Accelerated Joint Image Despeckling Algorithm in the Wavelet and Spatial Domains

Adib Akl & Charles Yaacoub

Abstract

Noise is one of the most widespread problems present in nearly all imaging applications. In spite of the sophistication of the recently proposed methods, most denoising algorithms have not yet attained a desirable level of applicability. This paper proposes a two-stage algorithm for speckle noise reduction jointly in the wavelet and spatial domains. At the first stage, the optimal parameter value of the spatial speckle reduction filter is estimated, based on edge pixel statistics and noise variance. Then the optimized filter is used at the second stage to additionally smooth the approximation image of the wavelet sub-band. A complexity reduction algorithm for wavelet decomposition is also proposed. The obtained results are highly encouraging in terms of image quality which paves the way towards the reinforcement of the proposed algorithm for the performance enhancement of the Block Matching and 3D Filtering algorithm tackling multiplicative speckle noise.

Keywords: Denoising, Spatial Filter, Speckle, Wavelet.

1. INTRODUCTION

Multiplicative in nature, speckle noise is a common problem found in different imaging applications such as ultrasound, sonar and radar imaging [1, 2]. Originating from the superposition of acoustical echoes coming with random phases and amplitudes during acquisition or transmission, it tends to reduce the image resolution and contrast and blur important details [1].

Despeckling can address the multiplicative nature of the noise, or transform the noisy image to the logarithmic domain where multiplicative noise becomes additive, and apply additive noise reduction techniques [2].

In the past decades, several algorithms have been proposed for image denoising. Hybrid order statistics filters for speckle reduction are proposed in [2]. A preprocessing technique is used in order to transform the noise in the logarithmic domain to a Gaussian-like noise, which allows for better filtering results using known denoising techniques [3-5]. Lee [6-8], Kuan [3, 9], and Frost [10] filters are still widely used in many applications. In general, they succeed to reduce speckle in homogenous areas. However, in heterogeneous areas, speckle is retained. Therefore, they are not able to perform a full removal of speckle without blurring any edges because they rely on local statistical data related to the filtered pixel and this data depends on the occurrence of the filter window over an area. Wavelet-based denoising techniques [4, 10, 11, 12] represent the image jointly in the spatial and frequency domains. They rely on the sparse representation of the wavelet sub-bands coefficients in order to be distinctly thresholded. The Block Matching and 3D
Filtering (BM3D) algorithm [13, 14] is a non-local denoising technique in a non-spatial domain. It combines non-local image modeling and the sparse representation of the wavelet domain.

In the following, a two-stage despeckling algorithm is proposed. It consists of jointly denoising the speckled image in the wavelet and spatial domains. At the first stage, the traditional speckle reduction filter of Kuan [3] is adapted to the specificities of the filtered image by estimating its optimal parameter value automatically, based on edge pixels statistics and noise variance. Then the resulting sub-optimal spatial filter is used at the second stage to smooth the approximation sub-band coefficients in the wavelet domain. Since wavelet decomposition is time-consuming when dealing with large sized images, a complexity reduction algorithm is also proposed. Note that the Kuan filter is used as the basis for the spatial filtering due to its versatility. However, other algorithms could be used.

The remainder of this paper is organized as follows. Existing noise reduction methods are reviewed in Section 2. The proposed algorithm is then described in Section 3. In Section 4, results are shown and discussed. Finally, conclusions are drawn and prospects are provided in Section 5.

2. EXISTING NOISE REDUCTION ALGORITHMS

In the past decades, several image denoising algorithms have been proposed. For instance, the spatial means filters (arithmetic mean, geometric mean, harmonic mean and contraharmonic mean filters [15]) smooth the local variations in an image by blurring the noise. The spatial order-statistic filters (median, max, min, midpoint and alpha-trimmed mean filters [15]) are based on ordering (ranking) the values of the pixels contained in the image area encompassed by the filter. The spatial adaptive filters (adaptive local noise reduction and adaptive median filters [15]) change their smoothing behavior based on statistical characteristics of the image inside the filter region. The spatial filters listed above have shown to successfully deal with a large panel of noisy images in situations when only additive noise is present. However, they are not able to reduce speckle noise which is non-additively combined with the underlying image.

Various nonlinear filtering techniques have also been proposed [16]. They seek to reduce the effect of speckle noise while preserving the informative structure of the underlying image. Some of the best known standard speckle noise reduction filters are the methods of Lee [6-8], Gamma [24], Kuan [3, 9] and Frost [24]. These filters use the second-order sample statistics within a minimum mean squared error estimation approach.

2.1 Lee Filter

The locally adaptive Lee multiplicative filter [7, 8] is based on a multiplicative noise image model as follows:

\[ g(x, y) = f(x, y) \times n(x, y) \]  \tag{1}

where \( g \) denotes the observed image, \( f \) the original image and \( n \) the multiplicative speckle noise [6].

Based on an assumption that the noise is white, with unity mean, and uncorrelated with the image \( f \), the multiplicative Lee filter gives the best mean-squared estimate of \( f \) at each pixel \( g(x, y) \) by:

\[ z(x, y) = \hat{f} + k_{(x, y)} \left[ g(x, y) - \hat{f} \right] \]  \tag{2}

where \( z(x, y) \) is the filtered pixel, \( \hat{f} \) is the mean of the original image pixels under the filtering window (i.e. the local mean) and \( k_{(x, y)} \) is the gain factor given by:

\[ k_{(x, y)} = \frac{\sigma_n^2}{\sigma_f^2 + \sigma_n^2} \]  \tag{3}
where $\sigma_f^2$ is the variance of the original image pixels under the filtering window (local variance), and $\sigma_n^2$ is the noise variance. The local adaptation of the filter is based on the calculation of the local statistics $\bar{f}$ and $\sigma_f^2$ from the data sample estimates $\bar{g}$ and $\sigma_g^2$ determined over a local neighborhood window.

This requires the knowledge of the Coefficient of Variation (CoV), which is the ratio of the standard deviation to the mean in homogeneous areas. If the original image is not available, the noise variance can be determined from the number of looks or the Equivalent Number of Looks (ENL). This parameter effectively controls the amount of smoothing applied to the image. The filter size greatly affects the quality of the processed image. If the filter is too small, the noise filtering algorithm is not efficient. If the filter is too large, some details of the image will be lost [17-19].

An improved version of the basic Lee filter also exists; the modified Lee filter which relies on a ratio-based edge detection algorithm used to estimate the edge strength at each pixel in the image [20].

### 2.2 The Gamma Maximum A Posteriori Filter

The Gamma Maximum A Posteriori (MAP) filter [21] is used primarily to filter speckled radar data while preserving high frequency features. It's based on a Bayesian analysis of the image statistics. It performs spatial filtering on each individual pixel in an image using the grey level values in a square window surrounding each pixel. The Number of Looks (NLOOK) parameter effectively controls the amount of smoothing applied to the image. It affects the speckle coefficient of variation ($C_u$) and the upper speckle coefficient of variation ($C_{\text{max}}$). A small NLOOK value leads to more smoothing and a larger NLOOK value preserves more image features.

If $C_i \leq C_u$ ($C_i$ is the image variation coefficient), the filtering window is over a homogenous area and smoothing is applied. If $C_i \geq C_{\text{max}}$, the filtering window is over an area of high local standard deviation (crossing edge-pixels), and therefore edge-pixel is replicated [24].

If $C_u < C_i < C_{\text{max}}$, the filtered pixel value based on the Gamma estimation of the contrast ratios within the appropriate filter window is given by:

$$z(x,y) = \frac{(W - \text{NLOOK} - 1) \times \bar{g} + \sqrt{D}}{2 \times W},$$

where:

$$W = \frac{1 + C_u^2}{C_i^2 - C_u^2}, \quad D = g^2 \times (W - \text{NLOOK} - 1)^2 + 4 \times W \times \text{NLOOK} \times \bar{g} \times g(x,y)$$

and $\bar{g}$ is the mean of the window pixels.

### 2.3 Kuan Filter

Similarly to the Lee filter, Kuan filter [3, 9, 22] is designed to smooth out speckle noise while retaining shape features in the image by applying the Minimum Mean Square Error (MMSE) criterion and it is applied to the logarithmic transformation of the noisy image. Kuan filter is used primarily to filter speckled radar data. It performs spatial filtering on each individual
pixel in an image using the grey level values in a square window surrounding each pixel [24]. The resulting grey-level value for the smoothed pixel is given by:

\[ z(x, y) = g(x, y) \times W + \bar{g} \left(1 - W\right), \]  

where \( g(x,y) \) is the center pixel of the filtering window, \( \bar{g} \) is the mean value of intensity within the filtering window, and \( W \) is a weighting function which depends on the Number of Looks (NLOOK) parameter [23] and given by:

\[ W = \frac{1 - C_u^2 / C_i^2}{1 + C_u^2}, \]  

where \( C_u \) and \( C_i \) are the estimated noise variation coefficient and the image variation coefficient, respectively:

\[ C_u = \frac{1}{\sqrt{\text{NLOOK}}}, \quad C_i = \frac{\sigma_g}{g}, \]  

where \( \sigma_g \) is the standard deviation of intensity within the filtering window.

Same as the Lee filter case, a small NLOOK value corresponds to a high noise variation and a low \( W \) and therefore leads to more smoothing, while a larger NLOOK value corresponds to a low noise variation and a high \( W \) which preserves more image features.

Theoretically, the correct value for NLOOK should be the Equivalent Number of Looks (ENL) of the image. Depending on the application, the user may experimentally adjust the NLOOK value so as to control the effect of the filter [22].

2.4 Frost Filter

Frost method consists in adjusting the filter's parameters according to local area statistics about the target pixel [24]. When uniform regions are filtered, the filter acts as a mean filter and when high contrast regions are filtered, the filter acts as a high-pass filter with rapid decay of elements away from the filter center. Thus, large uniform areas will tend to be smoothed out and speckle removed, while high contrast edges and other objects will retain their values [24]. The frost filter can be considered as an adaptive-weighted-mean filter since it uses an adaptive filtering algorithm, which is an exponential damped convolution kernel that adapts itself to features by computing a set of weighting factors for each pixel within the filtering window as follows:

\[ M_n = e^{-\left(DAMP \times \left(\frac{\sigma_g}{g}\right)^2\right) \times T}, \]  

where \( DAMP \) is a factor that determines the extent of the exponential damping for the image, \( \sigma_g \) is the standard deviation of the filter window, \( \bar{g} \) is the mean value within the window and \( T \) is the absolute value of the grey level distance between the center pixel and its surrounding pixels in the filter window [24].

The Enhanced Frost filter is an extension to the basic filter that further divides the image into homogeneous, heterogeneous and isolated point target areas [19]. It can significantly improve the ability of speckle mitigation in the vicinity of edges and small features, thus retaining more image details.

2.5 Denoising Using Wavelets

Wavelet transform [25] has been extensively studied in recent years and used for many application domains, mainly in image compression and noise reduction. It consists of a set basis functions used to analyze signals in both spatial and frequency domains simultaneously. The
basic functions of the wavelet transform help to isolate signal discontinuities since high pass filtering is used to obtain detail information and low pass filtering is used to retrieve a smoother approximation of the signal, making it possible to analyze the signal at different scales [22].

**Multi-resolution processing of the Discrete Wavelet Transform (DWT)**

Multi-resolution processing consists in constructing a set of child wavelets from a mother wavelet using scaling and wavelet functions [22]. These two functions form a filter bank consisting of a low pass and high pass filters. The idea of multi-resolution processing through wavelet decomposition is to pass the signal through the filter bank; the signal is decomposed to detail coefficients (output of the high pass filter) and approximation coefficients (output of the low pass filter). Then, the filter outputs are down-sampled by 2 in the purpose of discarding half the samples. The decomposition is repeated to further increase the frequency resolution [25-27].

**Application to Image Denoising**

Considering an image corrupted by an additive noise and modeled as \( g = f + n \), where \( f \) denotes the unknown, noise-free image and \( n \) the noise [28, 29], wavelet-based denoising (Figure 1) consists in:

1) Applying the DWT to \( g \).
2) Thresholding the detail coefficients (Wavelet shrinkage).
3) Inverse transforming (IDWT) the result to obtain an estimation \( z \) of the original image \( f \).

![FIGURE 1: DWT Denoising Block Diagram.](image)

When dealing with speckled images, a logarithmic transformation is applied to the noisy image before wavelet decomposition, to transform the multiplicative noise, into additive noise and after wavelet reconstruction, an exponential transformation is applied to reverse the logarithmic operation.

In wavelet domain, the output of the low pass filter consists of the high magnitude and low frequency components (approximation coefficients) and the output of the high pass filter consists of the low magnitude and high frequency components (detail coefficients). Figure 2 shows a one level, 2D wavelet decomposition scheme [30] where \( L(.) \) represents the low pass filtering operator, and the subscript \((.)\) represents a low pass filtered output. Similarly, \( H(.) \) and the subscript \((.)\) represent high pass filtering and a high-pass filtered output, respectively. The symbol \((\downarrow 2)\) represents a down-sampling operator by a factor of 2.

As a result of filtering and down sampling (Figure 2), four sub-bands are obtained: scaling (approximation) coefficients \((g_{LL})\), horizontal detail coefficients \((g_{HL})\), vertical detail coefficients \((g_{LH})\) and diagonal detail coefficients \((g_{HH})\). The \((HL)\), \((LH)\) and \((HH)\) sub-bands, represent the high frequency (and low magnitude) components and the \((LL)\) sub-band represents the low frequency (and high magnitude) components. At the next level of decomposition, only the \((LL)\) component is passed to the decomposition process.

The reconstruction process or the Inverse Discrete Wavelet Transform (IDWT) consists of assembling back the wavelet coefficients to the original image. After each inverse low and high pass filtering, an up-sampling process (zeros insertion) is required to reverse the decomposition process.
The main idea of wavelet denoising is to threshold only the high frequency components while preserving most of the features in the image by retaining the approximation sub-band.

There are two thresholding methods frequently used; hard [26, 31] and soft [32] thresholding. In hard thresholding, the input is kept if its amplitude is greater than a threshold \( T \), otherwise it is forced to zero. In soft thresholding, if the absolute value of the input is less than or equal to a threshold \( T \), then the output is forced to zero, otherwise, the output is a scaled version of the input.

\[ \text{FIGURE 2: One level, 2D Wavelet Decomposition Block Diagram.} \]

Finding a suitable threshold is an important task in wavelet shrinkage. In fact, choosing a very large threshold will shrink to zero almost all coefficients, which results in over smoothing the image, while choosing a very small threshold will yield a noisy result which is close to the input [29]. VisuShrink [25] is an approach that uses a Universal (global) threshold applied to all sub-bands and scales after decomposition [33]. However, this threshold can be unwarrantedly large because it depends on the number of pixels yielding overly smoothed images. In addition, it ignores the difference between sub-bands at different scales [34]. BayesShrink [25, 31] is a sub-band adaptive threshold selection technique that determines a specific threshold for each sub-band assuming a Generalized Gaussian Distribution (GGD) [35]. This method depends on the standard deviation of the noise-free image, and the GGD shape parameter [34]. The method consists of finding, for each sub-band, a threshold that minimizes the expected value of the mean square error (Bayesian Risk) [32, 36].

\[ \text{FIGURE 2: One level, 2D Wavelet Decomposition Block Diagram.} \]

2.6 The Block Matching and 3D Filtering (BM3D) Algorithm

The BM3D algorithm is a non-local denoising technique [13, 14] in a non-spatial domain. It combines non-local modeling and the sparse representation of the wavelet domain. The algorithm is divided into three processing stages: Block matching, 3D filtering, and aggregation.

In the block matching stage, the noisy image \( g \) is divided into blocks \( G_{x \in X} \) where \( x \) represents the position of each block in the whole image \( X \). For each block, the patches with most resemblance are grouped in a 3D array. Therefore, groups whose elements have a high degree of similarity are constructed separately. Two patches are similar if the Euclidean distance between them is less than a given threshold [14, 37]. The block matching advantages are the induction of a high correlation in the third dimension of the 3D array and the improvement of the dispersion of all possible configurations of details present in the image [13].
3D filtering is a procedure that jointly filters a group of similar blocks by exploiting the similarities between the grouped images and inside each image in the group. It consists of three different sections: 3D transform, shrinkage and inverse 3D transform. Those three procedures are jointly applied on the block. The 3D transform consists of applying a 2D wavelet transform on each patch of the block then a 1D wavelet or Discrete Cosine Transform (DCT) on the resulting patches. As a result of the sparse representation given by the 3D transform, the shrinkage process can effectively attenuate the noise by eliminating the coefficients relying under a certain chosen threshold. The inverse 3D transform consists of assembling back the thresholded coefficients to reconstruct the 3D block.

Finally, the aggregation stage consists of combining the patches in the 3D filtered groups including the reference patch. A trivial solution is to compute the weighted mean of all the estimated patches overlapping at a pixel position [13].

3. THE PROPOSED WAVELET/SPATIAL DESPECKLING ALGORITHM

In this section, a two-stage despeckling algorithm which consists in jointly denoising the corrupted image in the wavelet and spatial domains is proposed. At the first stage, an automatic estimation of the optimal Kuan filter parameter value based on edge pixels statistics and noise variance is developed. Then the resulting adaptive filter is used at the second stage to spatially smooth the approximation sub-band coefficients in the wavelet domain. Since a large number of wavelet levels makes the despeckling algorithm computationally expensive, a complexity reduction algorithm for wavelet decomposition is also proposed. Note that the proposed enhancement targets the Kuan filter due to its versatility and adaptability to speckle noise strength, while a similar study could have been performed with another spatial despeckling method to be used in the hybrid filter that will be discussed in Section 3.2; we therefore focus on the Kuan filter without loss of generality.

3.1 The Adaptive Spatial Filter

As explained earlier, Kuan filtering relies on the Number of Looks (NLOOK) parameter which significantly affects the filtering performance and is usually taken equal to the Equivalent Number of Looks (ENL) of the image. This parameter is manually adjusted, in order to control the strength of the smoothing applied to the image.

We start by implementing the additive model of the Kuan filter using a set of test images (e.g. Lena, Mandrill, LivingRoom, … [38]) corrupted by speckle noise with different distributions. By setting NLOOK = ENL, we notice that filtered images are over-blurred, that’s why came our idea of analyzing the Kuan filter performance with respect to the NLOOK parameter and proposing a novel technique for selecting a suitable NLOOK value that yields near-optimal performance compared to manual parameter selection, therefore eliminating the need for several runs of the filter to reach the optimal performance. In this purpose, we define the adjusted NLOOK value as:

$$NLK\_AD = (1 + A)ENL$$  \hspace{1cm} (9)

where $A$ is the adjustment coefficient.

*Figure 3* shows an example of the Lena image filtered with NLOOK = ENL (middle) and NLK AD as defined in (9), for $A = 0.65$ (right). It can be clearly seen that the image with the adjusted NLOOK parameter (NLK_AD) is much sharper and its details are better preserved.

In the first part of this study, we aim at finding a closed form expression for the adjustment coefficient that yields near optimal performance, without the need for several runs of the filter with different NLOOK values. Therefore, we consider the effect of the filter window size, the image detail, the noise distribution and the noise variance on the optimal adjustment coefficient $A$ and consequently, the NLK_AD parameter. Thus, all the listed parameters are fixed and the effect of varying each one of them is studied.
Effect of the Algorithm Parameters on the Optimal Adjustment Coefficient

A Gaussian speckle noise having a variance of 2600 is added to the Cameraman image [38] in order to find the adjustment coefficient $A$ that yields the best result using different window size. The Peak Signal-to-Noise Ratio (PSNR) and the Signal to Mean Squared Error (S/MSE) variations with respect to the NLK_AD parameter are plot and ensure that, for different odd-sized filter windows, the optimal values of $A$ are nearly constant.

On the other hand, test images are distorted with multiplicative speckle noise using the Uniform, Gaussian, Rayleigh, and Erlang distributions while fixing all the other parameters and the optimal value of $A$ remains the same regardless of the noise distribution, for each test image.

In the purpose of studying the influence of the image itself on the filtering process, the filter is applied on different images, having different features, from [38] and [39] with all the other parameters being fixed. It can be noticed that the optimal adjustment coefficient varies with image details. For example, the Pirate image [38] has a high percentage of edges, therefore, the obtained optimal adjustment coefficient is greater than with the Lena image, which is expected since more image features have to be preserved.

To find the relationship between the optimal adjustment coefficient $A$ and the amount of image detail, edge detection is performed on the set of images and the curves presenting the optimal $A$ with respect to the noise variance and the number of edge pixels in each image are drawn.

The Logarithmic and Polynomial Approximations

Since the noise variance affects the filtering process, the variation of $A$ with the noise variance and image detail is modeled as:

$$A = A_d + A_v,$$

where $A_d$ and $A_v$ are the detail and the noise variance components in $A$, respectively.
We observe that a higher noise variance corresponds to a lower value of $A$, which is expected since a lower $A$ value leads to more smoothing. Additionally, a higher noise variance corresponds to higher edge percentage ($P_e$) values because the edge detection algorithm cannot completely differentiate between noise and small edges. Curves for different noise variances have similar behavior with respect to the percentage of edges in the image, therefore, we take as reference one of the curves (for a given variance).

The curve with the solid dark line in Figure 4 shows the measured $A_d$ with respect to $P_e$. It can be seen that, for $P_e$ exceeding 2%, $A_d$ varies in a logarithmic or second order polynomial fashion. The former can be expressed as:

$$A_d = \alpha \ln(P_e) + \beta,$$

and the latter as:

$$A_d = \lambda P_e^2 + \delta P_e + \mu,$$

where $\alpha = 0.4015$, $\beta = 0.054$, $\lambda = -0.01215$, $\delta = 0.20512$ and $\mu = -0.01394$ determined by fitting the measured results with (11) and (12) using the Least Squares method.

It can be observed in Figure 4 that, as the image detail ($P_e$) increases, $A_d$ increases and consequently, NLK_AD increases. Therefore, less smoothing is performed, as expected. The logarithmic and polynomial approximations mainly overlap with measured data, except for $P_e < 2\%$, which proves high accuracy of the models in (11) and (12).

A similar study with respect to the noise variance shows that $A_v$ (Figure 5) can be accurately modeled by a second order polynomial as:

$$A_v = a(v - 2100)^2 + b(v - 2100),$$

where $a = -2x10^{-8}$ and $b = -42x10^{-8}$ determined by curve fitting.

Note that for $v = 2100$, $A_v = 0$. In fact, $A_v$ reflects the variation of the adjustment coefficient with respect to the noise variance compared to the reference $v = 2100$ which has been used to derive $A_d$ in (11) and (12). On the other hand, $A_v$ decreases as $v$ increases, which is also expected since more smoothing is required (lower value of NLK_AD) when the noise increases.
3.2 The Hybrid Wavelet-Spatial Filter

In wavelet denoising, only detail coefficients are thresholded and approximation coefficients remain stagnant, as explained earlier. This is intuitive since the approximation component is a low-frequency image, usually assumed to be noise-free. In fact, this is only a theoretical aspect since practically, approximation coefficients also contain speckle noise as it can be clearly observed in the example shown in Figure 6. Wherefore, the proposed algorithm consists of smoothing the approximation coefficients in addition to detail coefficients thresholding as follows:

1) Applying the Discrete Wavelet Transform to the speckled image.
2) Thresholding the detail coefficients (Wavelet shrinkage).
3) Spatial Smoothing of the approximation sub-band coefficients.
4) Applying the Inverse Discrete Wavelet Transform to the result.

FIGURE 6: Cameraman image of size 512×512 (up-left) [38], corrupted by speckle noise of unit mean and a variance of 2400 (up-right), decomposed structure (down left) and the denoised image (down right). Note the existence of noise in the approximation image of the decomposed structure and the remaining noise in the denoised image (using sym1 wavelets with BayesShrink threshold selection).
An important issue in the approximation image spatial smoothing would be the preservation of the image characteristics. In other words, the algorithm must efficiently smooth out the noise in the approximation image without blurring its details or reducing the contrast and quality. Otherwise, the image structure and dynamics would be lost. Therefore, the approximation sub-band coefficients in the wavelet domain are smoothed using the previously developed adaptive Kuan filter which leads to a better speckle removal without image distortion, as will be seen in Section 4.

3.3 The Proposed Wavelet Acceleration Algorithm

This section deals with accelerating the wavelet filtering process used in the proposed method. The acceleration algorithm consists in finding the number of decomposition levels that yields near optimal performance, without the need for several runs of the wavelet filter with different number of decomposition levels, regardless of the image size, its edge percentage, the noise variance, the wavelet type and base, the wavelet threshold selection technique and the wavelet thresholding technique.

Relative Difference Threshold Selection

We start by applying 100 times, wavelet denoising on an image corrupted by Gaussian speckle noise using different number of decomposition levels (from 1 level to the maximum number of decomposition levels calculated as in [40]). For every run, the simulated speckle noise is regenerated and the PSNR, the Coefficient of Correlation (CoC) and the ENL metrics of the denoised image are calculated and saved.

The second step is the calculation of the relative difference between the ENL of each two consecutive decomposition levels, for the purpose of finding a reasonable relative difference threshold that allows the algorithm to stop before reaching the maximum number of decomposition levels, without loss in performance. The same is performed for different images from [38] and [39], different noise variances (2500, 2600, 3000, 3100, 3500, 4100, 4600 and 6000), different image sizes (128×128, 256×256, 512×512 and 1024×1024), different wavelet types and bases (db2, db4, db7, sym1, sym2 and sym6) and using bayesShrink threshold selection.

After analyzing the results, we noticed that a relative difference smaller than 0.2 leads to an approximately steady PSNR value. As a result, 0.2 is taken as a threshold. Therefore, our proposed algorithm consists of applying a level of wavelet decomposition and denoising, computing the relative difference between the corresponding ENL and the ENL of the previous decomposition level, and stop when the relative difference becomes less than 0.2 as shown in Figure 7.

By limiting the number of decomposition levels, a computational gain is expected at the expense of some loss in image quality, as will be shown in Section 4.

![FIGURE 7: Flowchart of the Proposed Acceleration Algorithm.](image-url)
4. PRACTICAL RESULTS

In this section, the proposed algorithms are put in practice. Besides the subjective qualitative evaluation based on human perception, a quantitative analysis is essential. Thus, an objective benchmark is used to study the quality-related outcomes of the filtering process. Namely, the Peak Signal-to-Noise Ratio (PSNR), the Signal-to-Mean-Squared-Error (S/MSE), the Coefficient of Correlation (CoC) and the Equivalent Number of Looks (ENL).

**Peak Signal-to-Noise Ratio (PSNR)**

PSNR is considered to be the least complex metric, as it defines the image quality degradation as a plain pixel by pixel error power estimate. PSNR is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation [17]. In image processing, the signal is the original image, and the noise is the error (blur) introduced by the denoising procedure.

**Signal-to-Mean Squared Error (S/MSE)**

In order to quantify the achieved performance improvement, the Signal-to-Mean Squared Error can also be computed, based on the original and the noisy/denoised images as follows:

\[
S/MSE = 10 \log_{10} \left( \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i,j))^2}{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (K(i,j) - I(i,j))^2} \right),
\]

where \(I\) is the original image and \(K\) the noisy or denoised image. This measure corresponds to the classical SNR in the case of additive noise [41].

**Coefficient of Correlation (CoC)**

In ultrasound imaging, it is important to suppress speckle noise while at the same time preserving the edges of the original image that often constitute features of interest for diagnosis. For this reason, we also considered a qualitative measure for edge preservation, the Coefficient of Correlation (CoC) metric:

\[
CoC = \frac{\left(\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i,j) - \overline{I})(K(i,j) - \overline{K})\right)}{\left(\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i,j) - \overline{I})^2\right)^{\frac{1}{2}} \left(\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (K(i,j) - \overline{K})^2\right)^{\frac{1}{2}}},
\]

where \(\Delta_I\) and \(\Delta_K\) are the high-pass filtered versions of \(I\) and \(K\) respectively, obtained with a 3x3 pixel standard approximation of the Laplacian operator [25]. \(\overline{I}\) is the mean value of \(\Delta_I\) and \(\overline{K}\) is the mean value of \(\Delta_K\) and the operator \(\overline{\cdot}\) is given by:

\[
\overline{(I,K)} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} I(i,j)K(i,j),
\]

where \(m\times n\) is the size of the corresponding images \(I\) and \(K\). The CoC cannot exceed 1 in absolute value. It is 1 in the case of an increasing linear relationship and -1 in the case of a decreasing linear relationship. Its value lies in between in all other cases, indicating the degree of linear dependence between the images. The absolute value of the correlation measure should be close to unity to an optimal effect of edge preservation [41].

**Equivalent Numbers of Looks (ENL)**

The Equivalent Number of Looks is also a good approach for either estimating the speckle noise variance in a noisy image (in general a SAR image), or evaluating the performance obtained after filtering. ENL is often calculated over a uniform region, but due to difficulties in identifying uniform areas, the image is divided into small areas of 25 x 25 pixels, the ENL is computed for each area.
(17), then the average of these ENL values is taken as explained in [12]. Note that this latter method is used throughout this paper. A large ENL value usually corresponds to a better performance. The formula for the ENL calculation is given by:

$$ENL = \left(\frac{\text{mean}}{\text{standard deviation}}\right)^2.$$  \hspace{1cm} (17)

Note that PSNR, S/MSE, and CoC are quantitative metrics, they are based on a pixel by pixel calculation and the original image is needed for computation, whereas ENL is based on the mean and standard deviation, and it can be computed without the need of the original image [22].

4.1 The Performance of the Proposed Adaptive Spatial Filter

It is trivial that a good performing spatial filtering is necessary to get a denoised image of good quality using the proposed approach. Otherwise, the image characteristics will be lost. In this section, the NLOOK parameter estimation previously developed is exploited and compared to the results obtained with the optimal NLOOK value determined manually. Simulations were performed using different standard images from [38] and [39], corrupted by speckle noise of variance ranging from 2100 to 4100. Since many of the samples lead to an approximately same percentage ($Pe$) of edge pixels, and therefore to a similar algorithm behavior, table 1 summarizes sample results obtained for 8 different images having different $Pe$ values.

It can be observed that, except for $Pe < 2$ (e.g. Lady image), both the logarithmic and polynomial estimations result in near optimal values for the adjustment parameter, and the PSNR loss does not exceed 0.33 dB and 0.16 dB with logarithmic and polynomial estimations, respectively. In fact, the behavior of the filter when dealing with few-detailed images (Lady image in table 1) is expected, since both proposed approximations fail to be accurate in cases of very low $Pe$ values as shown in Figure 4. However, images with a percentage of edge pixels less than 2 rarely exist (in our test datasets from [38] and [39], only 11% of the images had $Pe < 2$). Furthermore, the approximation error can be largely reduced by setting $A_d$ to a constant value (approximately 0.43) for such low values of $Pe$ as implies Figure 4.

It is important to mention that experimental results with such high noise variances are shown to highlight the decent behavior of the proposed method in critical conditions. Moreover, when dealing with slightly speckled images, no filtering is performed ($W \approx 1$ in (5)), as will be verified in section 4.2.

Figure 8-a shows the Cumulative Density Function (CDF) of the PSNR loss obtained with both estimations, using the whole set of test images. It can be noticed that in both methods, 50% of the filtered images undergo a loss in PSNR not exceeding 0.05 dB, whereas 85% and 89% undergo a PSNR loss that does not exceed 0.11 dB in logarithmic and polynomial approximations, respectively.

Even though values of $Pe$ are more likely to occur between 2% and 8%, let us analyze the proposed models’ behavior for values of $Pe$ outside this interval. It can be deduced from Figure 4 that for low values of $Pe$, the measured $A_d$ seems to saturate at a value close to 0.43, while the logarithmic approximation reaches 0 for values of $Pe$ close to 0.9% as shown in Figure 8-b, and becomes negative for lower values.

Similarly, the polynomial approximation also fails to model $A_d$ at low values of $Pe$, but results in slightly better performance compared to the logarithmic model. This problem can be solved by using any of the two models only when $Pe$ exceeds a fixed threshold (e.g. for $Pe \geq 2$%), or by setting $A_d$ to a constant value (e.g. $A_d = 0.43$) when $Pe$ is below that threshold. On the other hand, the polynomial approximation reaches a peak for $Pe \approx 8.5\%$, and then decreases as $Pe$ increases. Therefore, the polynomial model becomes inaccurate in this case. The same does not occur with the logarithmic approximation since it keeps increasing as $Pe$ increases.
<table>
<thead>
<tr>
<th>Noise variance</th>
<th>Image</th>
<th>$P_e$ (%)</th>
<th>Speckled Image PSNR</th>
<th>Log approximation</th>
<th>Polynomial approximation</th>
<th>Optimal values</th>
</tr>
</thead>
<tbody>
<tr>
<td>2100</td>
<td>Lady</td>
<td>0.9719</td>
<td>26.1012</td>
<td>4.2606</td>
<td>12.1041</td>
<td>24.276</td>
</tr>
<tr>
<td></td>
<td>WomanDarkHair</td>
<td>2.6634</td>
<td>20.6058</td>
<td>44.7361</td>
<td>30.1964</td>
<td>29.115</td>
</tr>
<tr>
<td></td>
<td>I06</td>
<td>3.1488</td>
<td>20.5841</td>
<td>51.2811</td>
<td>28.3623</td>
<td>27.569</td>
</tr>
<tr>
<td></td>
<td>Lena</td>
<td>4.4518</td>
<td>20.7322</td>
<td>65.3613</td>
<td>28.6966</td>
<td>27.766</td>
</tr>
<tr>
<td></td>
<td>Board</td>
<td>6.0764</td>
<td>19.6519</td>
<td>77.8522</td>
<td>26.12</td>
<td>24.224</td>
</tr>
<tr>
<td></td>
<td>LivingRoom</td>
<td>7.0168</td>
<td>20.9285</td>
<td>83</td>
<td>28.1954</td>
<td>25.992</td>
</tr>
<tr>
<td></td>
<td>Mandrill</td>
<td>11.597</td>
<td>20.5485</td>
<td>103.801</td>
<td>30.417</td>
<td>24.865</td>
</tr>
<tr>
<td>2600</td>
<td>Lady</td>
<td>0.9887</td>
<td>25.2349</td>
<td>2.3487</td>
<td>10.611</td>
<td>24.451</td>
</tr>
<tr>
<td></td>
<td>WomanDarkHair</td>
<td>3.0743</td>
<td>20.8759</td>
<td>47.8966</td>
<td>26.4071</td>
<td>28.423</td>
</tr>
<tr>
<td></td>
<td>I06</td>
<td>3.3284</td>
<td>19.6843</td>
<td>52.0432</td>
<td>23.9424</td>
<td>27.107</td>
</tr>
<tr>
<td></td>
<td>Lena</td>
<td>4.5052</td>
<td>19.855</td>
<td>63.216</td>
<td>24.2587</td>
<td>27.195</td>
</tr>
<tr>
<td></td>
<td>Board</td>
<td>6.0455</td>
<td>18.7322</td>
<td>73</td>
<td>22.0966</td>
<td>23.658</td>
</tr>
<tr>
<td></td>
<td>I13</td>
<td>6.2467</td>
<td>20.5233</td>
<td>74.9848</td>
<td>22.1354</td>
<td>26.260</td>
</tr>
<tr>
<td></td>
<td>LivingRoom</td>
<td>6.5968</td>
<td>20.041</td>
<td>78.5514</td>
<td>23.8295</td>
<td>25.51</td>
</tr>
<tr>
<td></td>
<td>Mandrill</td>
<td>11.382</td>
<td>19.6209</td>
<td>100.451</td>
<td>26.1791</td>
<td>24.308</td>
</tr>
<tr>
<td>3100</td>
<td>Lady</td>
<td>0.9901</td>
<td>24.5231</td>
<td>-1.1945</td>
<td>9.3747</td>
<td>24.321</td>
</tr>
<tr>
<td></td>
<td>WomanDarkHair</td>
<td>3.0277</td>
<td>20.177</td>
<td>43.674</td>
<td>23.3196</td>
<td>27.926</td>
</tr>
<tr>
<td></td>
<td>Lena</td>
<td>4.4182</td>
<td>19.1324</td>
<td>58.8572</td>
<td>20.8946</td>
<td>26.676</td>
</tr>
<tr>
<td></td>
<td>Board</td>
<td>6.339</td>
<td>18.0201</td>
<td>73.3508</td>
<td>19.8109</td>
<td>23.235</td>
</tr>
<tr>
<td></td>
<td>I13</td>
<td>6.356</td>
<td>19.8108</td>
<td>75.473</td>
<td>19.4415</td>
<td>25.622</td>
</tr>
<tr>
<td></td>
<td>LivingRoom</td>
<td>7.2327</td>
<td>19.2983</td>
<td>78.6463</td>
<td>21.3184</td>
<td>25.078</td>
</tr>
<tr>
<td></td>
<td>Mandrill</td>
<td>12.095</td>
<td>18.9053</td>
<td>99.2916</td>
<td>23.3454</td>
<td>23.803</td>
</tr>
<tr>
<td>3600</td>
<td>Lady</td>
<td>0.9953</td>
<td>23.9122</td>
<td>-5.5841</td>
<td>8.2428</td>
<td>24.029</td>
</tr>
<tr>
<td></td>
<td>WomanDarkHair</td>
<td>3.1719</td>
<td>19.5572</td>
<td>40.9514</td>
<td>19.5081</td>
<td>27.608</td>
</tr>
<tr>
<td></td>
<td>I06</td>
<td>3.6442</td>
<td>18.3526</td>
<td>53.8066</td>
<td>17.2142</td>
<td>26.201</td>
</tr>
<tr>
<td></td>
<td>Lena</td>
<td>4.5715</td>
<td>18.5216</td>
<td>55.6266</td>
<td>18.2715</td>
<td>26.398</td>
</tr>
<tr>
<td></td>
<td>Board</td>
<td>6.3700</td>
<td>17.4414</td>
<td>68.9511</td>
<td>17.4682</td>
<td>22.829</td>
</tr>
<tr>
<td></td>
<td>I13</td>
<td>6.4014</td>
<td>19.0212</td>
<td>73.3316</td>
<td>17.9968</td>
<td>25.596</td>
</tr>
<tr>
<td></td>
<td>LivingRoom</td>
<td>7.1453</td>
<td>18.7005</td>
<td>73.5582</td>
<td>18.6148</td>
<td>24.753</td>
</tr>
<tr>
<td></td>
<td>Mandrill</td>
<td>11.821</td>
<td>18.2575</td>
<td>93.7692</td>
<td>20.7329</td>
<td>23.4</td>
</tr>
<tr>
<td>4100</td>
<td>Lady</td>
<td>0.9971</td>
<td>23.4004</td>
<td>11.1116</td>
<td>7.1373</td>
<td>23.456</td>
</tr>
<tr>
<td></td>
<td>WomanDarkHair</td>
<td>3.2227</td>
<td>19.0674</td>
<td>35.9893</td>
<td>16.9639</td>
<td>27.374</td>
</tr>
<tr>
<td></td>
<td>I06</td>
<td>3.6188</td>
<td>17.7216</td>
<td>38.2526</td>
<td>14.7722</td>
<td>25.921</td>
</tr>
<tr>
<td></td>
<td>Lena</td>
<td>4.7283</td>
<td>17.9916</td>
<td>51.3807</td>
<td>16.1424</td>
<td>26.111</td>
</tr>
<tr>
<td></td>
<td>Board</td>
<td>6.3188</td>
<td>16.8767</td>
<td>63.0227</td>
<td>15.4386</td>
<td>22.472</td>
</tr>
<tr>
<td></td>
<td>I13</td>
<td>6.3904</td>
<td>18.7049</td>
<td>64.2214</td>
<td>15.9697</td>
<td>25.455</td>
</tr>
<tr>
<td></td>
<td>LivingRoom</td>
<td>7.9094</td>
<td>18.1322</td>
<td>72.0373</td>
<td>16.8162</td>
<td>24.427</td>
</tr>
<tr>
<td></td>
<td>Mandrill</td>
<td>12.867</td>
<td>17.733</td>
<td>91.5761</td>
<td>18.8708</td>
<td>23.045</td>
</tr>
</tbody>
</table>

**TABLE 1:** Summary of results obtained with different 512×512 speckled images using the proposed adaptive spatial filter.
Figure 8: (a) CDF of the loss in PSNR due to logarithmic (dotted line) and polynomial (solid line) approximations. (b) Logarithmic (dashed line) and polynomial (solid line) approximations as defined in (11) and (12).

Table 2, 3, 4 and 5 show a comparison of the Lee filter, standard (NLOOK=ENL) Kuan filter, Gamma filter, Frost filter, Enhanced Frost filter, and the proposed enhanced Kuan filter using both the logarithmic and polynomial approximations with a speckle noise of Gaussian, Uniform, Rayleigh and Exponential distributions respectively.

The same is performed for different images of different edge percentage and the results were approximately similar. It can be clearly seen that the performance of the filters is affected by the noise distribution. For the examples shown in Tables 2-5, the Lee multiplicative filter performs better than the Enhanced Frost filter with Uniformly distributed noise, and worse with Gaussian noise, the Kuan filter gives better results than the Lee multiplicative filter except for the case with NLOOK=ENL (where this latter parameter is not adjusted). Depending on the noise distribution, either of the approximations (logarithmic or polynomial) can outperform the other in Kuan filtering with adjusted NLOOK parameter.

In tables 2-5, it can also be observed that the best PSNR is obtained with either the Gamma MAP filter or the proposed (modified) Kuan filter, whereas the Frost and enhanced Frost filters preserve image features more than other filters as it can be noticed from CoC values.

<table>
<thead>
<tr>
<th>Noise Distribution</th>
<th>PSNR</th>
<th>S/MSE</th>
<th>CoC</th>
<th>ENL</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Gaussian noise</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Noisy image</td>
<td>19.2482</td>
<td>13.5918</td>
<td>0.1781</td>
<td>13.3615</td>
</tr>
<tr>
<td>Lee multiplicative filtered image</td>
<td>26.9749</td>
<td>21.3185</td>
<td>0.3028</td>
<td>98.0283</td>
</tr>
<tr>
<td>Kuan filtered image (NLOOK=ENL)</td>
<td>22.8675</td>
<td>17.2111</td>
<td>0.1155</td>
<td>31.6124</td>
</tr>
<tr>
<td>K.filtered image (Log. approximation)</td>
<td>27.2248</td>
<td>21.5685</td>
<td>0.3106</td>
<td>106.6198</td>
</tr>
<tr>
<td>K.filtered image (Poly. approximation)</td>
<td>27.2389</td>
<td>21.5826</td>
<td>0.3154</td>
<td>110.6581</td>
</tr>
<tr>
<td>Gamma filtered image</td>
<td>28.1836</td>
<td>22.5272</td>
<td>0.3892</td>
<td>233.4135</td>
</tr>
<tr>
<td>Frost filtered image</td>
<td>26.3968</td>
<td>20.7404</td>
<td>0.4225</td>
<td>233.3297</td>
</tr>
<tr>
<td>Enhanced Frost filtered image</td>
<td>27.0823</td>
<td>21.4259</td>
<td>0.4394</td>
<td>240.8940</td>
</tr>
</tbody>
</table>

Table 2: Comparison of speckle noise filtering techniques on a 512×512 Lena image corrupted by a Gaussian speckle noise using an 11×11 window size.
Table 3 shows a comparison between the proposed adaptive Kuan filter and the basic and enhanced versions of Lee, Kuan, and Frost filters using five different images. It can be noticed that the optimal Kuan filter yields the highest PSNR, with the drawback of performing several filter runs in order to tune the filter to yield the best possible output quality. Furthermore, by setting the $\text{NLOOK}$ parameter to ENL, Lee and Frost filters outperform the Kuan filter in their basic and enhanced versions. Moreover, our proposed solution approaches the performance of the optimal Kuan filter, with a negligible PSNR loss, and outperforms the basic and enhanced Lee and Frost filters, for all the speckled test images used in the filtering process.
<table>
<thead>
<tr>
<th></th>
<th>WomanD.H.</th>
<th>Pirate</th>
<th>Lena</th>
<th>CameraMan</th>
<th>LivingRoom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy image</td>
<td>19.55</td>
<td>20.02</td>
<td>19.42</td>
<td>20.1</td>
<td>19.9</td>
</tr>
<tr>
<td>Lee</td>
<td>25.39</td>
<td>25.91</td>
<td>27.14</td>
<td>26.01</td>
<td>26.2</td>
</tr>
<tr>
<td>Frost</td>
<td>23.69</td>
<td>25.08</td>
<td>26.38</td>
<td>25.9</td>
<td>25.89</td>
</tr>
<tr>
<td>Enhanced Lee</td>
<td>25.61</td>
<td>26.01</td>
<td>27.30</td>
<td>26.78</td>
<td>26.88</td>
</tr>
<tr>
<td>Enhanced Frost</td>
<td>25.47</td>
<td>25.60</td>
<td>27.08</td>
<td>26.12</td>
<td>26.18</td>
</tr>
<tr>
<td>Kuan</td>
<td>23.34</td>
<td>24.21</td>
<td>24.02</td>
<td>24.92</td>
<td>25.1</td>
</tr>
<tr>
<td>Kuan (poly. approximation)</td>
<td>25.78</td>
<td>26.57</td>
<td>27.35</td>
<td>26.95</td>
<td>27.21</td>
</tr>
<tr>
<td>Kuan (log. approximation)</td>
<td>25.88</td>
<td>26.59</td>
<td>27.33</td>
<td>26.89</td>
<td>27.1</td>
</tr>
<tr>
<td>Kuan (optimal NLOOK)</td>
<td>25.92</td>
<td>26.70</td>
<td>27.44</td>
<td>27.11</td>
<td>27.35</td>
</tr>
</tbody>
</table>

**TABLE 6:** PSNR obtained with different despeckling filters using images of 512×512 size and an 11×11 filtering window.

It is important to note that in the literature [13, 14, 37], the Block Matching and 3D filtering (BM3D) algorithm is tested on images corrupted by additive noise. Moreover, the method's different parameters should be adequately chosen (depending on the noise strength, image size and percentage of edge pixels etc) in order to perform well dealing with speckle noise. Those parameters are the reference block size, the searching window size, the number of similar patches in the 3D block and the shrinking threshold. Therefore, finding the optimal set of parameters dealing with speckle noise is a very difficult task and choosing a non-optimal set of parameters usually fails to outperform traditional despeckling filters. On the contrary, the proposed adaptive filter can automatically estimate the suitable ENL parameter which will be used to spatially smooth the approximation image in the wavelet domain without need for many filter runs with different parameters.

*Figure 9* shows a real spine lumbar MRI image obtained from [42], filtered with Enhanced Frost, Enhanced Lee, and the proposed adaptive Kuan filter using the polynomial approximation. A visual inspection of the results shows that all the filters succeed to reduce speckle noise and result in a pleasant visual appearance, preserving edges and contours. However, compared to our proposed solution, the enhanced Lee and Frost filters show over-blurred images.

*FIGURE 9:* Spine lumbar MRI image [42]. Top left: original (noisy) image. Top right: Enhanced Frost filtered image. Bottom left: Enhanced Lee filtered image. Bottom right: result of filtering with our proposed solution using the polynomial approximation. A 13×13 window size is used for all the results.
Figure 10 shows a real cardiac catheterization speckled image obtained from [43], filtered with the enhanced versions of Frost and Lee and with the proposed Kuan filter (polynomial approximation). It can be observed that all the filters succeed to reduce speckle noise. However, the proposed adaptive Kuan filter outperforms the enhanced Frost and enhanced Lee filters in edge preservation. In other words, the Kuan filter with the proposed polynomial approximation yields visually better performance with the resulting image efficiently smoothed, while significantly preserving image details.

**FIGURE 10:** Cardiac catheterization image [43]. Top left: original (noisy) image. Top right: Enhanced Frost filtered image. Bottom left: Enhanced Lee filtered image. Bottom right: result of filtering with our proposed solution using the polynomial approximation. A 13×13 window size is used for all the results.

4.2 Performance of the Hybrid Wavelet-Spatial Despeckling Filter

This section deals with implementing the proposed hybrid wavelet-spatial domain filter. First, wavelet denoising with one decomposition level is applied on the 25 images of TID2013 database [39], each corrupted by 5 different levels of Gaussian speckle noise, then the results are compared to the same algorithm but using our enhanced Kuan filter to smooth the approximation image before wavelet reconstruction. We also apply wavelet denoising with the number of decomposition levels determined dynamically as proposed earlier, and compare the results to the same algorithm but using our enhanced Kuan filter to smooth the approximation image before wavelet reconstruction. Finally, the proposed enhanced Kuan filter is applied to the approximation image in each reconstruction level. Table 7 presents the gains (average, minimum, maximum, and standard deviation) with respect to the noisy image using different evaluation metrics (PSNR, CoC and ENL). It can be noticed that a Kuan smoothing on the approximation image with one decomposition level leads to an increase of 6.12 dB in average PSNR compared to 4.88 dB without the approximation image smoothing.

It is important to note that Kuan smoothing on the last decomposition level does not result in any significant enhancement. In fact, the smoothing strength of our adaptive Kuan filter depends on the NLK_AD parameter determined dynamically, which is very high in the approximation image of the last level, due to repetitive smoothing at every decomposition. Therefore, \( W \approx 1 \) (in (5)) and no filtering is performed in this case. Moreover, a Kuan smoothing filter on the approximation image of each level before reconstruction, leads to an increase of 0.2 dB in average PSNR compared to the case where approximation smoothing is performed only at the last decomposition level, without blurring the denoised image, which is obviously seen from the CoC metric.
TABLE 7: Evaluation metrics for wavelet-based denoising with and without Kuan smoothing. Sym1 wavelets and BayesShrink threshold selection technique are used.

<table>
<thead>
<tr>
<th>Gaussian noise</th>
<th>Gains with respect to the noisy image</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR</td>
</tr>
<tr>
<td></td>
<td>AVG</td>
</tr>
<tr>
<td>1L without Kuan smoothing</td>
<td>4.8808</td>
</tr>
<tr>
<td>1L with Kuan smoothing</td>
<td>6.1168</td>
</tr>
<tr>
<td>Dynamic without Kuan smoothing</td>
<td>7.1225</td>
</tr>
<tr>
<td>Dynamic with Kuan smoothing on the last level</td>
<td>7.1228</td>
</tr>
<tr>
<td>Dynamic with Kuan smoothing on all reconstructed levels</td>
<td>7.3229</td>
</tr>
</tbody>
</table>

4.3 Performance Evaluation for the Proposed Complexity Reduction Algorithm

In this section, the proposed accelerated algorithm is tested with different scenarios. Figure 11 shows an example of the denoising results obtained using the Lena image corrupted by a Gaussian speckle of variance 3100, Haar, db4, sym4, and bior6.8 wavelets, Soft and Hard VisuShrink and BayesShrink.

The algorithm results in two wavelet decomposition levels. The first and second columns of Figure 11 show the images obtained with one and two decomposition levels respectively. It can be seen that two decomposition levels give better performance than one decomposition level as expected. In addition, Universal Soft and Hard thresholding give more pleasant results than BayesShrink.

The same work is done with different standard test images, different noise variances and distributions, different thresholding methods, different wavelet types and families and using different decomposition levels, with images from [38, 39]. From our intensive test scenarios, we observed that a “best” denoising setup using wavelets cannot be generalized. In other words, there is no one optimal threshold selection technique and wavelet type that always give the best performance; performance depends on the wavelet thresholding algorithm, the wavelet family, the noise variance, the number of decomposition levels and the noisy image itself.

In the purpose of studying the complexity gain and PSNR loss when we stop at the estimated number of decomposition levels denoted \(L_{opt\ algo}\), we apply \(10^4\) times wavelet denoising on the images from TID2013 database [39] corrupted by five different levels of multiplicative Gaussian speckle noise, using BayesShrink threshold selection, ‘sym1’ wavelet type and different number of decomposition levels.

Figure 12 shows the histogram of the difference between the number of decomposition levels that yields the highest PSNR (i.e. to which we refer as “optimal”) and the number \(L_{opt\ algo}\) obtained with our proposed method. It can be observed that 49% of the time, our algorithm successfully determines optimal number of levels whereas it performs 1 additional level for 11% of the time, and one and two levels less than the optimal number for 23% and 17% of the time, respectively. It is important to mention that these numbers could vary with a different dataset or simulation setup, but they give a general overview of the behavior of the proposed algorithm. As a result of erroneously estimating the optimal number of decomposition levels, some PSNR loss is incurred.

Figures 13 and 14 respectively show the histogram and the cumulative density function (CDF) of the PSNR loss due to the proposed algorithm. It can be observed that the loss does not exceed
0.04 dB and thus, can be considered as negligible. Therefore, the proposed acceleration algorithm has a good performance in terms of robustness against PSNR loss.

*Figure 13* shows that there exists only 2% of no loss (loss of 0 dB), whereas *Figure 12* indicates that 49% of the images should have no loss, which could seem contradictory; in fact, referring to *Figure 14*, we notice that the PSNR loss is less than 0.01 dB for 49% of the images, which can be explained to be due to numerical rounding errors in computer simulations and therefore the results in *Figures 13 and 14* are not contradictory but rather consistent with those in *Figure 12*.
FIGURE 12: Histogram of the difference between $L_{opt\_algo}$ and the optimal number of decomposition levels.

FIGURE 13: Histogram of the PSNR loss when using the acceleration algorithm.

FIGURE 14: CDF of the PSNR loss when using the acceleration algorithm.

The advantage of the acceleration algorithm is the gain in complexity while keeping the PSNR loss negligible. In the purpose of analyzing this gain, the CPU time usage is computed for a number of ‘sym1’ wavelet decomposition levels followed by BayesShrink thresholding and the corresponding wavelet reconstruction. Figure 15 shows the processor time usage (average of 100 runs) on a 2 GHz Intel Core 2 Duo CPU, for images from the TID2013 database [39], corrupted by 5 levels of multiplicative Gaussian speckle noise. The figure shows that one could save 0.12 microseconds per pixel by performing one decomposition level instead of two, and 0.10 microseconds per pixel for two levels instead of three. The incremental gain exponentially decreases as the number of decomposition levels increases, which is expected due to the size of the approximation image that is exponentially reduced (by a power of 2 in each dimension) for each additional level.
FIGURE 15: CPU time usage according to the number of decomposition levels.

5. CONCLUSION
We have proposed a hybrid wavelet-spatial denoising algorithm based on two-stage processing. An automatic parameter selection technique for the spatial Kuan filter is proposed in the first stage, which is later used to spatially smooth the approximation image of the wavelet coefficients in the second stage. An acceleration algorithm for wavelets computation is also proposed. It consists in selecting a suitable number of wavelet decomposition levels yielding near-optimal performance and eliminating the need for additional and unnecessary decomposition levels.

The procedures have been materialized on a set of different images under different conditions. In order to evaluate the quality of the results, objective metrics were used besides subjective ones identifying the method's capacity in reducing speckle noise. The obtained results quality was highly encouraging, in terms of speckle reduction and image characteristics preservation, and proved that the proposed acceleration algorithm is advantageous in complexity gain which suggests that further research in this direction could be promising, in particular in the optimal choice of the decomposition level where spatial smoothing can be performed on the approximation component.

The evaluation of repetitive runs of the spatial optimized filter is of our interest, to verify whether it allows for better noise suppression or reduction while critically preserving edges and texture. Moreover, it is known that smoothing is usually performed in a pre-processing phase before edge detection. In the proposed algorithm, edge detection was performed for the purpose of smoothing. Therefore, we propose the use of our enhanced Kuan filter recursively for the sake of edge detection. Finally, we aim at reinforcing the use of the proposed algorithm to enhance the performance of the BM3D denoising algorithm, leaving this perspective an open discussion for future considerations.

6. ACKNOWLEDGMENT
This work has been partly supported by a research grant from the Higher Center for Research at the Holy Spirit University of Kaslik (USEK), Lebanon.

7. REFERENCES


[34] Shi Guangmin and Li Fudong, “Image Denoising with Optimized Subband Threshold,” *Proceeding of the Fifth International Conference on Computational Intelligence and Multimedia Applications, 2003.*


[38] ImageProcessingPlace.com, Image Databases.


