# Aerial Images Directional Denoising by Splitting-Signals

Fatma T. Arslan Andrew K. Chan and Artyom M. Grigoryan

Department of Electrical Engineering
University of Texas at San Antonio

# Outline

- Introduction
- Tensor Representation of Images.
- Wavelet Denoising Methods
- Directional Denoising of Aerial Images by Tensor Transform
- Conclusions

# Introduction

The oceonagraphic aerial images are commonly used for studying of ocean current flow, seabed structures, rock locations, sediment formation, etc. Usually, these aerial images are captured with wave clutters. The clutters are classified into two types:

- ripple wave (long-waves)

- spark wave (short-waves)

Waves are added to the test images. These clutters are directionally located.

# Introduction

Tensor Transform are used to denoise these images. By Tensor Transform, images are represented in the form of 1-D "independent" splitting-signals. The splitting-signals show certain directional effects in fourier domain. In this study, we benefit from these features of splitting-signals.

# A New Look on Image and 2-D DFT

- Analyze the tensor and paired representations of images in forms of sets of 1-D ”independent” splitting-signals.

- Tensor Representation transfers image to $3N/2$ splitting-signals of length $N$.

- Seperately process all (or only a few) splitting-signals and then calculate and compose the 2-D DFT of the processed image by means of new 1-D DFTs of the processed splitting-signals.

# Tensor Representation

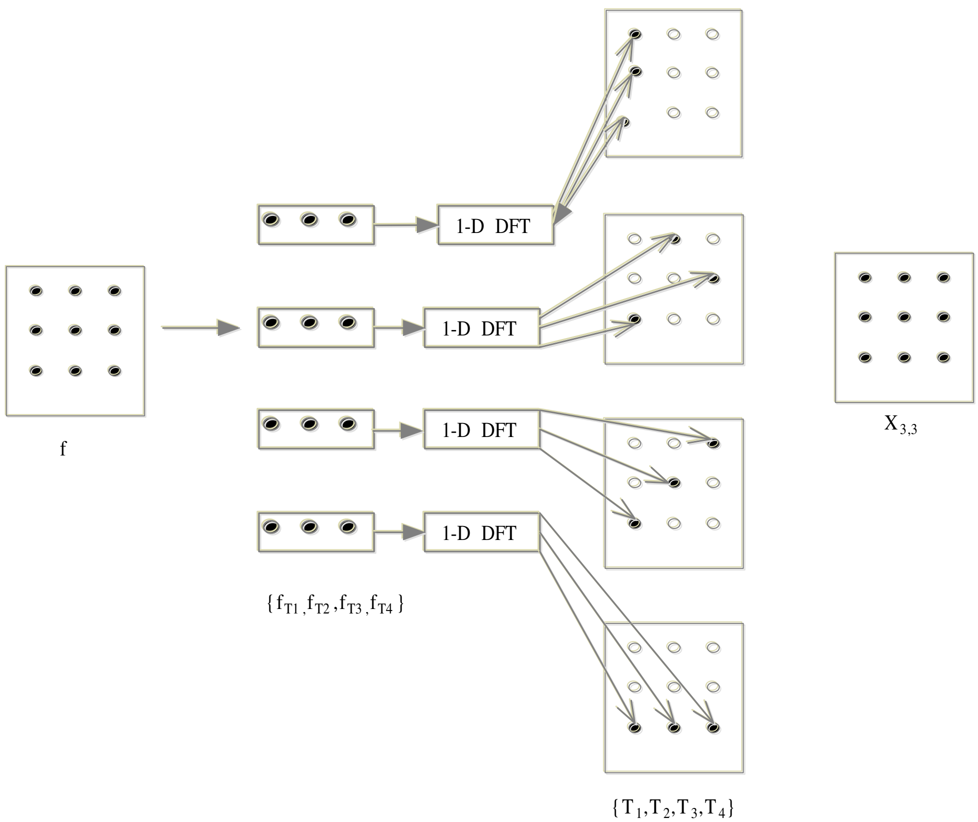



Diagram of $3 \times 3$ point sequence $f$ transformation into four signals $f_{T_1}, f_{T_2}, f_{T_3}, f_{T_4}$ whose 1-D DFT defines the 2-D DFT of $f$.

# Tensor Representation

The irreducible covering $\sigma = (T)$ of $X_{N,N}$ is defined by the following cyclic groups with generators $(p,s)$

$$T_{p,s} = \left\{(0,0),(p,s),(\overline{2p},\overline{2s}),\ldots,(\overline{(N-1)p},\overline{(N-1)s})\right\} \qquad (1)$$

For instance, the covering of the $3 \times 3$ set $X$ is defined by $\sigma = (T_{1,0}, T_{1,1}, T_{1,2}, T_{0,1})$ as shown

$$\underset{X_{3,3}}{\begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{bmatrix}} = \underset{T_{1,0}}{\begin{bmatrix} \bullet & \circ & \circ \\ \bullet & \circ & \circ \\ \bullet & \circ & \circ \end{bmatrix}}, \underset{T_{1,1}}{\begin{bmatrix} \circ & \circ & \bullet \\ \circ & \bullet & \circ \\ \bullet & \circ & \circ \end{bmatrix}}, \underset{T_{1,2}}{\begin{bmatrix} \circ & \bullet & \circ \\ \circ & \circ & \bullet \\ \bullet & \circ & \circ \end{bmatrix}}, \underset{T_{0,1}}{\begin{bmatrix} \circ & \circ & \circ \\ \circ & \circ & \circ \\ \bullet & \bullet & \bullet \end{bmatrix}}$$

# Tensor Representation

The covering $\sigma = (T_{p,s})$ reveals the 2-D DFT. The following property holds for the Fourier transform:

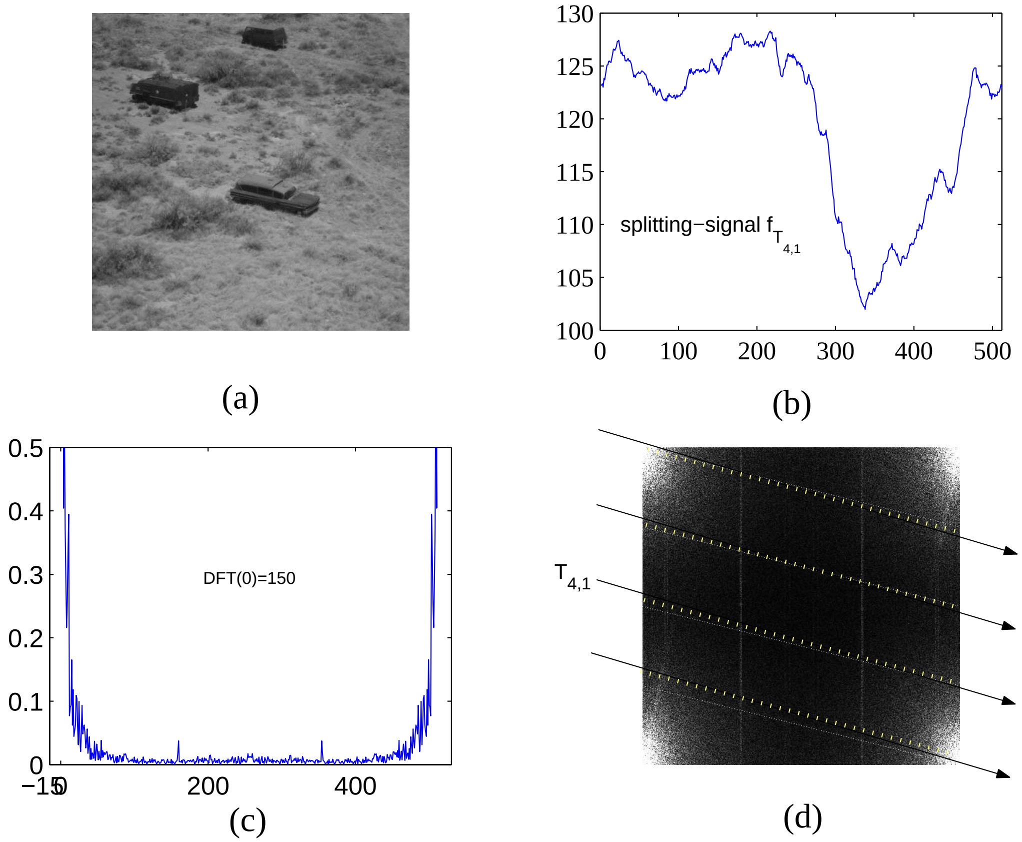
$$F_{\overline{kp},\overline{ks}} = \sum_{t=0}^{N-1} f_{p,s,t} W^{kt}, \quad k = 0\colon (N-1) \qquad (2)$$

where

$$f_{p,s,t} = \sum_{V_{p,s,t}} f_{n,m}\,, \quad t = 0\colon (N-1) \qquad (3)$$

and $V_{p,s,t} = \{(n,m);\ np + ms = t \bmod N\}$.

# Tensor Representation




(a) Original image. (b) Splitting-signal $f_{T_{4,1}}$. (c) The 1-D DFT of the splitting-signal.

(d) Arrangement of values of the 1-D DFT in the 2-D DFT of the image.

# Wavelet Denoising Methods

The algorithm used in this paper to denoise 1-D signal is described as follows:

- Perform the forward wavelet transform with $j$ level.
- Discard certain coefficients that has noise.
- Reconstruct the denoised signal by the inverse wavelet transform.

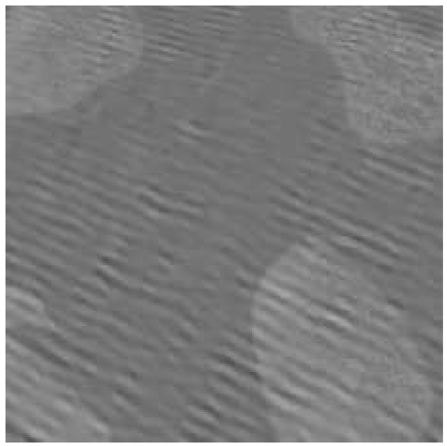
The coefficients that are to be discarded are all high frequency coefficients at $j-1$ levels.

# Directional Denoising by Tensor Transform

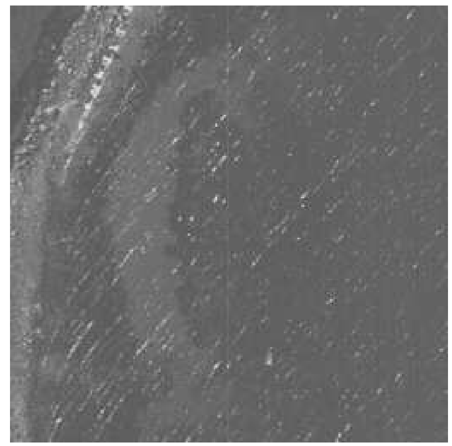
- We propose a novel method combining Tensor Transform by wavelet transform and denoise these images.

- To remove the long-waves subband filtering technique is used. 2-D subband filtering is reduced to 1-D filtering by using Tensor Transform. To remove the short-waves SMEME filter is used. Tensor Transform is used as pre-step of SEME filter.

# Directional Denoising by Tensor Transform



(a)



(b)

(a) Aerial Image 1 (b) Aerial Image 2

# Directional Denoising by Tensor Transform

Modelling long-waves and short-waves

Long-waves are modeled as :

$$\begin{cases} c_{ripple}(x, y_o) = A \cdot \sin(\frac{2 \cdot \pi}{T(x)}) \\ T(x) = \alpha + \sqrt{\frac{x}{\beta}}. \end{cases} \tag{4}$$

$\alpha$ : controls the initial period of long-waves. $\beta$ : controls the variation of the ripple frequencies. $A$ : amplitude of the wave. $y_o$ : location at which the long-wave occurs. The $c_{ripple}$ function has higher frequency near the locations where $x = 0$ and lower frequency far from there.

# Directional Denoising by Tensor Transform

short-waves are modeled as:

$$\begin{cases} c_{spark}(x,y) = \dfrac{A}{1+(\sqrt{2}-1)\cdot(\frac{D(x,y)}{D_o})} \\ D(x,y) = \sqrt{\dfrac{(x-x_o)^2}{a^2}+\dfrac{(y-y_o)^2}{b^2}}. \end{cases} \tag{5}$$

$(x_o, y_o)$ : location of the short-wave. $n$ : affects the shape of waves. $a, b$, and $D_o$ : determine the width and length of each wave. $A$ : amplitude of the short-wave peak.

# Directional Denoising by Tensor Transform

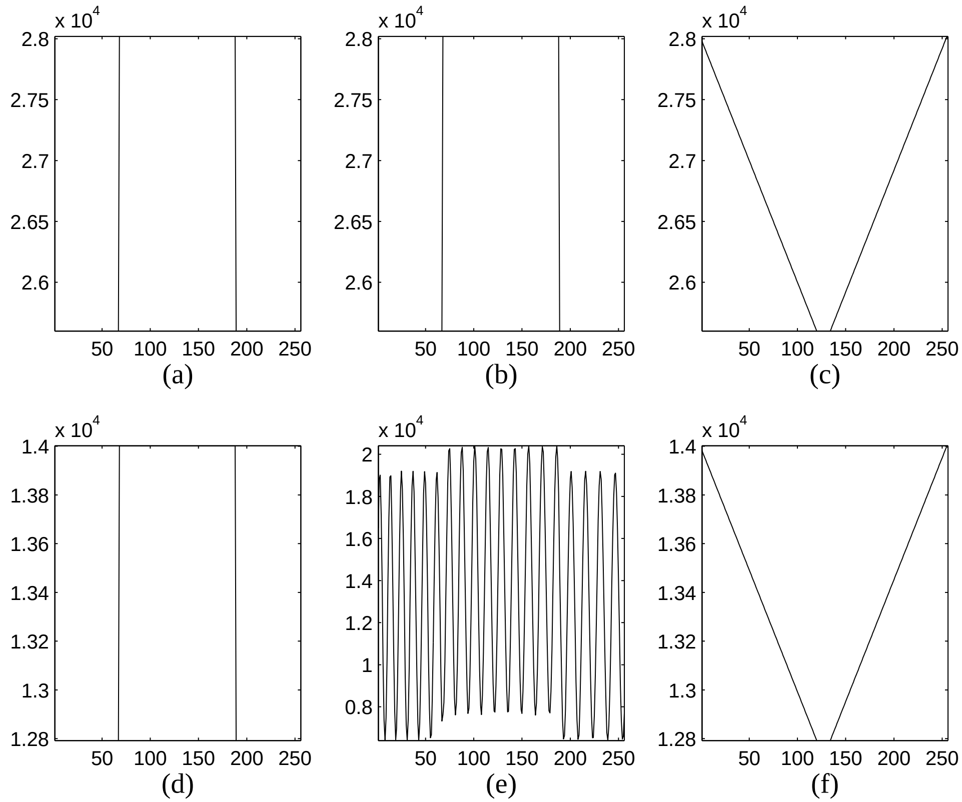
(a) (c) (e)

(a) Original Test Image (b) Long-waves added (c) Short-waves added noisy images

Removal of Long waves

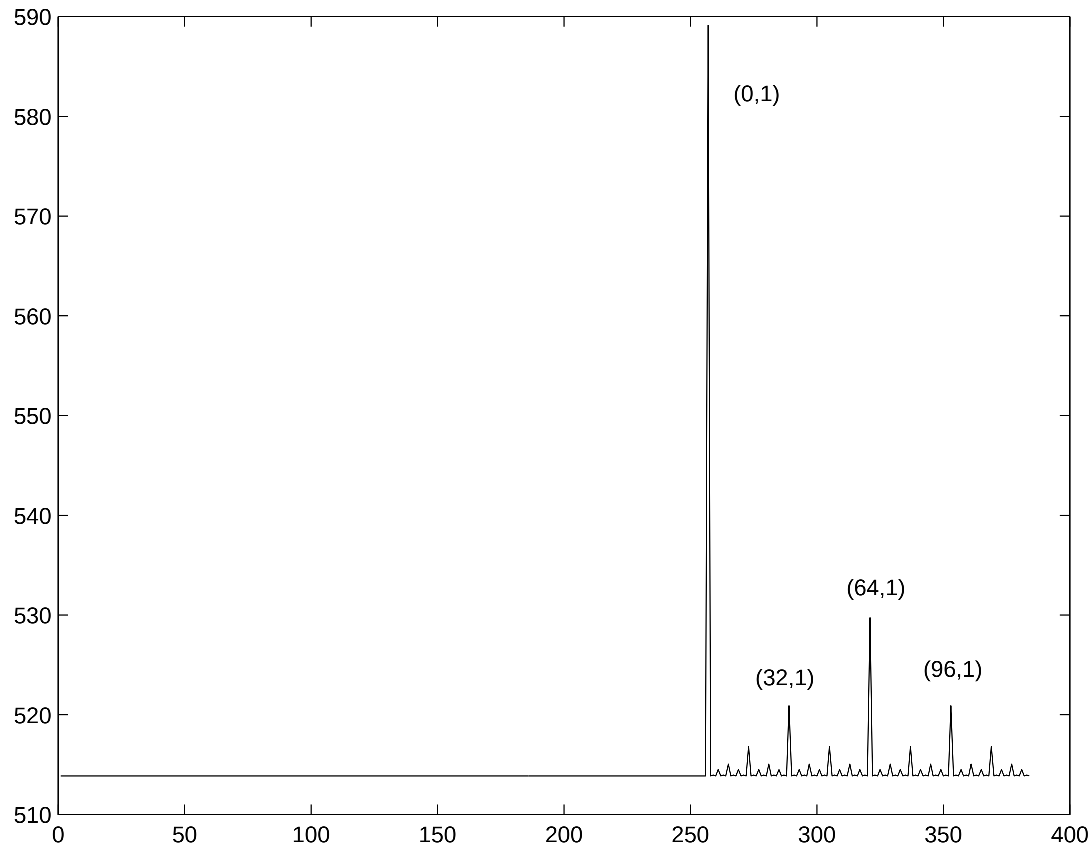
We denoised this wavy images by using image-signals. The Image-signals have directional effect on image structure. We chose the clutters in test images to be horizontal then we expect the horizontal image-signal to be corrupted most in the wavy image.

# Directional Denoising by Tensor Transform




(a) $f_{T(0,1)}$, (b)$f_{T(1,0)}$, (c)$f_{T(1,1)}$, (d) noisy $f'_{T(0,1)}$, (e)$f'_{T(1,0)}$ and, (f) $f'_{T(1,1)}$

# Directional Denoising by Tensor Transform

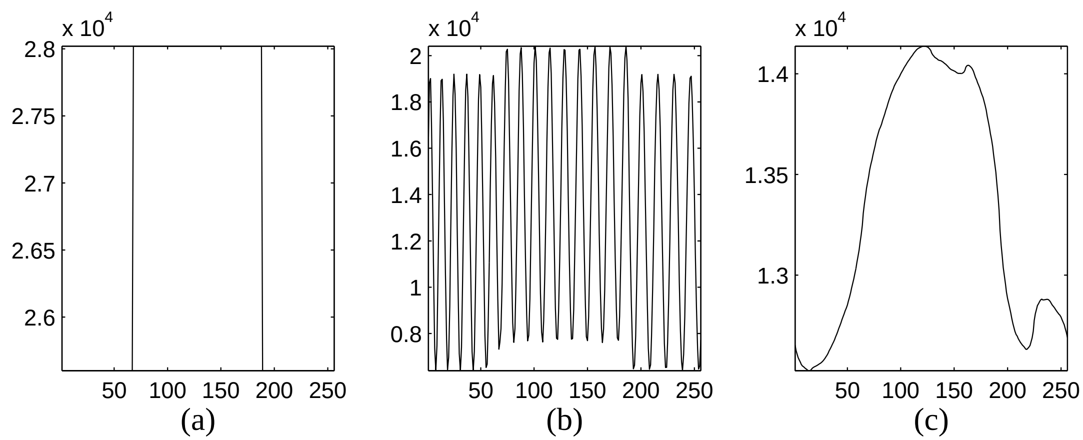



Difference of the noisy Image signals from original Image signals

# Directional Denoising by Tensor Transform

Since image-signal is $f_{T(0,1)}$ the mostly corrupted one, we denoised only this signal. We used the wavelet denoising on this image signal $f_{T(0,1)}$. We took 3 level wavelet transform (in this experiment we used Daubechies wavelet three) and discarded highpass coefficients and obtained the denoised signal. After combining the denoised image-signal with noisy image we got a wave free image.

# Directional Denoising by Tensor Transform




(a) Original Image-signal $f_{T(0,1)}$ (b) Denoised Image-signal $f_{T(0,1)}$.

# Directional Denoising by Tensor Transform

(a) (b) (c)

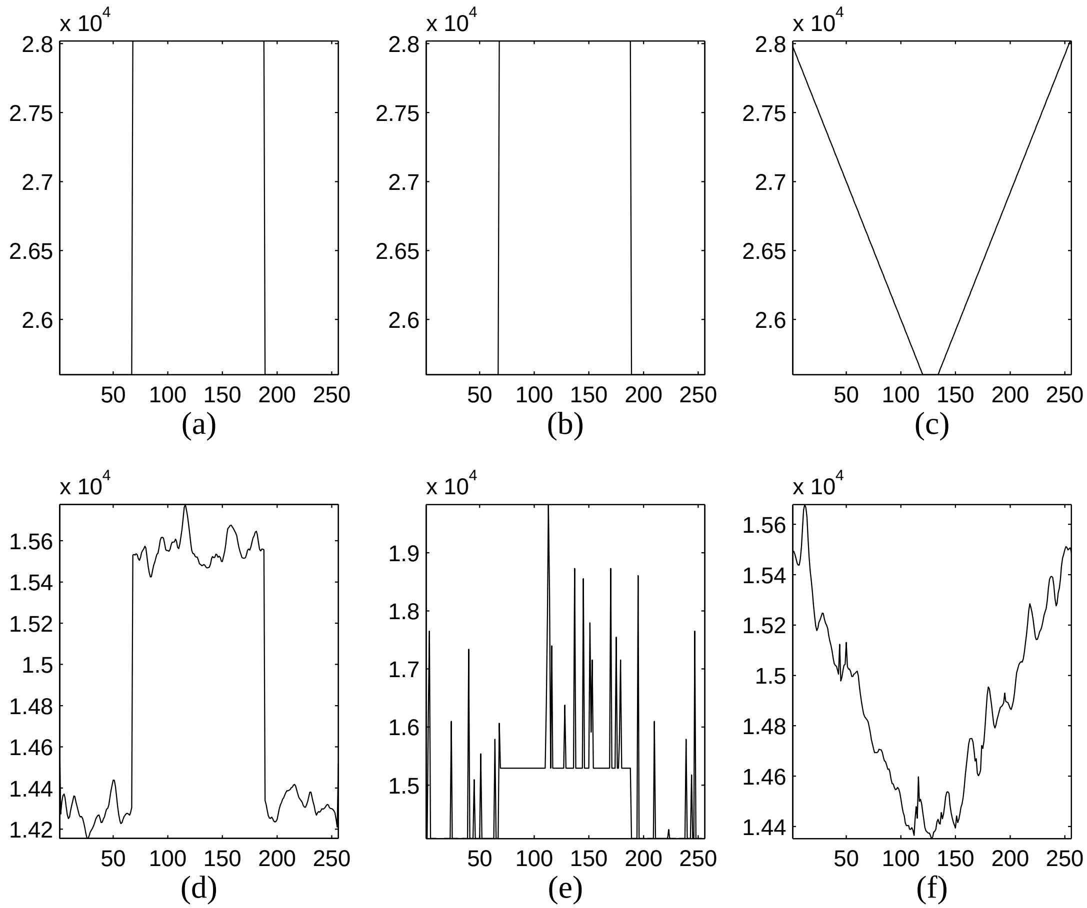
(a) Original Image, (b) Noisy Images, and (c) Denoised Image.

# Directional Denoising by Tensor Transform

Removal of Short Waves

Image-signals are used as well and image-signal $f_{T(0,1)}$ is mostly corrupted. Image-signal $f_{T(0,1)}$ is used in the first step of the design of SMEME filter. SMEME filter is refined verrsion of SMEM (spectral-spatial maximum exclusive mean filter) filter. The clutters should be located, then the abnormal pixels at that points are removed by SMEME filtering. The noisy image-signal $f'_{T(0,1)}$ has some peaks, the locations of these peaks are the locations of shortwaves in the spatial domain.

# Directional Denoising by Tensor Transform




(a) $f_{T(0,1)}$, (b) $f_{T(1,0)}$, (c)$f_{T(1,1)}$, (d) noised $f'_{T(0,1)}$, (e)$f'_{T(1,0)}$, (f)$f'_{T(1,1)}$.

# Directional Denoising by Tensor Transform

The motivation behind SMEME filter is to remove the abnormal pixels of the noisy image but keep the background information data. Therefore the size of the filter window is chosen such as a part of it would cover abnormal pixels and part of it would cover background. Window with size $3 \times 3$ centered at position $(m, n)$ and $X(m, n)$ be the pixel value of image $X$ at position $(m, n)$.

# Directional Denoising by Tensor Transform

SMEME filtering algorithm

- Find the peaks in noisy image-signal $f_{T(0,1)}$ by differentiating the signal. The places of these peaks correspond to the places of short-waves (m,n), where the filter window center will be located.

- Find last K pixels $X_i(m,n)$ which has smallest gray values within the filter window and calculate the average by

$$Avg = \sum_{i=1}^{K} X_i(m,n)/K$$

# Directional Denoising by Tensor Transform

- Replace the pixel at the center of the filter window $X_i(m,n)$ by $Avg$ obtained in Step 2

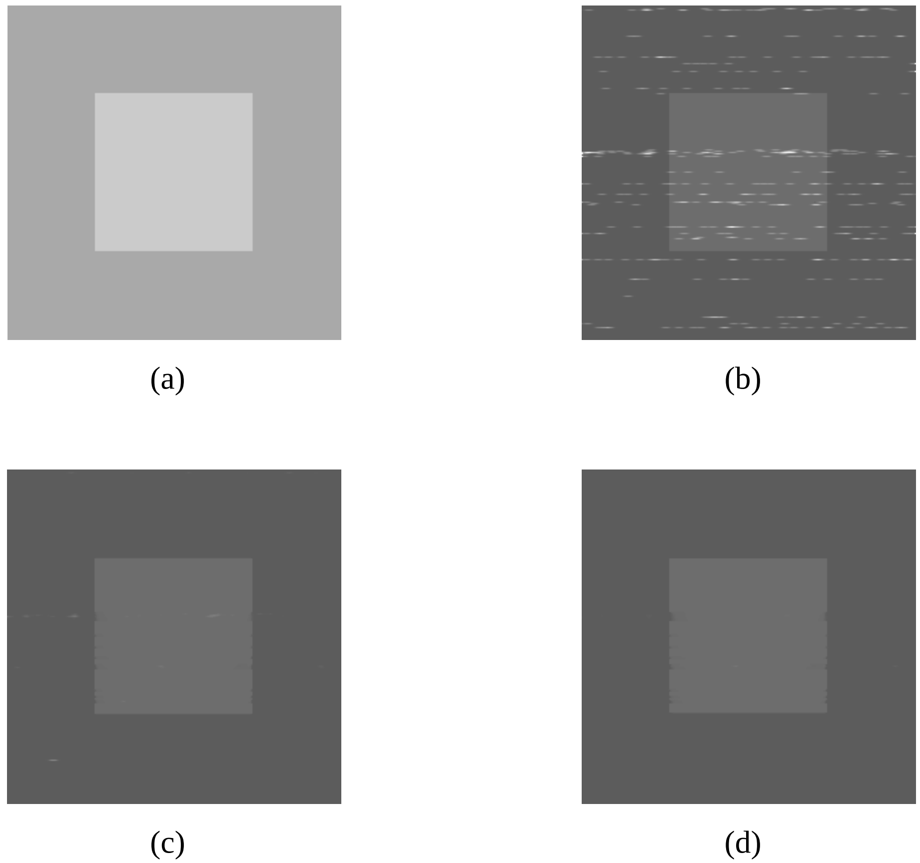
- Apply this procedure again, if needed, to the denoised image

# Directional Denoising by Tensor Transform

The final algorithm of denoising short waves :

- Calculate the tensor representation of noisy image and find the image-signal that has the information in clutter direction.
- Find the peaks of the 1-D image-signal by differentiating.
- Center SMEME filter at the clutter and smooth the waves

# Directional Denoising by Tensor Transform



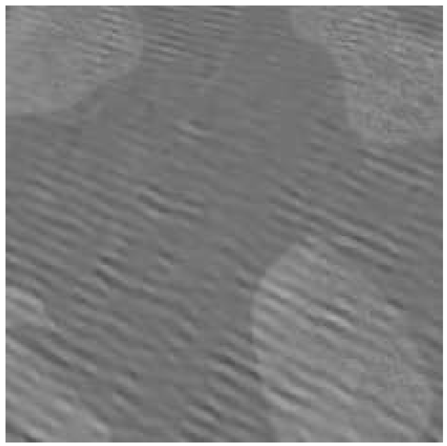

(a) Circle image (b) Noisy image (c) SMEME filtered (d) SMEME filtered two times

# Directional Denoising by Tensor Transform

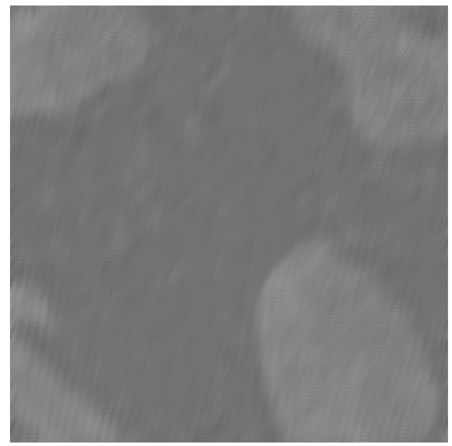
| SNR(dB) | longwave | shortwave |
|---|---|---|
| noisy | 16.28 | 15.26 |
| Denoised | 34.40 | 25.88 |

SNR of noisy and original images

# Directional Denoising by Tensor Transform



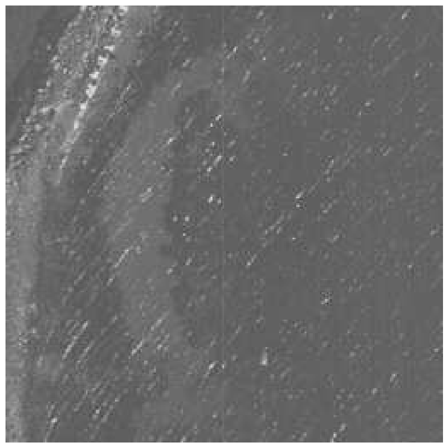
(a)



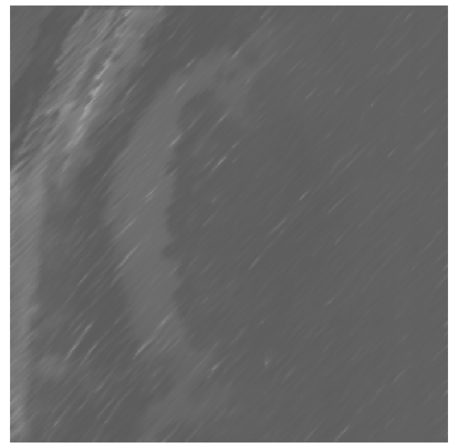
(b)

(a) $f_{T(0,1)}$, (b) $f_{T(1,0)}$, (c)$f_{T(1,1)}$, (d) noised $f^{'}_{T(0,1)}$, (e)$f^{'}_{T(1,0)}$, (f)$f^{'}_{T(1,1)}$.

# Directional Denoising by Tensor Transform



(a)



(b)

(a) $f_{T(0,1)}$, (b) $f_{T(1,0)}$, (c)$f_{T(1,1)}$, (d) noised $f'_{T(0,1)}$, (e)$f'_{T(1,0)}$, (f)$f'_{T(1,1)}$.