A Novel Curvelet Thresholding Function for Additive Gaussian Noise Removal

Yaser Norouzzadeh
Computer Engineering Department at
Shahid Bahonar University of Kerman
Kerman, Iran
YNR@Mail.uk.ac.ir

Mahdi Jampour
Computer and IT Department at
Kerman Institute of Higher Education (KIHE)
Kerman, Iran
Jampour@IEEE.org

Abstract—Improving quality of noisy images has been an active area of research in many years. It has been shown that removing additive Gaussian noise by nonlinear methods such as Wavelet denoising and Curvelet denoising had better results than classic approaches. However estimation of threshold and selection of thresholding function are still challenging tasks. In this paper, a new thresholding function is proposed for Curvelet thresholding and thresholding neural network is extended to use Curvelet coefficients instead of wavelet coefficients. This function is continuous and has higher order derivation. Therefore it is suitable for gradient decent learning methods such as thresholding neural network (TNN). This function is used by the TNN and threshold values for Curvelet sub-bands are estimated according to least mean square (LMS) algorithm. The experimental results show improvement in noise reduction from images based on visual assessments and PSNR comparing with well-known thresholding functions.

Key Words: Image Denoising, Curvelet thresholding, Thresholding function, Thresholding neural network.

I. INTRODUCTION

Images may be corrupted by noise in acquisition and transmission phases. Various noise removal methods reported by researchers. Linear methods have some side effects while removing noises. Therefore non-linear denoising methods in wavelet domain have been an active area for two decades. It is interested for researchers due to its ability to improve quality of noisy images.

Wavelet domain based noise removal techniques need some threshold value to removing small coefficients of detail sub-bands and preserving large coefficients; because small coefficients are usually noisy and large coefficients contains main features of image. Thresholding needs a thresholding function to decide how improve coefficients by using threshold. Therefore, estimating threshold and determining thresholding rules are still challenging problems in wavelet denoising.

The methods for estimation of threshold are divided in three groups. First group uses universal threshold value for all wavelet subbands of noisy image [1], [2]. In second group, namely subband-adaptive, the thresholds are determined differently for any detail subband [3]-[9]. Spatially adaptive thresholds are selected for each wavelet coefficient or some group of them in third group [10].

Hard and soft thresholding functions [1] are the most commonly used thresholding functions. Hard thresholding is discontinuous and is not differentiable. Soft thresholding is continuous but does not have first order derivation. Therefore these thresholding functions can’t be used in gradient based learning tools such as Thresholding Neural Networks [11], [12]. Some New thresholding functions such as garrote [13] and Zhang functions [12], [14] are reported in recent years. Garrote thresholding function has better properties than hard and soft thresholding, but they are not differentiable.

In recent years Curvelet transform has become more interested in image processing tasks such as image denoising. In this paper, concept of TNN in extended to use Curvelet coefficients instead of wavelet coefficients. In addition a new thresholding function is proposed which is continues and differentiable. Hence it has higher order derivation and can be used in gradient based learning tools such as Thresholding Neural Networks [11], [12]. Some New thresholding functions such as garrote [13] and Zhang functions [12], [14] are reported in recent years. Garrote thresholding function has better properties than hard and soft thresholding, but they are not differentiable.

In this paper, a new thresholding function is proposed for Curvelet thresholding and thresholding neural network is extended to use Curvelet coefficients instead of wavelet coefficients. In addition a new thresholding function is proposed which is continues and differentiable. Hence it has higher order derivation and can be used in gradient based learning tools such as Thresholding Neural Networks [11], [12]. Some New thresholding functions such as garrote [13] and Zhang functions [12], [14] are reported in recent years. Garrote thresholding function has better properties than hard and soft thresholding, but they are not differentiable. In the learning process the best threshold value is obtained for the proposed thresholding function. This paper includes following sections. Section 2 explains the existing wavelet denoising methods in brief. In section 3 Curvelet transform is described. Some background of TNN is reviewed in section 4. Section 5 describes the proposed thresholding function. Section 6 represents comparison of the proposed method with the most well-known image denoising methodologies. Finally section 7 concludes the paper.
II. WAVELET DENOISING

Let the image is defined by 
\[ f_{ij}, \quad i, j = 1, 2, ..., N \]
where \( N \) is an integer power of 2. If \( f_{ij} \) is corrupted with additive white Gaussian noise \( n_{ij} \), the observed noisy image \( g_{ij} \) will be given by (1):

\[ g_{ij} = f_{ij} + n_{ij} \quad i, j = 1, 2, ..., N \]

The goal of image denoising is to remove the noise from \( g_{ij} \) and estimate \( \hat{f}_{ij} \) which minimizes the mean square error (MSE):

\[ \text{MSE} = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} (\hat{f}_{ij} - f_{ij})^2 \]

In the recent years, there has been a large amount of research on the image denoising based on WT [15]. In wavelet domain, small wavelet coefficients are more likely to be noise, while large coefficients are major feature of original image. To decide which coefficient is small, a threshold is needed. Estimation of threshold is a major problem in this field. One of the first methods for estimation of threshold was VisuShrink [1]. In this method the value of threshold is obtained from

\[ \text{thr} = \log L \sigma \]

where \( \sigma \) is noise variance and \( L = N^2 \) is the size of image. SureShrink [1] which is based on minimizing the Stein’s unbiased risk [5] has better results than VisuShrink.

The most popular thresholding methods are soft and hard thresholding. They are given by (3) and (4), respectively.

\[
\begin{align*}
\text{Soft} & : & y_{\text{soft}}(t) &= \begin{cases} 
\text{sgn}(x(t)) \left( |x(t)| - \text{thr} \right) & |x(t)| > \text{thr} \\
0 & |x(t)| < \text{thr}
\end{cases} \\
\text{Hard} & : & y_{\text{hard}}(t) &= \begin{cases} 
x(t), & |x(t)| > \text{thr} \\
0, & |x(t)| < \text{thr}
\end{cases}
\end{align*}
\]

Where \( \text{thr} \) is threshold and \( x \) and \( y \) are modified and noisy version of image in the wavelet domain, respectively.

Zhang’s thresholding functions (5) and (6) are continues and differentiable. Hence they don’t have disadvantages of soft and hard functions. As a result they are suitable for use in the TNN.

\[
\begin{align*}
\text{Zhang} & : & \phi_{\text{zhang}}(x, x_i) &= \begin{cases} 
\frac{x}{\text{thr}} - \frac{\text{thr}}{2k+1} & x < -\text{thr} \\
1 & x \leq \text{thr} \\
\frac{x - \text{thr} + \frac{\text{thr}}{2k+1}}{\text{thr}^{2k+1}} & x > \text{thr}
\end{cases}
\end{align*}
\]

In (5) and (6), \( x \) is wavelet coefficients of noisy image, \( \text{thr} \) is threshold value, \( k \) and \( \lambda \) are parameters for adjusting the shape of thresholding functions.

III. CURVELET TRANSFORM

Curvelet transform theory is introduced in recent years and it is under development [17]-[19]. Major advantages of Curvelet are: Directionality and Anisotropy; Wavelet allows us to analysis image in three different directions (Vertical, horizontal and Diagonal), but Curvelet support more directions. To capture smooth curves, basis element should use a variety of shapes with different aspect ratios which cause Curvelet have Anisotropy [18]. Therefore Curvelet transform is more powerful than Wavelet to represent curves as is shown in Fig. 1. Thus Curvelet have better results in image denoising than wavelet [17]. In [20] the Curvelet transform based on Ridgelet transform is described. The Continues Ridgelet Transform (CRT) represents smooth functions and straight edges sparsely. The CRT for two dimensional functions \( f(x, y) \) has following form:

\[ f(x, y) = \int \int f(x \cos \theta + x \sin \theta - b) dx \sin \theta dy \sin \theta \]

Mentioned CRT can be calculated using Wavelet in a domain which defined by (8):

\[ R(\theta, t) = \int \int f(x, y) \delta(x \cos \theta + x \sin \theta - t) dx \sin \theta dy \sin \theta \]

Where \( (\theta, t) \in [0, 2\pi) \times R \) and \( \delta \) is Dirac distribution. Thus by applying one dimensional wavelet transform

Figure 1. Wavelet (left) versus Curvelet (right). Curvelet can represent curves better than Wavelet.
to $R_{\theta}(t)$, the CRT is obtaining [21]. Equation (9) shows this relation:

$$CRT_{a}(a,b,\theta) = a^{-\theta} \int_{a}^{\infty} \phi(t-b)R_{\theta}(t)dt$$ (9)

In image processing, edges are curves rather than straight lines. So Ridgelet alone isn’t effective to represent edges. Although Curvelets are based on Ridgelet, but Curvelet can separates different scales using band-pass filtering in special domain [22]. Curvelets occur in all scales, directions and locations as Ridgelet. Ridgelets have global lengths and variable widths, while Curvelets have variable lengths and widths.

![Curvelet transform flow graph](image)

Figure 2. Curvelet transform flow graph. The figure illustrates the decomposition of the original image into subbands followed by the spatial partitioning of each subband. The Ridgelet transform is then applied to each block.

Discrete Curvelet transform using scale and band-pass filter banks $(P,\Delta f,\Delta_{a}f,\Delta_{b}f,...)$ where band-pass filter $\Delta_{a}$ is near of $[2^{13},2^{15}]$ frequencies. The Curvelet transform steps are illustrated in Fig. 2.

In this work, real version of Fast Discrete Curvelet Transform (FDCT) [19] is used. FDCT is based on regular rectangular grid instead of tiled grid. Here, number of directions and levels are 16 and 6, respectively.

IV. THRESHOLDING NEURAL NETWORK

Thresholding Neural Network is combination of two concepts: neural network and wavelet thresholding [12]. In TNN, thresholding function is used instead of activation functions in feed forward neural network and TNN weights are constant and equal to 1. Consequently threshold value can be adjusted in learning phase. In other words in neural network activation function structure is constant and weights are changing in learning process but in TTN weights are constant and thresholding function structure can be tuned by threshold value. Fig. 3 represents TNN structure. Inputs of TTN are wavelet coefficients $(u_{i})$ of noisy image $(y)$ and outputs are thresholded wavelet coefficients $(v_{i})$. After inverse wavelet transform denoised image is available $(\hat{x})$.

In this paper, wavelet coefficients replaced by Curvelet coefficients.

TNN learning method is least mean squares (LMS). In step $j$ of learning, threshold value is adjusted using (10):

$$\Delta thr(j) = thr(j) + \Delta thr(j)$$ (10)

Where $\Delta thr(j)$ is calculated by (11):

$$\Delta thr(j) = -\alpha \frac{\partial MSE}{\partial thr}_{thr=thr(j)}$$ (11)

Where $\alpha$ is learning rate.

![Zhang Thresholding neural network structure](image)

Figure 3. Zhang Thresholding neural network structure [12]

V. PROPOSED THRESHOLDING FUNCTION

Equation (12) shows proposed thresholding function:

$$f(x,thr) = x - \frac{x}{\exp[(x/thr)^{2} - 1]} + \frac{1/8x}{\exp(x/0.7/thr)^{2}}$$ (12)

Where $x$ is noisy image and $thr$ is threshold value. This function is continues, differentiated and have higher order derivation. Hence it is suitable for TNN and any gradient decent learning algorithm.

Fig. 4 shows proposed thresholding function for $thr=3$. Comparison of proposed thresholding function with hard thresholding function and soft thresholding function is shown in Fig. 5. As can be seen thresholding function is near to zero in $[-thr,thr]$ and for other threshold values it converges to $f(x,thr) = x$.

Therefore noisy coefficients in $[-thr,thr]$ are shrunk. In
addition proposed thresholding function is one to one near of thr, while Zhang thresholding functions don’t have this property.

Equations (13) and (14) represent other proposed thresholding functions. They are continuous and have high order derivation, but shape of these thresholding functions is not as good as first proposed thresholding function. Also they have more parameters which make tuning of functions more difficult. Fig. 6 and fig. 7 represent these functions for thr=3.

\[ f(x,a) = x - \frac{a^2 x}{x^2 + a^2} \quad ; \quad a = 1500 \text{thr} \]  \hspace{1cm} (13)

\[ f(x,a) = x - \frac{a^2 x}{x^2 + a^2} \quad ; \quad a = 1/5 \text{thr} \]  \hspace{1cm} (14)

VI. EXPERIMENTAL RESULTS

Proposed thresholding function is used in TTN. The 256×256 “Lena” image is used in training phase for each noise variance. First Gaussian noise is added to this image and then FDCT of noisy image is computed to provide TNN inputs. Learning rate, convergence value are chosen 1e-6 and 1e-6, respectively and universal threshold value is obtained during learning process. TNN uses proposed thresholding function (6). In test phase, computed threshold value in learning phase is used by proposed thresholding function to denoised Curvelet coefficients of test images. Fig. 8 shows this process. Table 1 shows experimental results for various thresholding functions. It can be seen proposed thresholding function has produced better results in terms of Peak-to-Signal-Noise-Ratio (PSNR) value.

VII. CONCLUSION

In this paper an effective thresholding function is proposed which utilize TNN for tuning universal threshold value. This function is continuous and have higher order derivation which make it suitable for gradient decent learning algorithms such as TNN. In addition proposed thresholding function doesn’t need to additional parameters. Therefore problem of tuning parameters is resolved. In
previous works, wavelet coefficients are used by TNN, but in this work Curvelet coefficients is used. Using sub-band adaptive threshold value in TNN can be used in future works.

Figure 8. Block diagram of thresholding neural network using proposed thresholding function (12).

REFERENCES


Table 1: Comparing proposed thresholding function with other thresholding functions on PSNR value

<table>
<thead>
<tr>
<th>Image</th>
<th>$\sigma$</th>
<th>Soft</th>
<th>Hard</th>
<th>Garrote</th>
<th>Zhang 01</th>
<th>Proposed Thresholding function (12)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>10</td>
<td>24.89</td>
<td>27.95</td>
<td>26.22</td>
<td>29.91</td>
<td>32.05</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>22.66</td>
<td>25.04</td>
<td>23.57</td>
<td>25.74</td>
<td>28.16</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>21.61</td>
<td>23.48</td>
<td>22.27</td>
<td>23.92</td>
<td>26.05</td>
</tr>
<tr>
<td>Barbara</td>
<td>10</td>
<td>22.18</td>
<td>24.83</td>
<td>23.23</td>
<td>28.97</td>
<td>30.25</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>20.38</td>
<td>22.06</td>
<td>21.03</td>
<td>24.05</td>
<td>25.82</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>19.55</td>
<td>20.92</td>
<td>20.05</td>
<td>21.83</td>
<td>23.69</td>
</tr>
<tr>
<td>Cameraman</td>
<td>10</td>
<td>21.76</td>
<td>24.81</td>
<td>23.08</td>
<td>28.85</td>
<td>30.05</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>19.49</td>
<td>21.79</td>
<td>20.43</td>
<td>23.8</td>
<td>25.58</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>18.39</td>
<td>20.17</td>
<td>18.97</td>
<td>21.38</td>
<td>23.16</td>
</tr>
</tbody>
</table>