A Joint Multiscale Algorithm with Auto-adapted Threshold for Image Denoising

Jin He1 Yinpei Sun1 Ying Luo1 Qun Zhang1,2
1. Telecommunication Engineering Institute, Air Force Engineering University, Xi’an 710077 China
2. Key Laboratory of Wave Scattering and Remote Sensing Information (Ministry of Education), Fudan University, Shanghai 200433, China
Phone: + (86. 29) 84798434, email: hjelva@163.com

Abstract

Abstract—Curvelet transform is one of the recently developed multiscale transform, which can well deal with the singularity of line and provides optimally sparse representation of images with edges. But now the image denoising based on curvelet transform is almost used the Monte Carlo threshold, it is not used the feature of images’ curvelet coefficients effectively, so the best result can not be reached. Meanwhile, the wavelet transform codes homogeneous areas better than the curvelet transform. In this paper a joint multiscale algorithm with auto-adapted Monte Carlo threshold is proposed. This algorithm is implemented by combining the wavelet transform and the fast discrete curvelet transform, in which the auto-adapted Monte Carlo threshold is used. Experimental results show that this method eliminate white Gaussian noise effectively, improves Peak Signal to Noise Ratio (PSNR) and realizes the balance between protecting image details and wiping off noise better.

Keywords—Curvelet transform; Wavelet transform; Auto-adapted threshold; Image denoising; Peak Signal to Noise Ratio

I. INTRODUCTION

Many image processing tasks take advantage of sparse representations of image data where most information is packed into a small number of samples. Typically, these representations are achieved via invertible and nonredundant transforms. Currently, the most popular choice for this purpose is the wavelet transform [1]–[2]. Image denoising with wavelet transform has developed for many years, but now it is limited by the wavelet transform. Curvelet transform is a multiscale transform which can denote the curve singularity in the multidimensional signal generally. So it is used in image denoising widely [5]–[6]. But this curvelet transform involves a complicated index structure which makes the mathematical and quantitative analysis especially delicate, and it uses overlapping windows increasing the redundancy. Therefore, E.J.Cande proposed Fast Discrete Curvelet Transform (FDCT) which is based on the first generation of curvelet transform, and it is easier to realize the image denoising.

Though the image denoising based on curvelet transform has better efficiency in image edges, wavelet transform codes homogeneous areas better than curvelet transform. Meanwhile, The first generation of curvelet transform and the Fast Discrete Curvelet Transform are both using Monte Carlo threshold for image denoising, the feature of images’ curvelet coefficients are not used effectively with Monte Carlo threshold. In this paper, a new algorithm is proposed by combining the wavelet transform and the fast discrete curvelet transform. Moreover, a new threshold, which is called auto-adapted Monte Carlo threshold, is used in this algorithm.

II. FAST DISCRETE CURVELET TRANSFORM

Let \( \mathbf{x} \) be the collection of triple index \((j,l,k)\) , where \(j,l,\) and \(k=(k_1,k_2)\) are scale, orientation, and translation parameters, respectively. The curvelets are defined as functions of \( x \in \mathbb{R}^2 \) by
\[ \varphi_n(x) = \varphi(R_y(x - x^{(i,j)})e), \]  

(1)

In the above, \( \varphi \) is a waveform which is oscillatory in the horizontal direction and bell shaped (nonoscillatory) along the vertical direction. \( R_y \) is a rotation matrix with the rotation angle \( \theta_j = 2\pi \cdot 2^{1/2j} \cdot l \), with \( \lfloor \cdot \rfloor \) denoting the integer part and \( l = 0,1,\ldots \) such that \( 0 \leq \theta_j \leq 2\pi \), while the translation parameter is given by \( x^{(i,j)} = R_y^*(k_i,2^{j/2},k_j \cdot 2^{j/2}) \). The curvelet frame elements are obtained by anisotropic dilations, rotations, and translations of a collection of unit scale oscillatory blobs. The curvelet coefficients are given by

\[ c_n = \int_{\mathbb{R}^2} f(x) \varphi_n^*(x) dx, \]  

(2)

If there are a spatial variable \( x \), a frequency-domain variable \( \omega \), and with \( r \) and \( \theta \) polar coordinates in the frequency-domain, let a pair of windows, which we will call the “radial window” and “angular window,” respectively. These are both smooth, nonnegative and real-valued, with \( W \) taking positive real arguments and supported on \( r \in (1/2,2) \) and \( V \) taking real arguments and supported on \( t \in [-1,1] \). These windows will always obey the admissibility conditions:

\[ \sum_{j=-\infty}^{\infty} W^2(2^j r) = 1, \quad r \in (3/4,3/2) \]  

(3)

\[ \sum_{j=-\infty}^{\infty} V^2(t-l) = 1, \quad t \in [-1/2,1/2], \]  

(4)

for each \( j \geq j_0 \), we introduce the frequency window \( U_j \) defined in the Fourier domain by

\[ U_j(r,\theta) = 2^{-3j/4} W(2^{-j} r) V(\frac{2^{j/2} \theta}{2\pi}). \]  

(5)

Where \( \lfloor j/2 \rfloor \) is the integer part of \( j/2 \). Thus the support of \( U_j \) is a polar “wedge” defined by the support of \( W \) and \( V \), the radial and angular windows, applied with scale-dependent window widths in each direction.

The curvelet coefficients can be evaluated directly in the frequency domain \( U_j \),

\[ c_n = \int_{\mathbb{R}^2} \hat{f}^k(\omega) \varphi_n^*(\omega) d\omega \]  

(6)

The window \( U_j \) locates the frequencies near a polar wedge. But this is not adapted to Cartesian arrays in practical implementations. In the Cartesian case, the digital analog of coefficients can be given by [7]

\[ c_n = \int_{\mathbb{R}^2} \hat{f}^k(\omega) U_j(R_y^* \omega) e^{i\langle n,\omega \rangle} d\omega, \]  

(7)

\[ S_n = \begin{bmatrix} 1 & 0 \\ -\tan \theta & 1 \end{bmatrix}. \]

Here, \( h = (k_i \cdot 2^{-j}, k_j \cdot 2^{-j/2}) \). A crucial difference between (7) and (6) is the use of the shear matrix \( S_n \) instead of the rotation matrix \( R_y \).

Suppose now that we are given a Cartesian array \( f[t_1,t_2] \), \( 0 \leq t_1,t_2 < n \) and let \( \hat{f}[n_1,n_2] \) denote its 2D discrete Fourier transform

\[ \hat{f}[n_1,n_2] = \sum_{l_1,l_2=0}^{2^{1-j}} f[l_1,l_2] e^{-i2\pi \langle (n_1,n_2),(l_1,l_2) \rangle / n}, \quad -n/2 \leq n_1,n_2 < n/2. \]  

(8)

Assume the \( \hat{U}[n_1,n_2] \) is supported on some rectangle of length \( L_{1,2} \) and width \( L_{2,2} \)

\[ l^j = \{(n_1,n_2) : n_0 \leq n_1 < n_0 + L_{1,2}, n_0 \leq n_2 < n_0 + L_{2,2} \}, \]  

where \( (n_0,n_0) \) is the index of the pixel at the bottom-left of the rectangle. Because of the parabolic scaling, \( L_{1,2} \) is about \( 2^j \) and \( L_{2,2} \) is about \( 2^{j/2} \). With these notations, the curvelet coefficients can be rewritten as

\[ C^\omega(j,l,k) = \sum_{n_1,n_2 \in \ell^j} \hat{f}[n_1,n_2] e^{-i2\pi \langle (n_1,n_2),(j,l,k) \rangle / n}. \]  

(10)

III. AUTO-ADAPTED MONTE CARLO THRESHOLD

A. MONTE CARLO THRESHOLD

The image denoising with curvelet transform is based on Monte Carlo threshold which can be defined as follow [2]. Let \( T \) be the noisy curvelet coefficients, the hard-thresholding rule for estimating the unknown curvelet coefficients can be expressed as

\[ \hat{T} = \begin{cases} \hat{T} & |T| \geq k\sigma \sigma_j \\ 0 & |T| < k\sigma \sigma_j \end{cases}, \]  

(11)

where \( \sigma \) is the noise standard deviation and \( \sigma_j \) is the curvelet coefficients of white noise whose mean is 0 and variance is 1. We have the scale value \( k = 4 \) for the first scale while \( k = 3 \) for the others.

Here the curvelet coefficients of white noise is estimated by Monte Carlo method adequately, but the scale value \( k \) is so simple that the energy distribution property of images' curvelet coefficients is not used efficiently, so the detail information of images would be strangled to excess. If we can construct a kind of auto-adapted scale values which is based on the structure of curvelet coefficients, the results of image denoising would be better.

3.2 THE STRUCTURE OF CURVELET COEFFICIENTS

The scale of curvelet transform is correlate with the size of the image, it is calculated by \( \text{scale} = \log_2(n) - 3 \), where \( \text{size}(img) \). For example, an image with the size \( 256 \times 256 \) should be divided into 5 scales; its coefficients' structure can be expressed as

<table>
<thead>
<tr>
<th>Scale</th>
<th>Direction</th>
<th>Curvelet coefficients' structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>32 × 32</td>
</tr>
<tr>
<td>2</td>
<td>4 × 8</td>
<td>16 × 12, 12 × 16, 12 × 16, 12 × 16</td>
</tr>
<tr>
<td>3</td>
<td>4 × 8</td>
<td>32 × 22, 22 × 32, 32 × 22, 22 × 32</td>
</tr>
<tr>
<td>4</td>
<td>4 × 16</td>
<td>64 × 22, 22 × 64, 64 × 22, 22 × 64</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>256 × 256</td>
</tr>
</tbody>
</table>

As is shown in table, there is only one direction in the first scale, which contains low frequency coefficients of the image and consists of a matrix of the size \( 32 \times 32 \). The main part of image is in this scale. There are four main directions which all contain eight directions in the second and third...
scale, and the fourth scale has four main directions which all contain sixteenth directions. The middle frequency of image which contains the main detail information is included in these scales. There is one direction in the fifth scale, which includes the high frequency of image and is consisted of a matrix of the size $256 \times 256$, it only contains a little detail information.

Fig.1 is the result of Lena image with the size $256 \times 256$ which passes through the curvelet reconstruction. We can see that the main energy is included in the first scale, and present the degressive trend from the second scale to the forth scale, there is only a little information and energy in the fifth scale. If we can set threshold which is in the foundation of the energy distribution, the detail information of image would be protected better.

![Fig.1 the result of Lena image’s curvelet transforms](image)

3.3 AUTO-ADAPTED MONTE CARLO THRESHOLD

From Eq.7, we can find that the curvelet coefficients is simulate to the wavelet coefficients, which obey the binary distribution. So the Auto-adapted algorithm in wavelet image denoising can be used here. In our experiments, we obtain the auto-adapted Monte Carlo threshold, whose received step can be summarized as

1. Take curvelet transform to the white noise which obeys the distribution $\mathcal{N}(0,1)$ for 300 times, and average the result. We can obtain the noise standard deviation $\sigma_j$ from the results.
2. Set the scale value $C$.
3. Replace the scale value $k$ with the new scale value $C$, the threshold can be rewritten as

$$T = C \sigma_j$$

IV. A JOINT MULTISCALE ALGORITHM FOR IMAGE DENOISING

Because of wavelet only have good performance at representing point singularities, it is a good transform for denoising homogeneous areas [4] while ill-suited for removal of noise around the edges. And experimental results also show that curves provide optimally sparse representation for edges of images while homogeneous areas are a problem for us. So if both the transform can be combined for image denoising, the results would be better. We propose a method to solve this problem; the steps can be expressed as

1. For the discrete curvelet transform and wavelet transform to be translation invariant, a 2-D cycle spinning is implemented on subbands first. We choose the shifted around in eight.
2. Take the image which is shifted for wavelet transform.
3. Keep up the high frequency coefficients and take the low frequency coefficients to zero, then take wavelet inverse transform and obtain the high frequency information of image.
4. Take the high frequency parts of image for curvelet transform.
5. Deal with the curvelet coefficients with Auto-adapted Monte Carlo threshold, then take curvelet inverse transform, and obtain the disposed high frequency coefficients.
6. Take the disposed high frequency coefficients for wavelet transform and combine it with the low frequency coefficients, then take wavelet inverse transform.
7. Inverse shift the image and average the result, the denoised image will be obtained.

V. EXPERIMENT RESULTS

A $256 \times 256$, 8-bit grayscale Lena image and Barbara image were used in this experiment. Gaussian white noise $n$ with standard deviation $\sigma = 10, 15, 20, 25$ was added to the image $I$. All these images were denoised by wavelet and curvelet transform. The hard threshold methods were used in here. Peak signal-to-noise ratio (PSNR) was used to estimate
the quality of the denoising. The PSNR is given by

$$PSNR = 10 \log_{10} \left( \frac{\max (I(u,v))^2}{MSE} \right),$$

where $\tilde{I}$ is the image after denoising and $I$ is the initial image, $M$ and $N$ denote the length and width of the image and

$$MSE = \frac{1}{M} \frac{1}{N} \sum_{u=1}^{M} \sum_{v=1}^{N} (\tilde{I}(u,v) - I(u,v))^2.$$

Table 2 The compare of denoised images’ PSNR

<table>
<thead>
<tr>
<th>Image</th>
<th>Lena</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma$</td>
<td>10</td>
</tr>
<tr>
<td>Noise image</td>
<td>28.14</td>
</tr>
<tr>
<td>Denoised by wavelet</td>
<td>28.49</td>
</tr>
<tr>
<td>Denoised by curvelet (Monte Carlo threshold)</td>
<td>31.33</td>
</tr>
<tr>
<td>Denoised by curvelet (Auto-adapted Monte Carlo threshold)</td>
<td>34.70</td>
</tr>
<tr>
<td>Denoised by the joint multiscale algorithm</td>
<td>36.56</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image</th>
<th>Barbara</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma$</td>
<td>10</td>
</tr>
<tr>
<td>Noise image</td>
<td>28.14</td>
</tr>
<tr>
<td>Denoised by wavelet</td>
<td>28.74</td>
</tr>
<tr>
<td>Denoised by curvelet (Monte Carlo threshold)</td>
<td>30.77</td>
</tr>
<tr>
<td>Denoised by curvelet (Auto-adapted Monte Carlo threshold)</td>
<td>33.32</td>
</tr>
<tr>
<td>Denoised by the joint multiscale algorithm</td>
<td>35.29</td>
</tr>
</tbody>
</table>

From the Fig. 2, we can find that in the Lena image, the curvelet denoising based on Monte Carlo threshold is not better than wavelet denoising; this is due to that the Lena image does not have abundant detail information and the curvelet transform brings the sliver noise in the image. In the Barbara image, the results of curvelet denoising based on Monte Carlo threshold are better, but not good enough. As is shown in Table 2, the Auto-adapted Monte Carlo threshold is more suitable for curvelet denoising obviously, it reconstruct the Lena image’s cockpit, hair and Barbara images’ headband, table cloth well, but it still has some bug in images’ homogeneous areas. The joint algorithm solves this problem effectively. From the last image of Fig. 2, we find that not the edges of images, but also the homogeneous areas of images reach the best results.

VI. CONCLUSIONS

A joint multiscale algorithm, which is based on auto-adapted Monte Carlo threshold, is proposed in this paper. Experiments show that the new threshold is more suitable for curvelet transform, it realize the balance between protecting image details and wiping off noise better. And the joint algorithm, which uses wavelet transform to denoise homogeneous areas and curvelet transform to denoise areas with edges, improving the denoised images both in the PSNR and also in visual appearance, outperforming all other methods used in the experiments.

VII. REFERENCES