resonance images," *IEEE Trans. Med. Imag.*, vol. 27, pp. 425-445, 2008.

[20] J. Xu, Z. Wei, and Y. Sun, "Non-local means image denoising with local geometric structures matching strategy," in *Proc. International Congress on Image and Signal Processing, CISP'09*, pp. 1-4, 2009.

[21] C. Deledalle, F. Tupin, and L. Denis, "Polarimetric SAR estimation based on non-local means," in *Proc. IEEE International Geosci. Remote Sens. Symposium (IGARSS)*, pp. 2515-2518, 2010.

[22] S. Parrilli, M. Poderico, C. V. Angelino, and L. Verdoliva, "A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 2, pp. 606-616, 2012.

[23] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080-2095, Aug. 2007.

[24] Y. Guo, Y. Wang., and T. Hou, "Speckle filtering of ultrasonic images using a modified non local-based algorithm," *Biomedical Signal Processing and Control* (Elsevier), vol. 2, no. 2, pp. 129-138, 2011.

[25] W. Chen, M. Ding, Y. Miao, and L. Luo, "Ultrasound image denoising with multi-shape patches aggregation based non-local means," in *Proc. International Conference on Intelligent Computation and Bio-Medical Instrumentacion (ICBMI)*, pp. 35-38, 2011.

[26] C. Deledalle, V. Duval , and J. Salmon, "Non-local methods with shape-adaptive patches (NLM-SAP)," *Journal of Mathematical Imaging and Vision*, vol. 43, no. 2, pp. 103-120, 2012.

[27] C. Kervrann, J. Boulanger, and P. Coupé, "Bayesian non-local means filter, image redudancy and adaptive dictionaries for noise removal," in *Proc. Int. Conf. Scale Space Methods Variationals Methods Comput. Vis.*, pp. 520-532, 2007.

[28] H. Zhong, Y. Li, and L. Jiao, "Bayesian nonlocal means filter for SAR despeckling," in *Proc. Asia-Pacific Conf. Synthetic Aperture Radar, Xian, China*, pp. 1096 - 1099, 2009.

[29] P. Coupe, P. Hellier, C. Kervrann, and C. Barillot, "Nonlocal means-based speckle filtering for ultrasound images," *IEEE Trans. Image Process.*, vol. 18, no. 10, pp. 2221-2229, 2009.

[30] H. Zhong, Y. Li, and L. Jiao, "SAR image despeckling using Bayesian nonlocal means filter with sigma preselection," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 809-813, 2011.

[31] H. Takagi, "Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation," in *Proc. of the IEEE*, vol. 89, no. 9, pp. 1275-1296, Sep. 2001.

[32] C. Munteanu, F. Morales, J. Gonzalez, J. Rosa, and L. Gomez, "Enhancing obstetric and gynecology ultrasound images by adaptation of the speckle reducing anisotropic diffusion filter," *Artifitial Intelligence in Medicine*, vol. 43, pp. 223-242, 2008.

[33] C. Munteanu, F. Morales, J. Ruiz-Alzola, "Speckle reduction through interactive evolution of a general order statistics filter for clinical ultrasound imaging," *IEEE Trans. Biomedical Engineering*, vol. 55, no. 1, pp. 365-369, 2008.

[34] Y. Sato, "Voice quality conversion using interactive evolution of prosodic control," *Applied Soft. Computing*, vol. 5, no. 2, pp. 181-192, 2005.

[35] T. Takao and I. Masanori, *Data mining from clinical data using interactive evolutionary computation*, Advances in evolutionary computing: theory and applications (Natural Computing Series), pp. 847- 861. Springer-Verlag New York, 2003.

[36] P. Coupe, P. Hellier, C. Kevrann, and C. Barillot, "Bayesian non local means-based speckle filtering," in *Proc. IEEE International Symposium on Biomedical Imaging: Form Nano to Macro, ISBI 2008*, pp. 1291-1294, 2008.

[37] T. Back, *Evolutionary algorithms in theory and practice*, New York: Oxford University Press, 1996.

[38] R. Duda, P. Hart, and D. Stork, *Pattern classification, 2nd ed.*, New York: Wiley-Interscience, 2001.

[39] R. Hogg and J. Ledolter, *Engineering statistics*, New Jersey: MacMillan, 1987.

[40] The MathWorks. 2009.

[41] F. Argenti, T. Bianchi, A. Lapini, and L. Alparone, "Fast MAP despeckling based on Laplacian-Gaussian modeling of wavelet coefficients," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 1, pp. 13-17, 2012.

[42] http://iapp.det.unifi.it:8080/.

[43] A. Achim, E. Kuruoglu, and J. Zerubia, "SAR image filtering based on the heavy-tailed Rayleigh model," Institut National de Recherche en Imformatique et en Automatique (INRIA), Report no. 5493, 2005.

[44] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 6, 1986.

[45] S. Chabrier, H. Lauren, C. Rosenberger, and B. Emile, "Comparative stude of contour detection evaluation criteria based on dissimilarity measures," *Journal on Image and Video Processing*, vol. 2008, no. 2, 2008.

[46] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600-611, 2004.

[47] T. Lillesand, R. W. Kiefer, and J. Chipman, *Remote sensing and image interpretation*, Hoboken, NJ: Wiley, 2008.

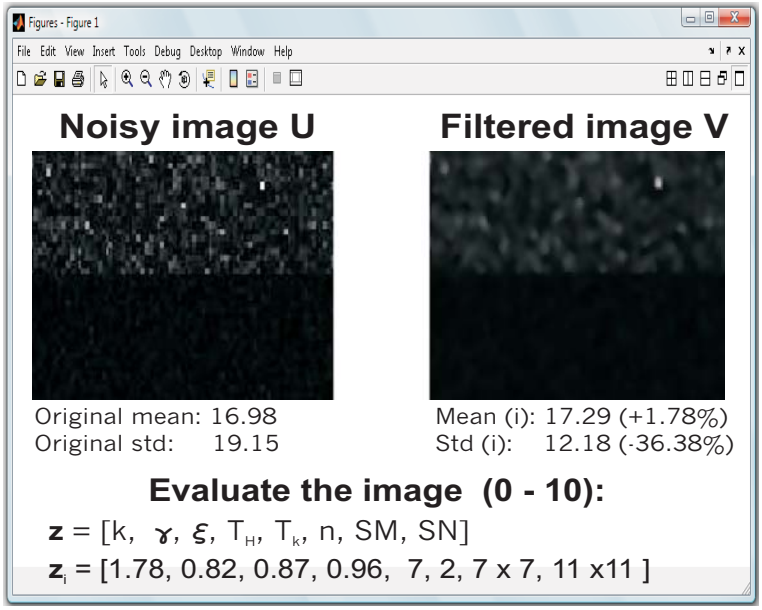

Figure 1: Interactive evaluation of the filtered image *i*.

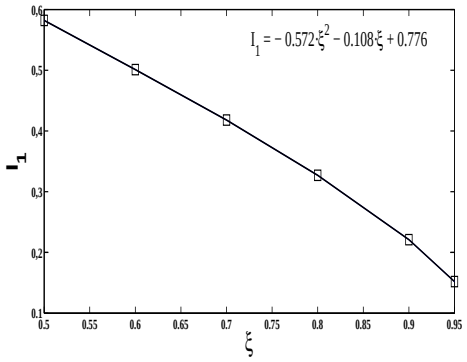

Figure 2: Second order fitting for I_1 (N = *2*-looks intensity). □: ξ values from [5], solid line: polynomial fitting.

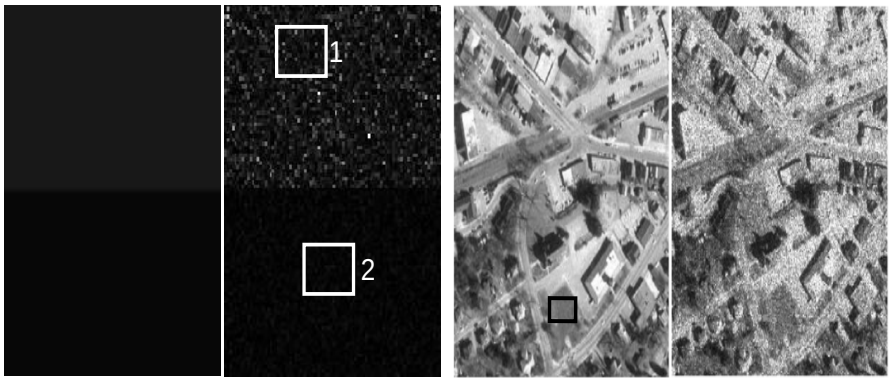

Figure 3: (Left to right) SAR phantom I, image degraded with simulated speckle (*ENL* = 1), SAR phantom II, image degraded with simulated speckle noise (ENL = 3).

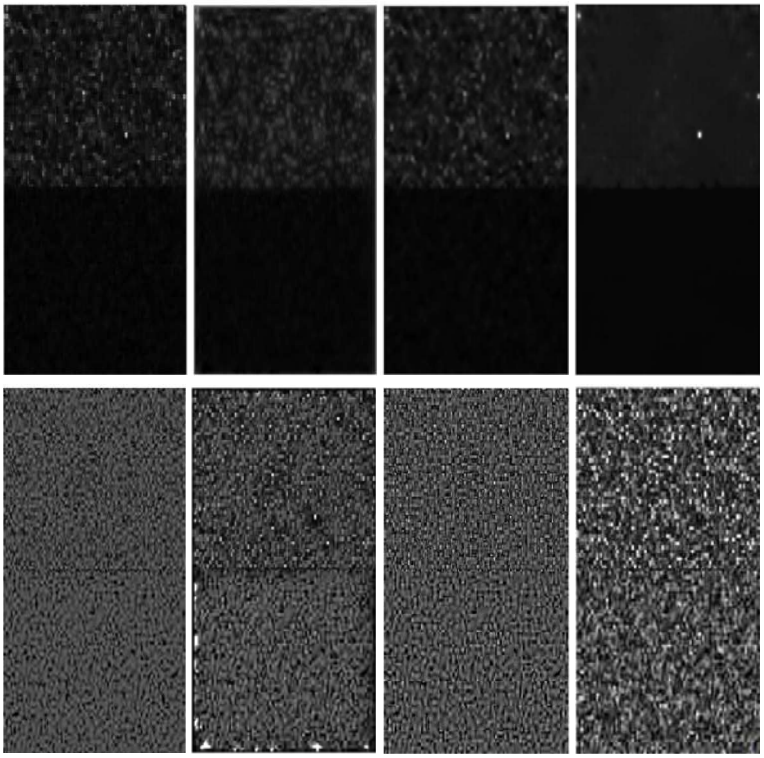

Figure 4: Results for the SAR phantom I: (top row and left to right) The SRAD filtered image, the MAP-GL filtered image, the EBNL filtered image, and the *O*-EBNL filtered result with $\mathbf{z} = [2.18, 0.61, 0.96, 0.72, 5, 2, 5 \times 5, 7 \times 7]$. The related ratio images are shown in the bottom row.

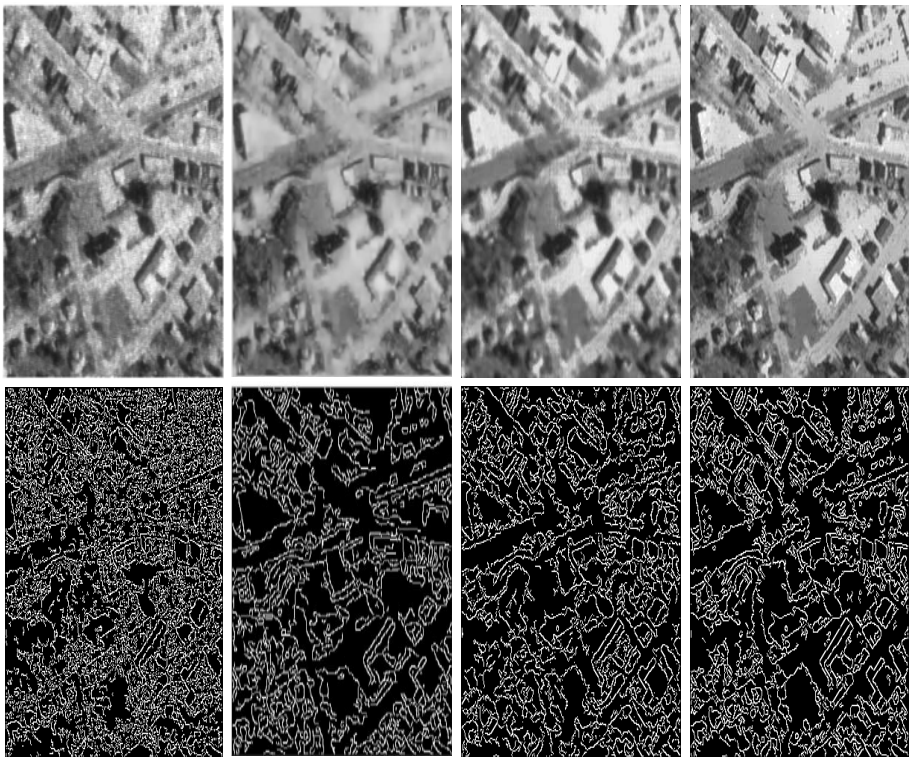

Figure 5: Results for the SAR phantom II: (top row and left to right) The SRAD filtered image, the MAP-LG filtered image, the EBNL filtered image, and the *O*-EBNL filtered result with $\mathbf{z} = [4.28, 0.84, 0.96, 0.98, 6, 1, 3 \times 3, 9 \times 9]$. Corresponding edge images respectively, using a Canny edge detector, are shown in the bottom row.

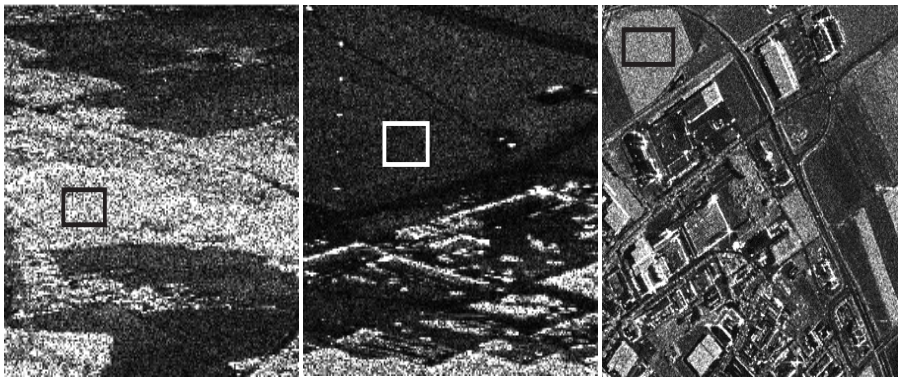

Figure 6: Real SAR images used in the experiments.(Left) *SAR Image I*. (Center) *SAR Image II*. (Right) *SAR Image III*. Note the ROI in each image.

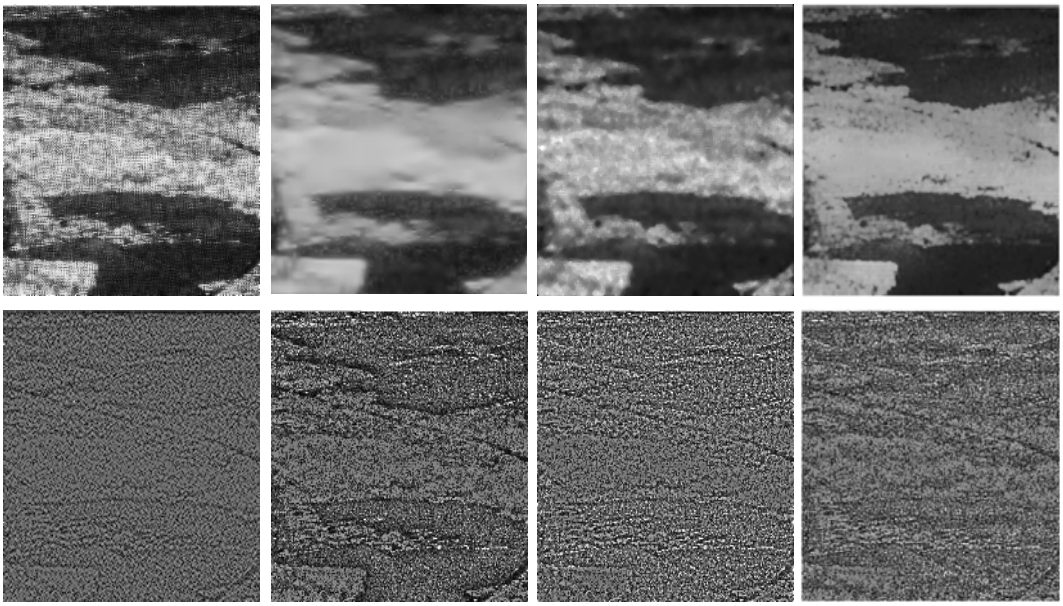

Figure 7: Results for *SAR Image I*: (top row and left to right) The SRAD filter, the MAP-LG filter, the EBNL filter, and the *O*-EBNL filter for $\mathbf{z} = [21.01, 0.58, 0.96, 0.95, 8, 1, 3 \times 3, 9 \times 9]$. (Bottom row: left to right) The corresponding ratio images.

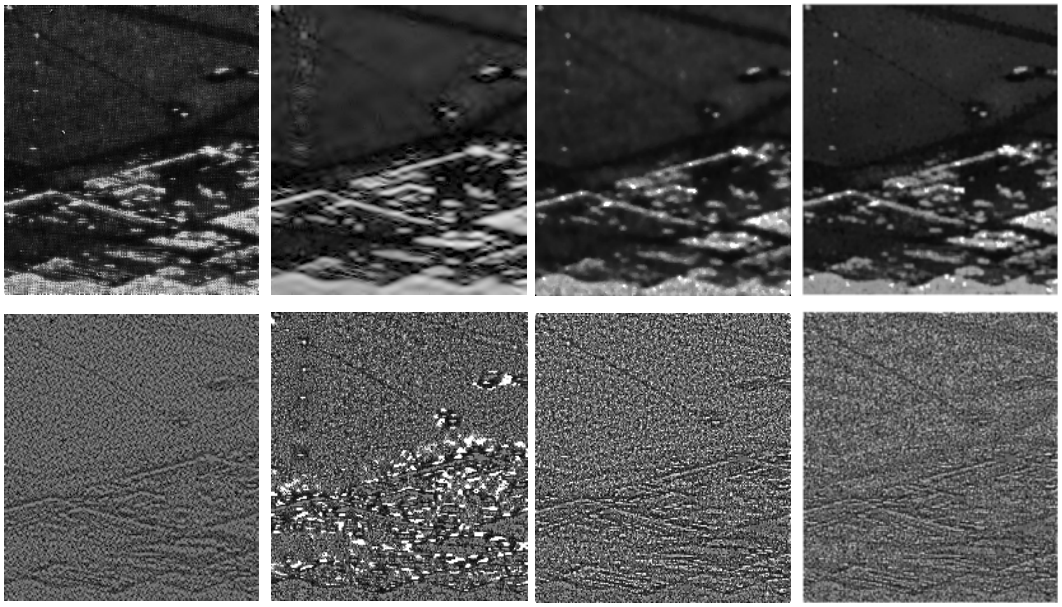

Figure 8: Results for *SAR mage II*: (top row and left to right) The SRAD filter, the MAP-LG filter, the EBNL filter, and the *O*-EBNL filter for $\mathbf{z} = [19.07, 0.65, 0.96, 0.66, 6, 1, 3 \times 3, 7 \times 7]$. (Bottom row: Left to right) The corresponding ratio images.

Figure 9: Results for the *SAR Image III*: (Left to right) The SRAD filter, the MAP-LG filter, the EBNL filter and the *O*-EBNL filter for $\mathbf{z} = [18.72, 0.85, 0.87, 0.72, 7, 1, 5 \times 5, 9 \times 9]$

Table 1: Sigma ranges for several cases (taken from [5])

1-look Intensity			***2-looks Intensity***		
Sigma ξ	I_1	I_2	**Sigma** ξ	I_1	I_2
0.50	0.436	1.920	0.50	0.582	1.584
0.60	0.343	2.210	0.60	0.501	1.755
0.70	0.254	2.582	0.70	0.418	1.972
0.80	0.168	3.094	0.80	0.327	2.260
0.90	0.084	3.941	0.90	0.221	2.744
0.95	0.043	4.840	0.95	0.152	3.206

Table 2: Quantitative evaluation for the *SAR Phantom I* (best values in boldface; second best values in italic)

SAR Phantom I		**Original**	**Simulated**	**SRAD filter**	**MAP-LG-S filter**	**EBNL filter**	***O*-EBNL filter**
The whole image	*SSIN*	—-	0.5125	0.6690	0.7504	*0.7614*	**0.9728**
ROI-1	mean	24	25.1331	25.0851	28.7519	**25.0984**	*24.6975*
	Std	—-	21.3443	13.4358	*10.3532*	10.3814	**1.8286**
	SSIN	—-	0.2022	0.4064	0.5674	0.5834	**0.9823**
	ENL	—-	1.3867	3.4874	7.7122	5.8426	**184.0345**
ROI-2	Mean	7	6.2288	6.2269	6.4158	6.2094	**6.5477**
	Std	—-	3.6619	2.4038	1.*9063*	1.9123	**0.2937**
	SSIN	—-	0.7966	0.8951	0.9310	*0.9325*	**0.9951**
	ENL	—-	2.8963	6.7314	*11.3271*	10.5435	**497.0151**

Table 3: Quantitative evaluation for the *SAR Phantom II* (best values in boldface;seco nd best values in italic)

SAR Phantom II		**Original**	**Simulated**	**SRAD filter**	**MAP-LG-S filter**	**EBNL filter**	***O*-EBNL filter**
The whole image	SSIN	—-	0.4137	0.5261	*0.5302*	0.5138	**0.6614**
	β	—-	0.2504	0.2366	*0.3596*	0.3065	**0.4527**
ROI	Mean	113.6723	111.6526	*112.3108*	111.6526	110.7235	**113.2740**
	Std	6.9728	33.6413	15.7687	7.7843	*6.5039*	**5.0682**
	SSIN	—-	0.0742	0.1988	*0.6712*	0.5138	**0.6901**
	ENL	—-	2.8963	13.8610	56.2136	*79.1909*	**136.4886**

Table 4: Quantitative filter evaluation for the real intensity *SAR Image I* (best values in boldface; second best values in italic)

SAR Image I		**Original**	**SRAD filter**	**MAP-LG-S**	**EBNL filter**	***O*-EBNL filter**
ROI	Mean	169.1220	**168.8615**	181.3066	171.2363	*168.1617*
	Std	72.6947	36.1980	*7.8577*	16.7353	**7.7555**
	ENL	1.47	5.94	*145.74 (*)*	28.60	**128.46**

Table 5: Quantitative filter evaluation for the real intensity *SAR Image II* (best values in boldface; second best values in italic)

SAR Image II		**Original**	**SRAD filter**	**MAP-GG-S filter**	**EBNL filter**	***O*-EBNL filter**
ROI	Mean	36.2050	**36.2946**	41.6171	36.4643	*36.3765*
	Std	20.5199	11.2554	*4.3090*	4.9502	**3.8369**
	ENL	0.85	2.84	*25.58 (*)*	14.82	**24.55**

Table 6: Quantitative filter evaluation for the real intensity *SAR Image III* (best values in boldface; second best values in italic)

SAR Image III	**Original**	**SRAD filter**	**MAP-GG-S filter**	**EBNL filter**	***O*-EBNL filter**
Mean	135.3119	**135.1859**	139.5422	136.1612	*134.9901*
Std	34.6285	19.6752	*4.2630*	15.5820	**3.3883**
ENL	4.1720	12.8993	*292.7682*	21.0852	**433.6936**