



## Curvelet transform with learning-based tiling

Hasan Al-Marzouqi<sup>a,\*</sup>, Ghassan AlRegib<sup>b</sup>

<sup>a</sup> Electrical Engineering Department, Petroleum Institute, Abu Dhabi, United Arab Emirates

<sup>b</sup> School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA



### ARTICLE INFO

**Keywords:**

Directional transforms  
Curvelets  
Signal representation  
Sparsity  
Denoising

### ABSTRACT

Compact signal and image representations are of crucial importance in a variety of application areas. Wavelet and wavelet-like transforms typically divide the frequency plane in a systematic non-adaptive approach. In this paper, we propose a learning-based method for adapting frequency domain tiling using the curvelet transform as the basis algorithm. The optimal tiling that better represents a single image or a given class of images is computed using denoising performance as the cost function. Simulated additive white Gaussian noise is removed from a given image using a thresholding algorithm. The curvelet tiling that generates maximal denoising performance as measured by PSNR or the logarithm of mean squared error (MSE) is considered optimal. The major adaptations considered are the number of scale decompositions, angular decompositions per scale/quadrant, and scale locations. A global optimization algorithm combining the three adaptations is proposed. Signal representations by adaptive curvelets are shown to outperform default curvelets in partial reconstruction error. Furthermore, adaptive curvelets are used in compressed sensing recovery of incomplete seismic datasets and face images. Visual and numerical improvements across a variety of images and different subsampling ratios are reported. Finally, adaptive curvelets denoising performance is tested on seismic datasets. Our results establish clear numerical and visual performance advantages over the default curvelet transform and the non-local means algorithm (NLM).

### 1. Introduction

Signal representation techniques can be classified into two main categories, transform-based and data-driven techniques. Transform based techniques divide the frequency domain of the input signal into frequency bands that are typically non-adaptive and independent from the content of the signal. Wavelet transforms, curvelets [1], contourlets [2], and shearlets [3,4] are examples of transforms belonging of this category. Data-driven methods rely on the use of greedy algorithms that minimize  $l^0$  and  $l^1$  sparsity measures. Examples of such algorithms include the method of optimal directions (MOD) [5] and KSVD [6].

In the transform category, frequency domain divisions are regularly performed in a systematic approach using a dyadic or other constant dilation factors. The number of frequency divisions and their locations are typically selected in non-adaptive manner. In this research, the goal is to develop a learning-based approach for frequency domain tiling adaptation using the curvelet transform as a basis algorithm. The optimal tiling representing a single image or a class of images is found using denoising performance as a cost function.

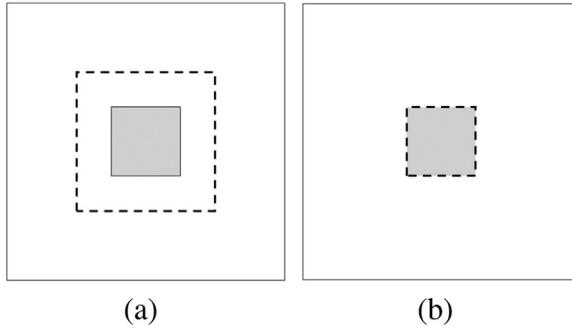
An example illustrating an advantage of content-based tilings is shown in Fig. 1. The figure shows the spectrum of a sample image

composed of a squared region with constant magnitudes located at the center of the Fourier plane. Two approaches for dividing the frequency plane are presented. First, assume that we apply a simple transform that divides the Fourier plane into two scales as shown in Fig. 1(a). The boundary between the highpass and lowpass regions is set using a dyadic scale, where the scale boundary divides the Fourier plane evenly. The dyadic scale is widely used in wavelet and wavelet-like transforms. Transform coefficients are generated by taking the Inverse Fast Fourier Transform (IFFT) of the highpass and lowpass regions. In this example, coefficients representing the inner region represent the centered squared region and the region with zero magnitudes surrounding it. In Fig. 1(b), a two-scale transform is applied with the boundary of the squared centered region used as the boundary separating the lowpass and highpass regions. In this example, the lowpass coefficients are compacted into a single impulse with magnitude that depends on the magnitude of Fourier coefficients in the squared region. The coefficients representing the outer highpass region are equal to zero. Thus, by adapting the divisions of the Fourier plane, the transform generated a compact representation of the image. Coefficients sparsity increased due to the improved transform design.

Wavelet packets [7] are one of the earliest methods proposed to

\* Corresponding author.

E-mail addresses: [hasalmarzouqi@pi.ac.ae](mailto:hasalmarzouqi@pi.ac.ae) (H. Al-Marzouqi), [alregib@gatech.edu](mailto:alregib@gatech.edu) (G. AlRegib).

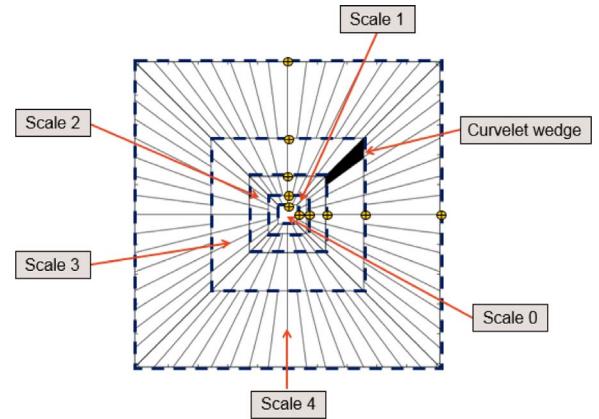


**Fig. 1.** The Fourier plane of a sample synthetic image divided to two scales using (a) a dyadic scale and (b) content-based divisions. The dashed line represents the boundary between the outer and inner scales. The gray squared region contains constant non-zero Fourier coefficients. The white region surrounding the inner gray region indicates zero Fourier magnitudes.

customize the tiling of the frequency plane. Wavelet packet decomposition finds the optimal representation of data using entropy as a cost function. An alternate rate-distortion based optimizer was proposed in [8]. A discrete wavelet transform tiling and a possible tiling achieved using wavelet packets are illustrated in Fig. 2. The adaptations allowed by wavelet packets do not include scale locations, which we define as coordinates of the Fourier transform plane that determine the outer boundary of each scale. Instead, scale locations are determined using a dyadic scale. Traditional wavelet packets implementations also inherit the lack of directional selectivity from the standard wavelet transform.

A multitude of transforms that better represent directional information are now available. Examples include curvelets [1], contourlets [2], shearlets [4], and others [9–11]. The proposed data dependent tiling approach will be demonstrated using the curvelet transform. Compared to other directional transforms, curvelets provide a direct relationship with Fourier plane contents. Curvelet coefficients are generated using Fast Fourier Transform (FFT) and inverse Fast Fourier Transform operations. In comparison, contourlets rely on spatial domain filtering and convolution methods. Since its introduction, the curvelet transform has shown successful performance improvements in a wide range of application areas including denoising [12,13], image fusion [14], edge enhancement [15], compressed sensing data recovery [16], face recognition [17,18], and texture image retrieval [19,20].

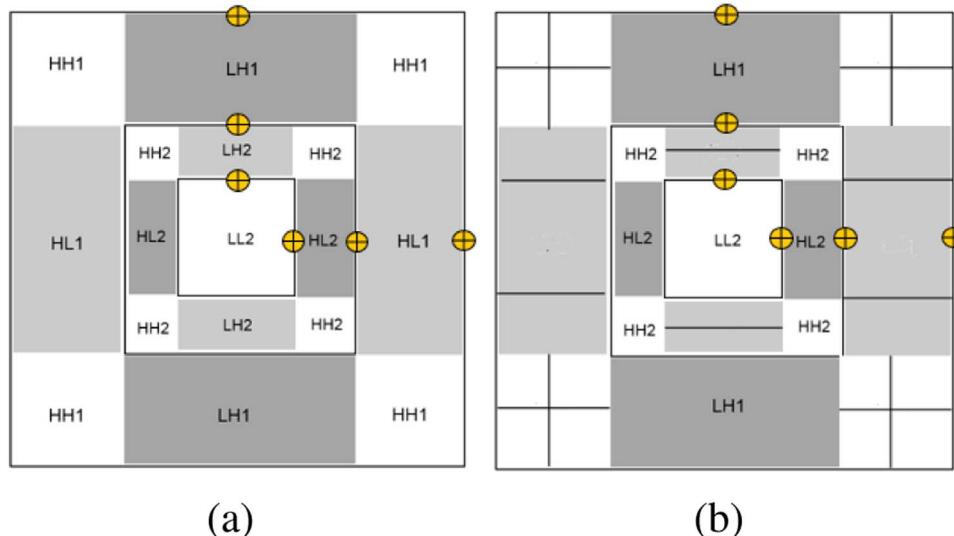
Curvelet coefficients are generated using the 2D FFT. FFT is first applied to an image. Next, the 2D Fourier plane is divided into different



**Fig. 3.** Default curvelet tiling. A five scales decomposition is shown. Each scale is divided into  $16 \times 2^{(j-1)/2}$  wedges, where  $j$  is the scale number.

scales. Each scale is further divided into a number of different tiles as shown in Fig. 3. In this work the goal is to adapt the number, size, and locations of curvelet tiles according to a denoising based objective function. Additive white Gaussian noise is removed from a given image using a thresholding algorithm. The PSNR or the logarithm of mean squared error (MSE) is used as the criterion for assessing denoising performance. The rational behind this cost function is presented in Section 3.1. Naturally, the approach used in this work can be extended to other directional transforms and to the broader family of wavelets and wavelet-like transforms. The number of images required by the proposed approach to learn the data-dependent tiles depends on the availability of the original data samples. In applications where access to the original data is possible, such as feature selection and data compression, the proposed learning method can use the image of interest. On the other hand, in applications where the original data is subsampled or corrupted by occlusion or noise a set of training data is used to learn the desired data dependent tiling.

A number of algorithms that propose using the traditional transform based approach in a data-dependent manner have been developed. These algorithms introduced adaptivity by operations applied prior and after the division of signal content into unique subbands. For example, the Easy Path Wavelet Transform (EPWT) [21] divides the input image into 1D paths chosen to maximize the correlations between its data points. Next, the transform applies a standard wavelet transform to the acquired paths. In [22], the arrangement of inputs to the wavelet



**Fig. 2.** Wavelet and wavelet packets coverage of the frequency domain: (a) Wavelet transform tiling and (b) a possible wavelet packet tiling where the HH1, HL1, and LH2 regions are further divided into sub-bands. Scale locations are denoted by the  $\oplus$  markers.

transform is designed to minimize a coding cost function. It is worth noting that pre- and post-transform adaptive operations can possibly benefit from the adaptive scheme presented in this paper. For example, EPWT wavelet's machinery handles vectors with highly correlated entries and a better representation of such vectors can be obtained using customized frequency domain tilings.

On the other hand, a number of data-driven greedy algorithms proposed incorporating transform based frequency divisions into their dictionaries. The wavelet matching pursuit algorithm [23] and the more recent wavelet KSVD [24] fall into this category. These algorithms are likely to benefit from the adaptivity scheme developed in this paper. Since using a transform equipped with customized frequency domain divisions is expected to yield performance improvements over general purpose transforms.

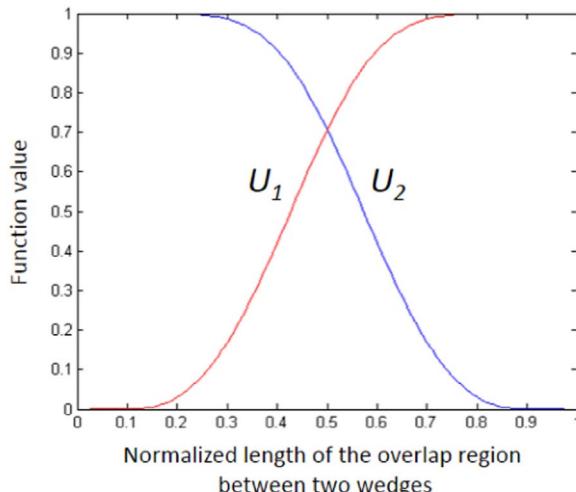
We would like to note that papers [25–27] use the default curvelet transform. Adaptivity in the title of these papers refers to operations applied to regular curvelet coefficients. Parts of the results in this paper appeared earlier in [28,29]. In this paper, we provide detailed analysis of the developed algorithm including rational, expanded algorithm description, and computational complexity. New application areas and experiments are also included such as image quality assessment and partial reconstruction error.

The remainder of this paper is organized as follows. An overview of the technical details of the curvelet transform is provided in the next section. The proposed objective function is presented in Section 3. In Section 4, we introduce the optimization algorithm used to train curvelet tiles. Section 5 presents examples illustrating adaptive curvelets coefficient decay and partial reconstruction error. Compressed sensing recovery and seismic image denoising results are presented in Section 5 as well. This is followed by conclusions and a view of future work in Section 6.

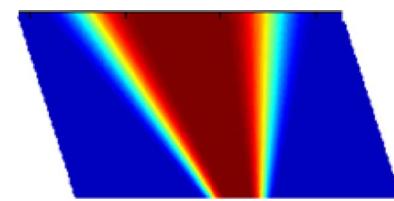
## 2. The curvelet transform

### 2.1. Flow of the algorithm

The first step in computing the curvelet transform of an image is to apply the 2D FFT. Next, the Fourier plane is divided into tiles as shown in Fig. 3. The inner most level is not directional. Periodic extension is used in the outer scale. To prevent discontinuities and enhance sparsity, each wedge is multiplied by overlapping complementary smoothing functions  $U_1$  and  $U_2$  on each of its four sides (Fig. 4). Each FFT pixel value  $p$  in the smoothing region between two wedges is multiplied by  $U_1$  and  $U_2$ .  $pU_1$  is used in computing the coefficients representing the first



**Fig. 4.** Two complementary smoothing functions  $U_1$  and  $U_2$  shown in red and blue respectively.



**Fig. 5.** A smoothed curvelet tile shown with its parallelogram-shaped support region. Function values decrease from 1 to 0 as color changes from red to blue.

wedge. Similarly,  $pU_2$  is used in computing the coefficients representing the second wedge.

Perfect recovery from the coefficients is made possible by ensuring that the smoothing functions are normalized, so that the following property holds [1]:

$$U_1^2 + U_2^2 = 1. \quad (1)$$

Curvelet coefficients are computed by taking the inverse FFT of each smoothed wedge. Taking the inverse FFT on these non-rectangular wedges can be performed by a “wrapping” operation. In this operation, every wedge is localized inside a parallelogram. Fig. 5 shows a smoothed curvelet tile along with its parallelogram-shaped support region.

Using the periodicity of the Fourier transform, the 2D plane is tiled with copies of this parallelogram in a manner similar to what is shown in Fig. 6. Note that the same plane tiling can equivalently be generated using copies of the rectangular region surrounding the origin. This allows us to compute the inverse FFT of the parallelogram-shaped region by taking the inverse FFT of the rectangle surrounding the origin with the parallelogram's length and height.

Another possible method for taking the inverse FFT involves using a shear and an interpolation procedure and is detailed in [1]. The inverse curvelet transform works by “reversing” forward transform operations.

### 2.2. Default parameters of the curvelet transform

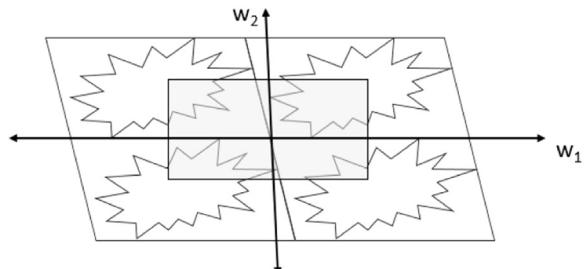
Scales are distributed in a dyadic manner where each scale is larger in size than the previous one by a factor of two. The outermost level is chosen to be one-third of the size of the image. The number of scales  $J$  is dependent on image dimensions  $N_1$  and  $N_2$ , and is given by

$$J = \lceil \log_2(\min(N_1, N_2)) - 3 \rceil, \quad (2)$$

where  $\lceil \cdot \rceil$  is the ceiling function. The number of divisions per scale is equal to  $16 \times 2^{[(j-1)/2]}$ , where  $j$  is the scale index. The subtraction by three ensures a minimum size of 8 by 8 coefficients in the inner most scale. In the next section, different strategies will be used to adapt default curvelet parameter values with the goal of finding the set of parameters that maximizes or minimizes a given cost function.

## 3. Adaptive curvelet

In adaptive curvelet, we propose frequency domain tilings described by the number of scale decompositions, scale locations, and angular



**Fig. 6.** The inverse FFT of the parallelogram-shaped regions is computed by taking the inverse FFT of the shaded rectangular region surrounding the origin.

divisions per scale/quadrant pair. In this section, we will propose algorithms adapting these parameters. This will be followed by a global optimization algorithm combining the three adaptations. The frequency-domain tiling that generates the best improvement in the cost function value is considered optimal. The cost function used in this work is described in the next section. This is followed by sections describing the algorithms used to search for the optimal curvelet tiling.

### 3.1. Denoising performance and connections with basis pursuit denoising

Enhancing sparsity by minimizing the  $l_1$ -norm of transform coefficients has proven to be a successful technique in a variety of application areas including signal denoising, feature extraction, and compressed sensing. However, this approach is not applicable to our problem since adapting curvelet tiles will change the total number of coefficients. The  $l_1$ -norm is not defined between vectors with different lengths. The proposed cost function relies instead on a denoising performance measure that is computed in the spatial domain.

Denoising performance was measured by adding Additive White Gaussian Noise (AWGN) with zero mean and standard deviation  $\sigma$  is added to the input image. Next, a value for the number decomposition scales is chosen by the proposed scale selection algorithm. This is followed by search algorithms finding the optimal scale locations and number of angular divisions for each scale/quadrant pair. The algorithms converge returning the optimal curvelet tiling for the given image. Given  $J$  number of scales, the optimization problem characterizing denoising-based adaptive curvelets can be described mathematically by

$$\arg \max_{S,A} \text{PSNR}(I, \hat{I}_{S,A}), \quad (3)$$

where  $I$  is the input image,  $\hat{I}_{S,A}$  is the denoised image. This image is obtained using adaptive curvelets described by parameters  $S$  and  $A$ ,  $S$  is a  $2 \times J$  matrix describing adaptive curvelet scale locations,  $A$  is a  $2 \times (J - 1)$  matrix that determines the number of angular decompositions used in each curvelet scale/quadrant pair.  $J$  is the number of decomposition scales.  $S$  and  $A$  have two rows referencing the two curvelet quadrants over which computation of curvelet coefficients is performed.  $\text{PSNR}(I, \hat{I})$  is function that computes the peak signal to noise ratio between images  $I$  and  $\hat{I}$ .  $\text{PSNR}$  is defined as:

$$\text{PSNR} = 20 \log_{10} \left( \frac{\max_I}{\sqrt{\text{MSE}}} \right), \quad (4)$$

where  $\max_I$  is the maximum possible intensity value in image  $I$ .  $\text{MSE}$  is the mean squared error. The logarithm of mean squared error ( $\text{MSE}$ ) can also be used as an alternative cost function. The logarithm of  $\text{MSE}$  is recommended for datasets that do not follow standard image quantization schemes. It is the function we used to optimize curvelets to better represent seismic data.

Using denoising as a cost function to guide the transform's behavior can be viewed as taking the reverse strategy of the basis pursuit denoising algorithm [30]. In basis pursuit denoising (BPDN), a restored image  $\hat{x}$  representation in a basis  $\beta$  is computed by solving the following minimization problem:

$$\hat{x} = \arg \min_x \frac{1}{2} \|y - x\|_2^2 + T \|x\|_1, \quad (5)$$

where  $x$  indicates noise-free representation of the data in  $\beta$ ,  $y$  is the acquired noisy signal represented in  $\beta$ , and  $T$  is a weight factor. Eq. (5) is solved by iterating through different coefficient realizations, until the one minimizing (5) is found. The computed representation is assumed to be optimal in terms of denoising performance. In this paper the coefficients representing a given signal are adapted by changing transformation parameters (i.e. changing the basis  $\beta$ ), until the representation that maximizes denoising performance is found. Transform parameters that generate the maximal denoising perfor-

mance are considered optimal.

Besides its use in denoising, the basis pursuit formulation has been used successfully in compressed sensing recovery [31] and coding [32,33]. The results in this paper demonstrate that curvelet tiles learned by optimizing a denoising-based cost function, provide improved performance in compressed sensing recovery and partial reconstruction error. We used the Gaussian noise model to be faithful to basis pursuit denoising which is designed to handle Gaussian noise.

Similar to wavelets, Gaussian noise can be eliminated in the curvelet domain using thresholding methods. Thresholding algorithms require an estimate for the noise level  $\sigma_{j,l}$  at each level  $j$  and orientation  $l$ . The thresholding algorithm used in this work [12] applies a simple Monte-Carlo simulation to estimate a wedge dependent noise standard deviation from the estimate of noise standard deviation  $\sigma$  corrupting the original image. The local noise level for wedge  $j, \sigma_{j,l}$  for scale decomposition  $S$  and angular decomposition  $A$  is estimated by applying the curvelet transform with the same scale and angular decomposition to a white noise image with  $\sigma$  noise level. Noise standard deviation  $\sigma_{j,l}$  is then computed using the generated coefficients. This local noise estimation can be repeated for a few more iterations to reach a reliable estimate. Let  $\hat{c}$  be the noisy curvelet coefficients. Restored image coefficients  $c$  are given by hard-thresholding according to the following equation:

$$c = \begin{cases} \hat{c}, & \text{if } |\hat{c}| \geq k\sigma_{j,l} \\ c = 0, & \text{if } |\hat{c}| < k\sigma_{j,l} \end{cases}, \quad (6)$$

where  $k = 4$  for the outer scale, and is equal to three otherwise.  $k$  values were determined experimentally. Varying these values allows for a balance between denoising blurring artifacts and accurate noise removal.

Besides maximizing denoising performance, other possible cost functions include Shannon's entropy, Gini index [34], and the coefficient of variation. A comparison between the performances of these cost functions in optimizing default curvelet tiling with seismic datasets can be found in [35].

### 3.2. Scale selection algorithm: number of decomposition scales

A connected region of high magnitude FFT values is observed in the center of the frequency plane of most natural images. An example of such activity is shown in Fig. 7(a) and (b). The figure shows a plot of FFT log-magnitudes for three images.

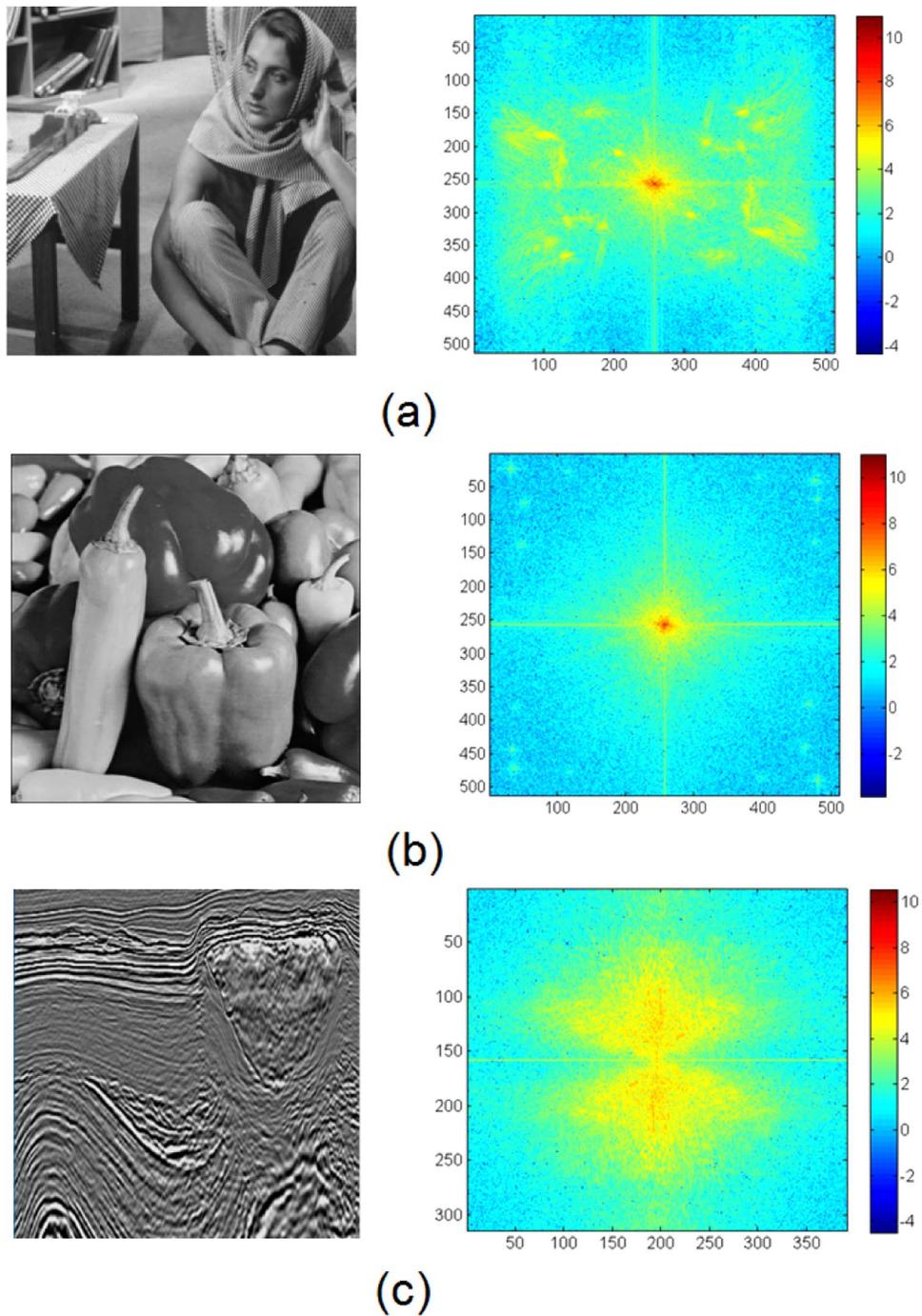
To achieve a better representation of the signal of interest, it is preferable to avoid dividing such connected regions into tiles while applying the curvelet transform. We developed a heuristic algorithm to determine the number of scales guided by this observation [28]. The algorithm works as follows:

Step 1 defines a quantity called mid-range  $mR$ , as follows:

$$mR \triangleq \sqrt{\max |FFT| \times \min |FFT|}. \quad (7)$$

- If the minimum Fourier magnitude value was equal to zero, the next minimum value is chosen.
- Step 2 let  $D = D_{\text{initial}} = 8$ . This is the smallest possible value for the size of the inner most curvelet level.
- Step 3 set a square surrounding the origin of the frequency domain with a diagonal length equal to  $D$  pixels.
- Step 4 if any FFT magnitude value within the square is smaller than the mid-range value, exit the algorithm and return  $D$ . Otherwise, go to step 5.
- Step 5 set  $D = D + 2$  and return to step 3.

The optimal number of scales is found using the value  $D$  computed



**Fig. 7.** Barbara, Peppers, and a seismic image and their corresponding Fourier log magnitude plots.

in step 4 above. Using dyadic scaling the optimal number of scales  $J$  is computed as follows:

$$J = \lceil (\log_2(\min(N_1, N_2) - 1) - (\log_2(D) - 1)) \rceil \quad (8)$$

where  $N_1$  and  $N_2$  are the image vertical and horizontal dimensions respectively. Eq. (8) computes the difference between the number of scales needed to reach the origin of the frequency domain from the edge of the image (i.e. total number of scales) and the number of scales needed to reach the origin from the outer edge of the finest level scale. The operator  $\lceil \cdot \rceil$  is the ceiling function.

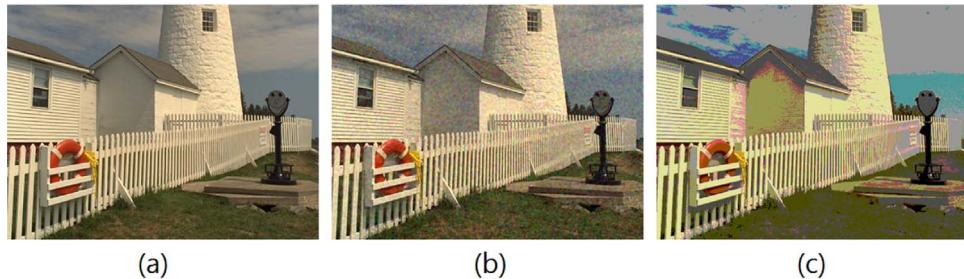
Seismic and fingerprint data are two examples of images that do not

exhibit the centered impulse behavior. A seismic example is shown in Fig. 7(c). A small value of  $D$  can detect such cases, where Eq. (2) is used instead for selecting the number of scales. The formula for the number of scales thus becomes

$$J = \begin{cases} \left\lceil \log_2\left(\frac{\min(N_1, N_2)}{D}\right) \right\rceil & D > 8, \\ \lceil \log_2(\min(N_1, N_2)) - 3 \rceil & D \leq 8. \end{cases} \quad (10)$$

This algorithm was tested on 500 test images acquired by crawling the website DPChallenge.com. 465 images were found to have a centered impulse region. Those images were corrupted by additive white Gaussian noise (AWGN) with  $\sigma = 20$ .

The scale selection algorithm generates denoising improvements in



**Fig. 8.** Relationship between image quality and the size of the centered impulse region  $D$  (Section 3.2) (a) Sample image  $D = 25$ . (b) Original image corrupted with spatially correlated noise  $\text{PSNR} = 27.04$ ,  $D = 23$ . (c) Original image corrupted with quantization noise  $\text{PSNR} = 21.10$ ,  $D = 19$ .

83.0% of cases. In images where adaptive curvelets outperform default curvelets, the average improvement in PSNR is 0.20 dB. Average PSNR improvement over all images is found to equal 0.10 dB. The number of scales chosen by the scale selection algorithm is going to be used as the basis for further improvements achieved by the remaining parts of the optimization algorithm.

It is noteworthy that the size of the centered impulse response is observed to have a relationship with image quality. For many distortion types, distorted images had a smaller impulse region size.

Fig. 8 shows a sample image from the 2008 Tampere Image Database (TID) [36] along with two distorted versions of the image. Fig. 8(b) shows the sample image corrupted with spatially correlated noise. In Fig. 8(c) the sample image is corrupted with quantization noise. In the original image, the size of the centered impulse region  $D$  is equal to 25. As shown in Fig. 8, the addition of noise reduces the value of  $D$  to 23 for the spatially correlated noise image. The value of  $D$  is reduced to 19 in the quantization noise image.

The decrease in the size of the centered impulse region is noticed with other types of noise used in the TID database. The value of  $D$  is reduced for images contaminated with additive Gaussian noise, masked noise, impulse noise, JPEG transmission errors, JPEG 2000 transmission errors, and the non-eccentric pattern noise. These experimental results show that the value of  $D$  and possibly other characteristics of the centered impulse region are promising tools for image quality assessment.

An algorithm to compute the number of scale decompositions in adaptive curvelets was presented in this section. Next, an algorithm that optimizes the number of angular divisions per scale/quadrant pair is presented.

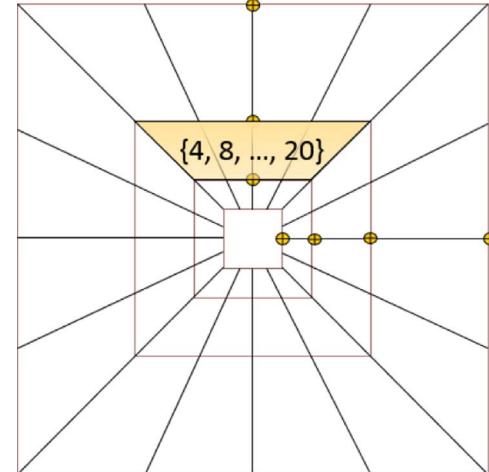
### 3.3. Optimal number of angular divisions per scale/quadrant

Optimizing the number of angular divisions is performed an exhaustive search procedure in each curvelet scale/quadrant pair (Fig. 9). The test parameters are chosen from the following sequence:  $\{4, 8, 12, 16, 20\}$ . The divisions are uniformly distributed in each scale/quadrant pair. Recalling FFTs symmetry for real data the number of parameters to optimize is  $2(J - 1)$  for real data, and is equal to  $4(J - 1)$  for the case of complex input data.

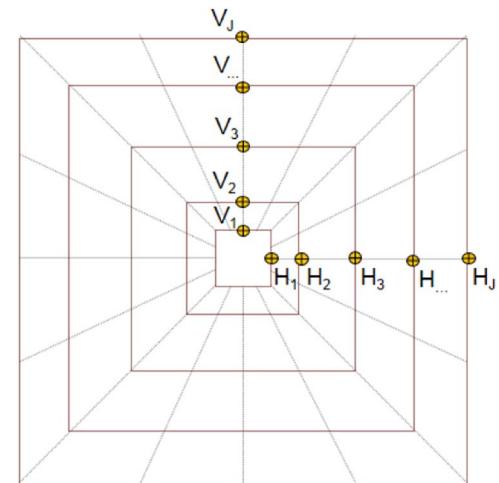
### 3.4. Optimal scale locations

We propose using derivative-free optimization methods to search for adaptive curvelet scale-locations. Such methods start at an initial point  $x_0$ , evaluate the cost function at a selected mesh of points in the neighborhood of  $x_0$ , and define the point in the mesh with the minimum function value as the new  $x_0$ . Next, the algorithm iterates until a specified convergence criterion is met. The Nelder-Mead simplex search method [37,38] is one of the popular methods for generating such a mesh. It has been used extensively in a variety of application areas. It is used in this work to find the optimal scale locations.

Let  $\{V_1, V_2, \dots, V_J\}$  be the coordinates of vertical scale locations, and



**Fig. 9.** Optimal angular decomposition is found through a brute-force search strategy for each scale/quadrant pair. The figure shows the second scale/first quadrant curvelet segment.



**Fig. 10.** Vertical and horizontal scale locations.

$\{H_1, H_2, \dots, H_J\}$  be the horizontal scale locations (Fig. 10). Curvelet scales are not constrained to be of the same length and width. This makes the number of optimizing parameters equal to  $2J$  for real data. To ensure meaningful choices for scale locations, the simplex method optimizes a constraint checking function that returns a PSNR value equal to zero whenever any of the constraints is violated. Otherwise, it returns the denoising PSNR for the requested parameters (Fig. 11).

The enforced constraints are:

1. Inner scale locations are above and to the right of the periodically extended image center:  $V_i > [V_i/2]$  and  $H_i > [H_i/2]$ .

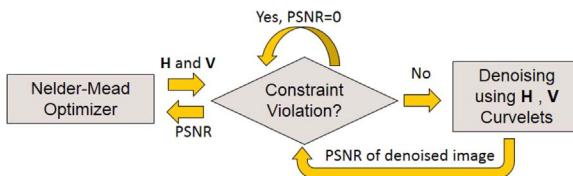


Fig. 11. Scale locations optimizer and the constraint checking function.

2. Scales are organized in an ascending order:  $V_j > V_{j-1}$  and  $H_j > H_{j-1}$ .
3. Periodic extension is used in the outer level  $J$  to enhance curvelet's performance by connecting border elements. Fig. 12 graphically illustrates the extension region. By definition, the outer border of this level needs to be outside the image boundaries:  $V_j > N_1$ , and  $H_j > N_2$ .
4. Level  $J - 1$  is constrained to lie inside the original image boundaries. This constraint is used to avoid dividing the periodic extension level by moving this level outside the image boundaries:  $V_{j-1} \leq N_1$  and  $H_{j-1} \leq N_2$ .

The algorithm iterates until it converges or until a maximum number of iterations are reached. More details about the Nelder-Mead algorithm are provided in Section 3.6.

### 3.5. Global optimization

In the previous section, three separate algorithms for finding the optimal number of scales, scale locations, and angular divisions were presented. In this section, we turn our attention to the development of a global optimization algorithm combining the previous adaptations. The algorithm uses a multi-resolution search strategy. This helps in avoiding convergence to local minima and reduces the computational cost of the algorithm. Optimization is done in hierarchical manner consisting of  $n$  iterations. In each iteration, optimal scale and angular locations are found for a downsampled smoothed version of the input image. The flow of this algorithm is illustrated in Fig. 13.

The algorithm starts with finding the optimal number of scales  $J$  using the scale selection algorithm (Section 3.1). Next, the original image is downsampled  $n - 1$  times by a factor of 1/2. The image is also smoothed by a Gaussian filter to prevent aliasing artifacts. The optimal angular decomposition is found using dyadic scale locations and the optimal number of scales computed. The angular decompositions found and the dyadic scale locations are used as initial points for the scale locations search algorithm. Next, the value of  $n$  is decremented by one.

and a new loop starts. For each step, scale locations will be initialized by the locations found in the previous iterations after being rescaled to the new image coordinates. After  $n$  iterations, once the scale locations search algorithm converges to a minimum, the angular decomposition algorithm is run to search for the optimal angular decomposition at the calculated scale locations. Finally, the algorithm terminates returning the determined optimal scale and angular decompositions.

### 3.6. Nelder–Mead's convergence and computational complexity

Convergence of the proposed method relies on the convergence of the Nelder–Mead optimization algorithm. In this section, details on this algorithm are given along with the convergence criteria. The algorithm starts by forming a simplex of size  $n + 1$ , where  $n$  is the dimension of the vector to be optimized. Vertices of this simplex will be placed at the initial point  $x_0$  and at  $n$  other points formed by increasing the value of each element of  $x_0$  by 5%. The cost function is evaluated at each of the vertices and the worst vertex is eliminated. In each iteration, the Nelder–Mead algorithm performs the following sequence of operations to determine a new vertex:

1. Compute function values at each vertex. Sort function values so that  $f(x_1) \leq f(x_2) \leq \dots \leq f(x_{n+1})$ . Assuming that a minimization problem  $x_{n+1}$  becomes the worst vertex in the simplex. Compute the center of mass for the first  $n$  points  $\bar{x} = \sum_i^n x_i/n$ .
2. Compute the function value at the reflection point  $R$ . This point is the reflection of  $x_{n+1}$  around  $\bar{x}$ . if  $f(x_1) \leq f(R) < f(x_n)$  replace vertex  $x_{n+1}$  with  $R$ . If  $f(R) < f(x_1)$  perform an expansion step and compute the function value at  $f(E)$ , where  $E$  is at the direction of  $R$  and is at twice the distance between  $R$  and the center of mass. If  $f(E) < f(R)$  replace vertex  $x_{n+1}$  with  $E$ . Otherwise, use vertex  $R$ . Finally, if  $f(R) \geq f(x_n)$  move to the next step, the reflection  $R$  and expansion  $E$  points are illustrated in Fig. 14.
3. Perform a contraction with the minimum of  $f(R)$  and  $f(x_{n+1})$ . If  $R$  was found to be the minimum, the contraction is outside the original simplex at point  $C_{\text{out}}$ . Otherwise, the contraction is within the simplex at  $C_{\text{in}}$ . The two contraction points  $C_{\text{out}}$  and  $C_{\text{in}}$  are illustrated in Fig. 14. If  $f(C_{\text{in}}) < f(x_{n+1})$  accept  $f(C_{\text{in}})$ . Otherwise, move to the next step. Similarly, if  $f(C_{\text{out}}) < f(R)$  accept  $f(C_{\text{out}})$ . Otherwise, move to the next step.
4. Shrink the simplex:  $x_1$  remains fixed. Other vertices move so that their distance to  $x_1$  is shortened by 1/2.

Nelder–Mead's implementation was set up so that convergence

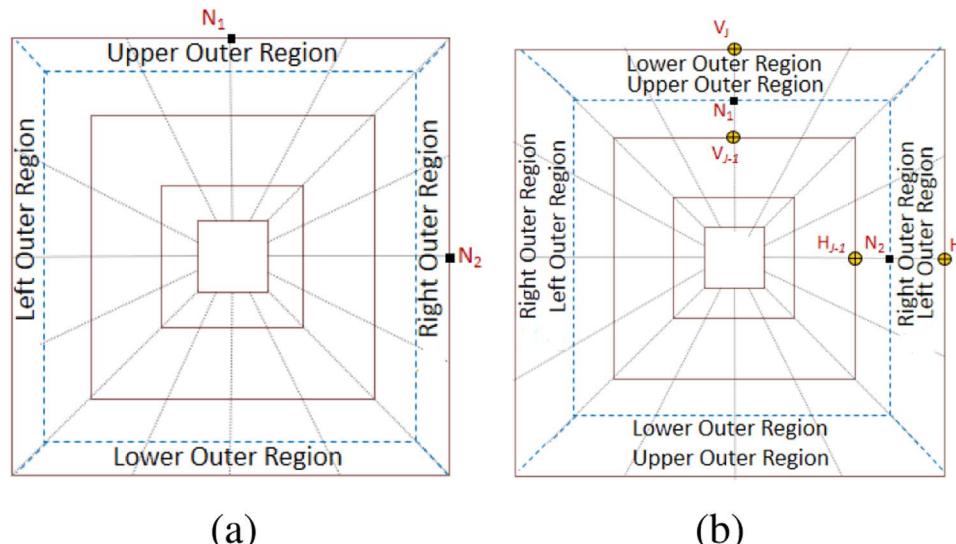


Fig. 12. Curvelet's outer periodic extension region: (a) Fourier Transform (FT) plane before adding the extension region to an image of size  $N_1, N_2$  and (b) FT plane after the extension.

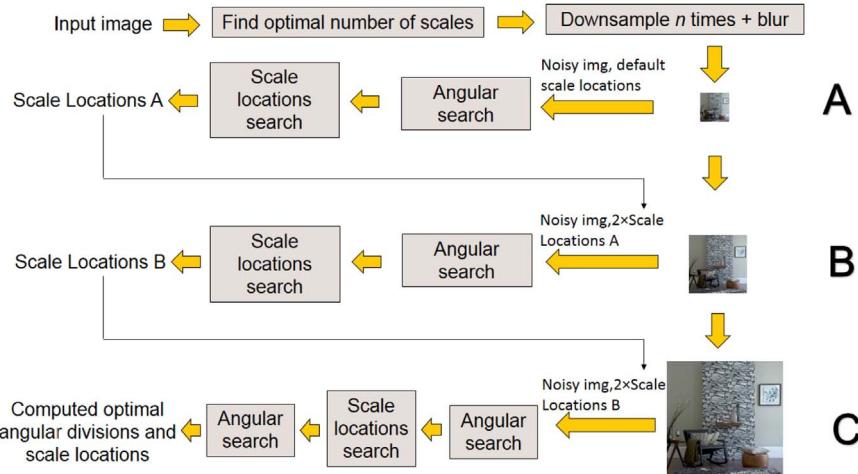


Fig. 13. Flow diagram of the global optimization algorithm.

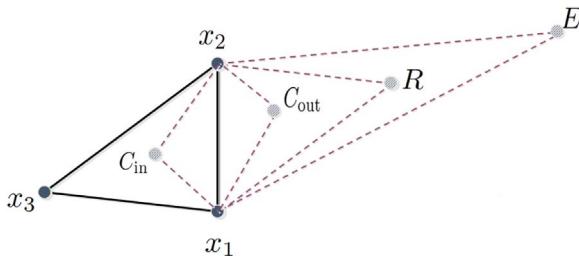


Fig. 14. Graphical illustration of the reflection, expansion, and contraction steps in the Nelder-Mead algorithm.

occurs when the maximum difference between scale locations falls below one pixel or when a maximum number of iterations are reached. The maximum number of iterations was set to 500.

Even though Nelder-Mead and other derivative free optimization algorithm lack coherent mathematical analysis, they remain popular and reliable tools that are widely used in a variety of applications. For example, derivative-free optimization algorithms such as Nelder-Mead and Powell's direction set method are frequently used in medical image registration [39–41]. Other application areas include molecular biology [42] and dynamic pricing [43].

The developed global search algorithm depends on repeated application of curvelet denoising operations. Assuming an image of size  $n \times n$ , the forward and inverse curvelet transforms have a complexity in the order of  $O(n^2 \log n)$ . The thresholding operation complexity is in the order of  $O(n^2)$ . This makes the total complexity of the developed algorithm equivalent to  $O(n^2 \log n)$ .

Actual computational times for a sample of images with sizes  $512 \times 512$  and  $256 \times 256$  are shown in Table 1. The table also shows the number of cost function evaluations. The number of cost function evaluations includes cost function calls by the Nelder-Mead algorithm and the evaluations made using the angular search algorithm. The experiments were run using an Intel i7-4500 processor operating at

**Table 1**  
Time and total number of cost function evaluations used in computing denoising-based adaptive curvelets for images Barbara, Bird, and Cameraman.

Test image	Image size	Elapsed time (min)	# of function evaluations
Barbara	$512 \times 512$	15.0	1039
Bird	$512 \times 512$	12.7	1042
Cameraman	$512 \times 512$	12.1	949
Barbara	$256 \times 256$	3.8	917
Bird	$256 \times 256$	3.1	925
Cameraman	$256 \times 256$	3.3	860

1.80 GHz with a maximum frequency of 3.00 GHz. The machine has 8.00 GB of installed memory. The search algorithm was built using MATLAB's (2014b) implementation of Nelder-Mead.

#### 4. Training curvelets

In certain applications, access to the original data is not available. For example, the dataset could be subsampled or corrupted by occlusions or corrupted by noise. In such cases, curvelet tiles can be learned from a group of images representing the specialized class to which the dataset of interest belongs. Examples of these classes include medical images, synthetic aperture radar (SAR) data, and seismic datasets. Fig. 15 illustrates an example of two classes of images and their unique fingerprints in the 2D Fourier transform plane. Fourier activity in face images is distributed around the center of the Fourier plane. While in seismic data, the activity is clustered into two groups separated by a narrow horizontal gap. Training curvelets to learn tiles that are representative of the class of interest will result in improved data representations.

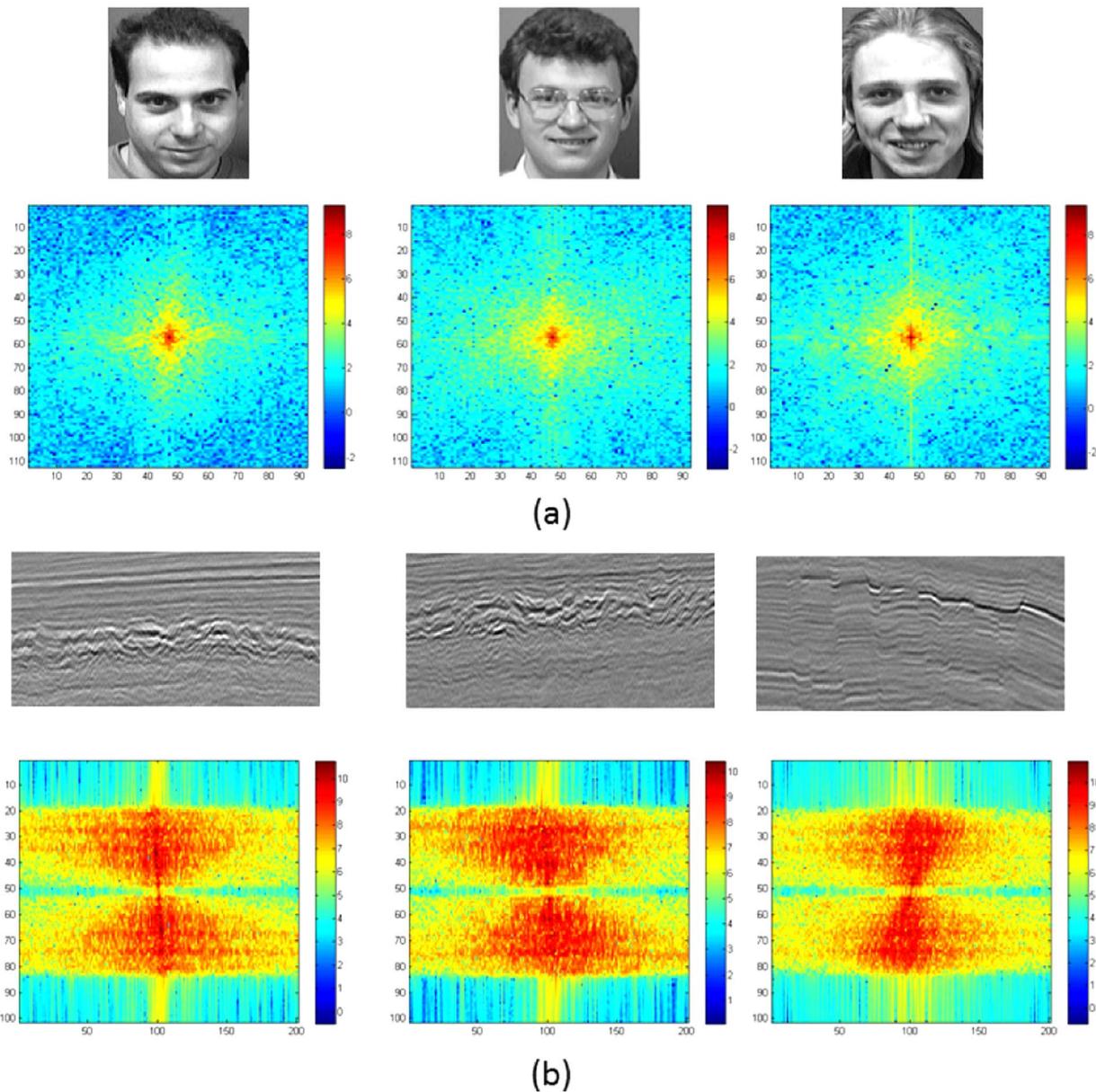
In what follows, an algorithm to learn a curvelet representation that better represents a class of images is presented. First, the scale selection algorithm is used to find the number of scale decompositions in each image in the training dataset. The average over all images is rounded to the nearest integer and is used as the number of scale decompositions. Next, a search for the optimal tiling is run for each image in the training dataset using the averaged number of scale decompositions. The average scale locations and angular decompositions, rounded to the nearest integers, are considered as the tiling representative of the class of interest. In our experiments, the average angular decomposition was obtained by rounding to the nearest multiple of four since the implementation of adaptive curvelets developed admits only multiples of four as possible values for the number of angular decomposition elements. The training algorithm is illustrated in Fig. 16.

The training procedure described in this section is not necessary for data compression and feature extractions purposes. In such applications, learning the improved tiling can be performed using the image of interest. An example of using adaptive curvelets for feature extraction was presented in [44]. In the next section adaptive curvelets partial reconstruction error is computed without using a training step.

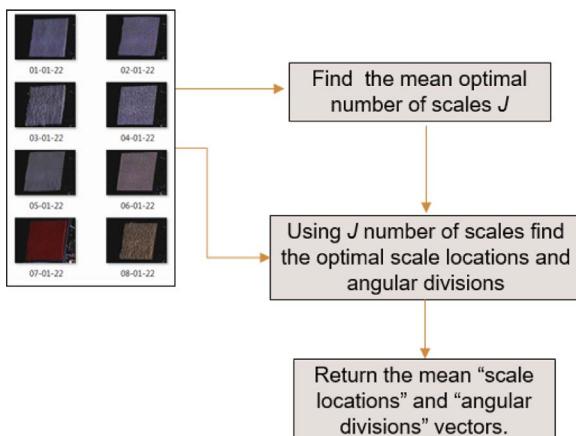
#### 5. Experimental results

##### 5.1. Coefficient decay and partial reconstruction error

In this section, we present a sparsity analysis of adaptive curvelet coefficients. Adaptive curvelet tiles were learned using AWGN with



**Fig. 15.** (a) Face images and their corresponding Fourier log-magnitudes plot. (b) Seismic images and their corresponding Fourier log-magnitude plots.

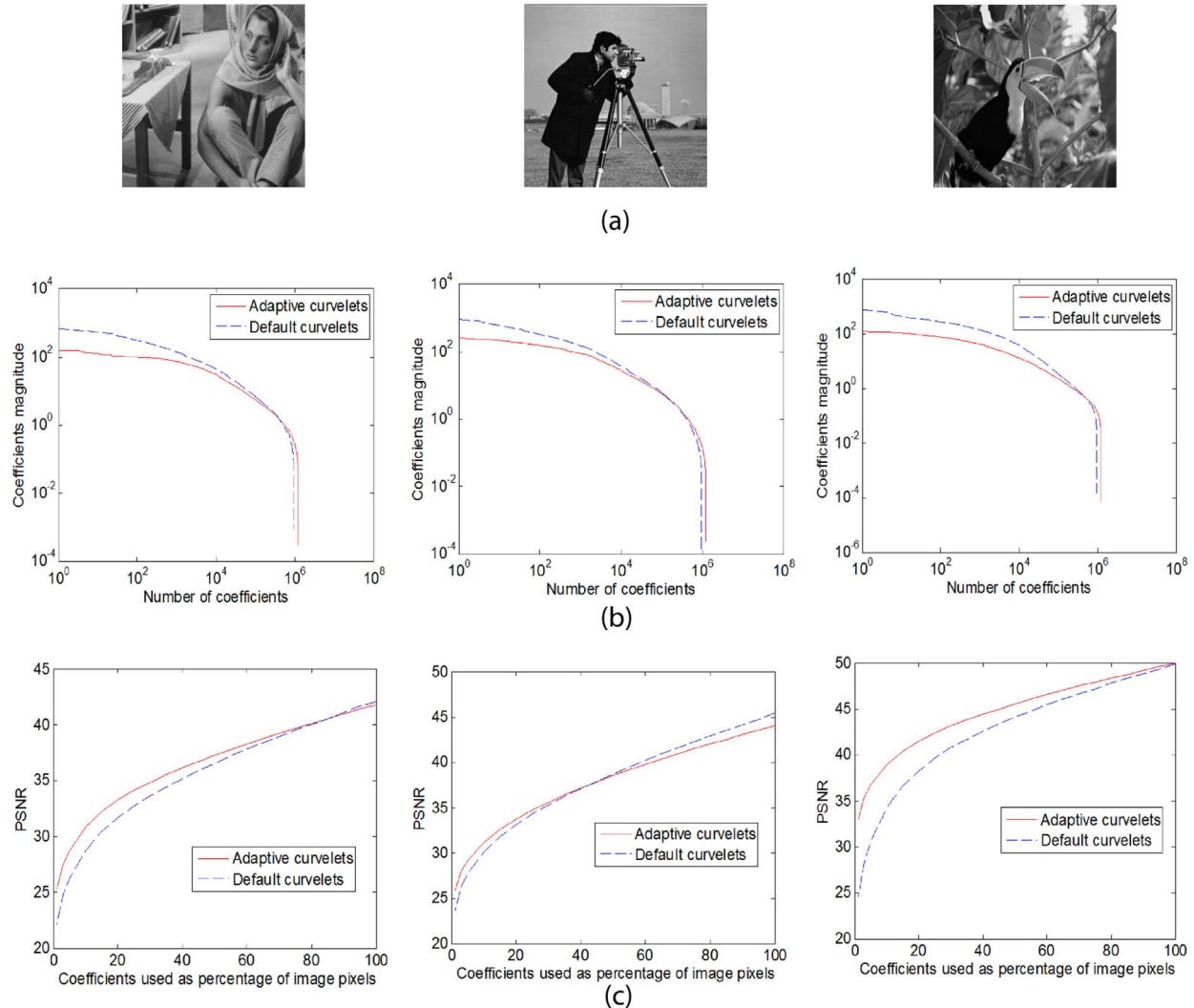


**Fig. 16.** Flow diagram of the algorithm used for training curvelets to better represent a given set of images. The training algorithm is needed when the original data is not available (Section 4).

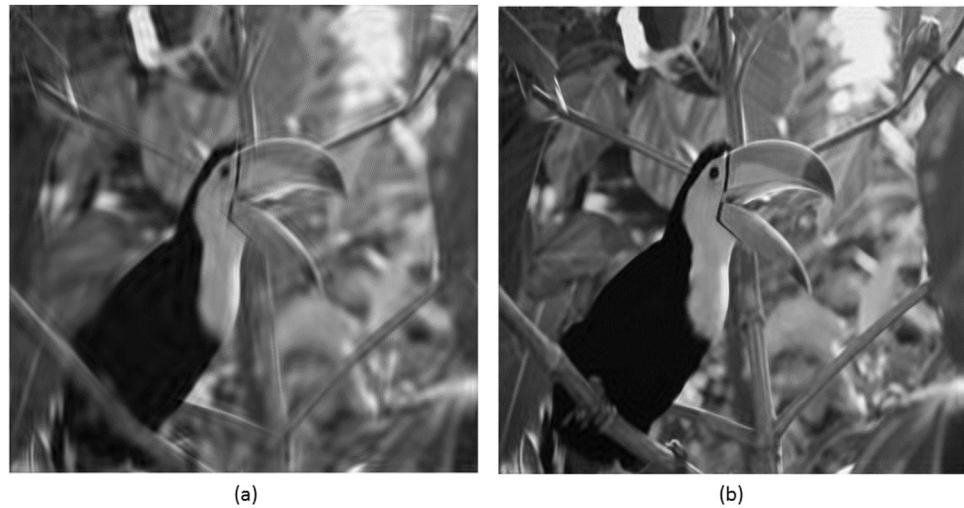
$\sigma = 25$ . Magnitude plots of curvelet and adaptive curvelet coefficients for three different images of size  $512 \times 512$  are shown in Fig. 17(b). Adaptive curvelets are shown to reduce the magnitudes of curvelets coefficients at the expense of generating a higher number of them. Comparisons between default and adaptive curvelets partial reconstruction error are presented in Fig. 17(c). The figure plots the PSNR of images reconstructed using the highest magnitude curvelet coefficients. The number of curvelet coefficients used is shown as a percentage of the number of image pixels. Adaptive curvelet coefficients show improved performance in mid- and low-quality compression regions.

A visual example illustrating the performance of the compact representation generated by adaptive curvelets is shown in Fig. 18. Two images are formed by thresholding default and adaptive curvelet coefficients. The threshold was set to select the highest 7864 coefficients. This number represents 3% of the total number of image pixels. The image formed using adaptive curvelet has significant improvements in image quality over the default curvelet image. A similar performance improvement was observed at other low- and mid-quality compression ratios.

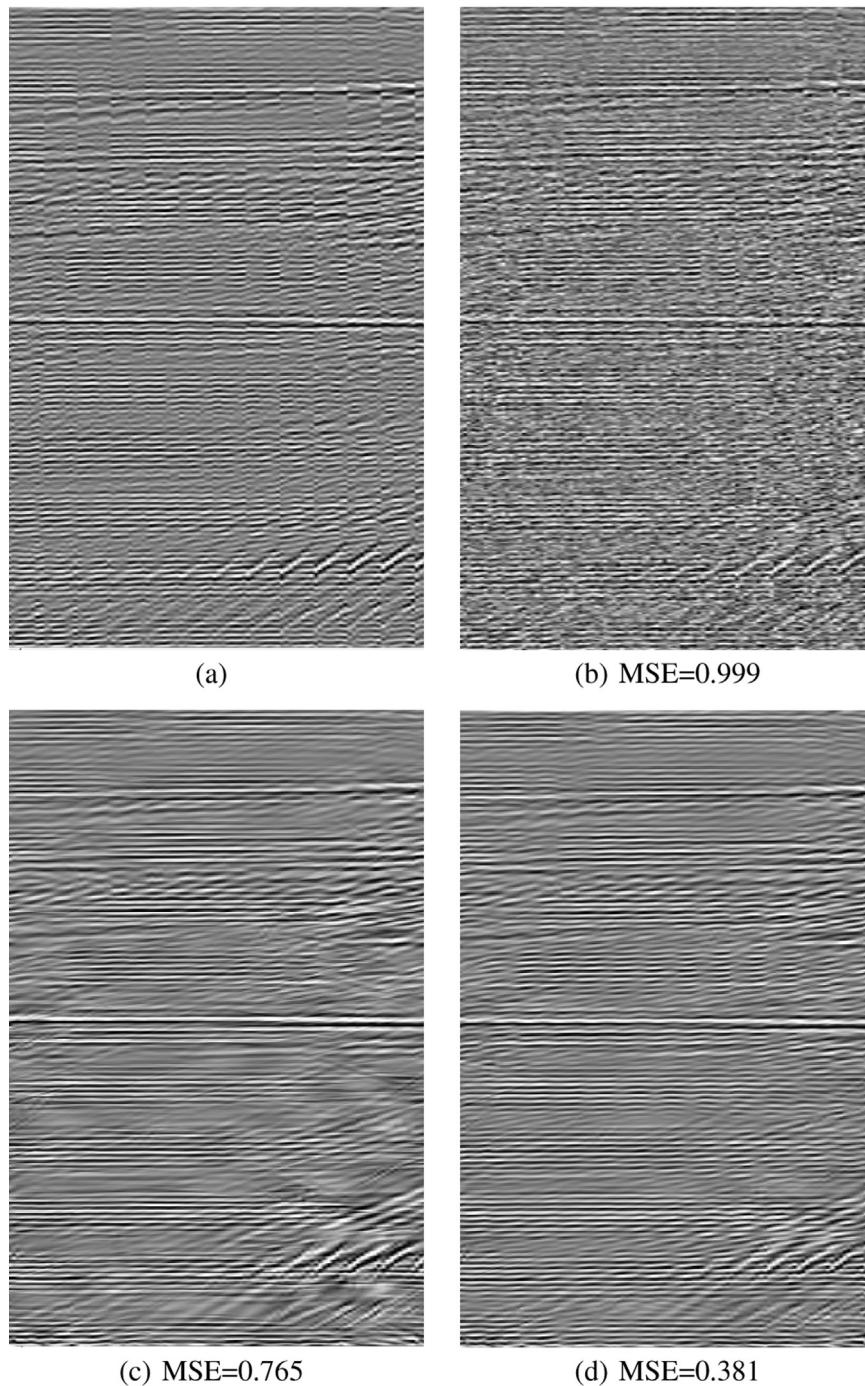
Default curvelets are reported in the literature to generate improved



**Fig. 17.** (a) Images Barbara, Cameraman, and Bird. (b) Default and adaptive curvelet decay curves shown on a log–log scale. (c) Partial reconstruction error from the highest curvelet coefficients. The number of coefficients used is shown as a percentage of the number of image pixels  $512 \times 512$ .



**Fig. 18.** Partial reconstruction of the Bird image using (a) default curvelets ( $\text{PSNR} = 28.32$ ) and (b) adaptive curvelets ( $\text{PSNR} = 35.32$ ). The threshold was set to select 7864 coefficient which is equivalent to 3% of the total number of image pixels  $512 \times 512$ .



**Fig. 19.** Denoising results: (a) original image, (b) noisy images, (c) curvelet denoising and (d) adaptive curvelet denoising.

partial reconstruction performance over wavelets [1,45]. The results shown in this section indicate that adaptive curvelets are promising candidates for the development of competitive image compression systems.

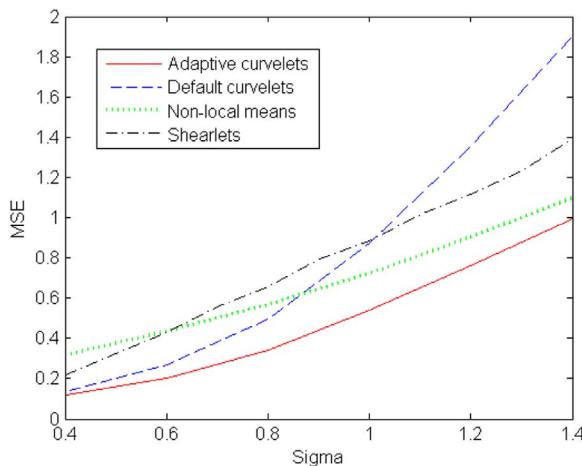
## 5.2. Denoising seismic data

The developed algorithm was tested on a seismic dataset. Training data used were  $K = 25$  collections of seismic traces. Each collection is of size  $500 \times 100$  pixels. The curvelet transform is optimized using this training set. The training set was contaminated with AWGN with  $\sigma = 1$ . The cost function used is denoising performance as measured by the logarithm of mean squared error (MSE). The number of scale decompositions  $J$  is found to be 4 for all the images. The optimal scale

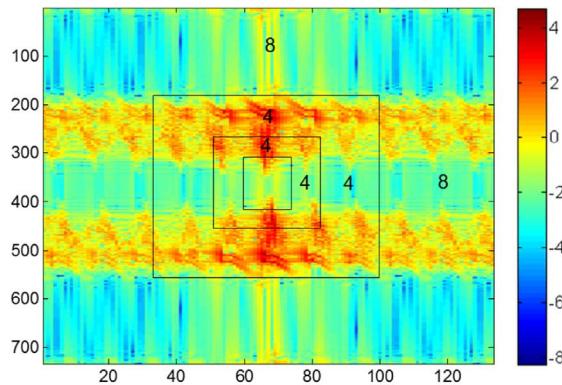
locations are found by rounding the average  $K$  optimal results to the nearest integer. Similarly, optimal angular divisions were found by averaging the  $K$  optimal divisions and rounding to the nearest multiple of four. Optimal tilings were applied to the testing data and denoising results were compared with the original curvelet algorithm. Results show considerable quantitative and visual improvement as shown in Fig. 19. Comparison with the popular non-Local means (NLM) filter [46] and the recently proposed compactly supported shearlets [4] over varying  $\sigma$  values is shown in Fig. 20.

Training results were obtained using  $\sigma = 1$ . Varying the value of  $\sigma$  used in training did not affect scale locations and angular decompositions of the optimal tiling. NLM results were obtained using the author's recommended parameters [47].

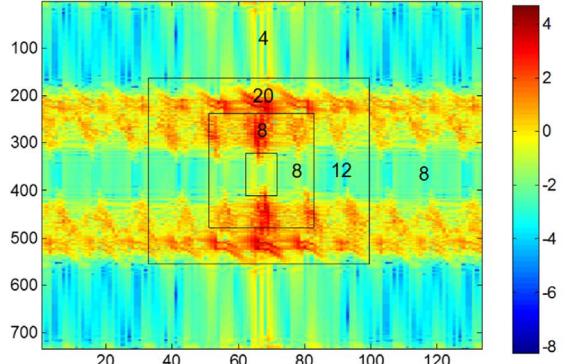
In Fig. 21, default and adaptive scale locations are plotted. The



**Fig. 20.** Seismic denoising results for default curvelets, adaptive curvelets, and NLM using various values of  $\sigma$ .



(a) Default curvelet scale locations and angular distribution

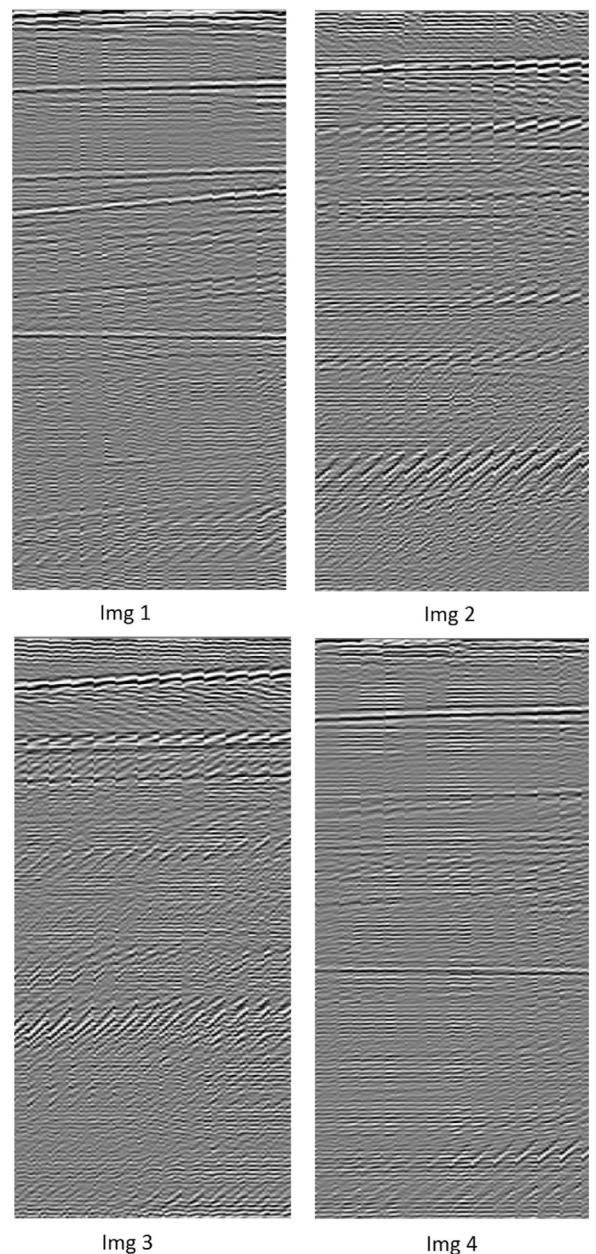


(b) Adaptive curvelet scale locations angular distribution

**Fig. 21.** Comparison between default and adaptive curvelet scale locations and angular distributions. Scale locations are plotted on a log-magnitudes plot of the Fourier transform of a sample seismic image. The numbers inside each scale/quadrant pair indicate the number of angular divisions used.

background of these images include a log-magnitude plot of the Fourier transform of a sample seismic image. Overall, adaptive curvelet scale locations are more uniformly distributed across the Fourier plane. The distance between Scale 0 (i.e. inner most scale) and Scale 1 is roughly equal to the distance between Scale 1 and Scale 2. The inner most scale is also reduced in size. The reduction in the size of inner most non-directional region increases the directionality of the transform.

Adaptive curvelets increased the number of angular divisions used in all scales. The overall number of angular decompositions in adaptive



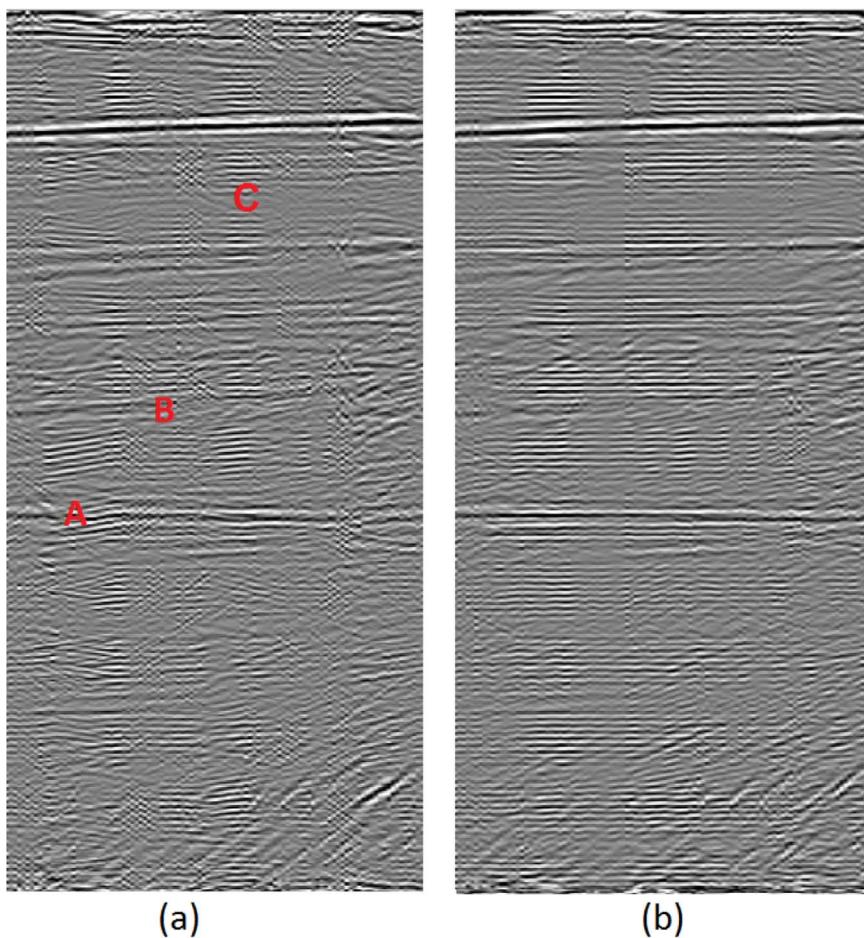
**Fig. 22.** Four seismic images used in sparsity based reconstruction experiments.

**Table 2**

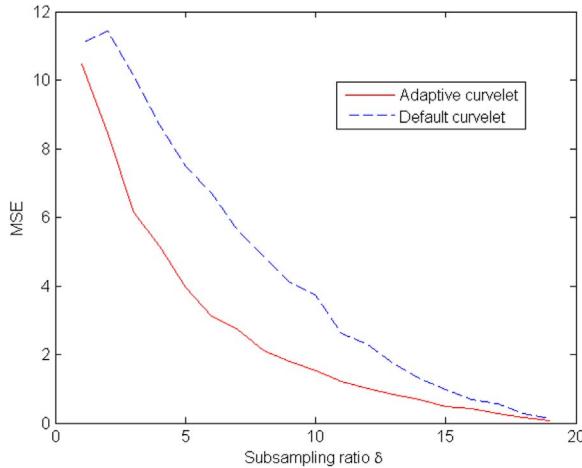
Sparse recovery results shown in MSE for four seismic datasets using a sampling ratio  $\delta = 0.3$ .

Test image	Default curvelets	Adaptive curvelets
Img1	4.22	2.11
Img2	5.31	3.71
Img3	7.62	6.11
Img4	7.80	5.10

curvelets is 60, while in default curvelets the number is 40 divisions. The increase seems related to the signal's Fourier transform activity. Close to 50% of the total number of angular divisions are used to describe Scale 2. This is a sign of the relative importance of the Fourier activity in this region. Fourier activity in this region is likely to play an important factor in the appearance of the overall image. The number of angular decompositions in Scale 1 is increased from eight in default curvelets to 16. This increase indicates a need to increase the



**Fig. 23.** Img4 reconstructed from subsampled measurements  $\delta = 0.3$  using (a) default curvelets  $MSE = 3.68$  and (b) adaptive curvelets  $MSE = 1.62$ . Letters A, B, and C highlight samples of regions where adaptive curvelets preserve the continuity of seismic horizons.

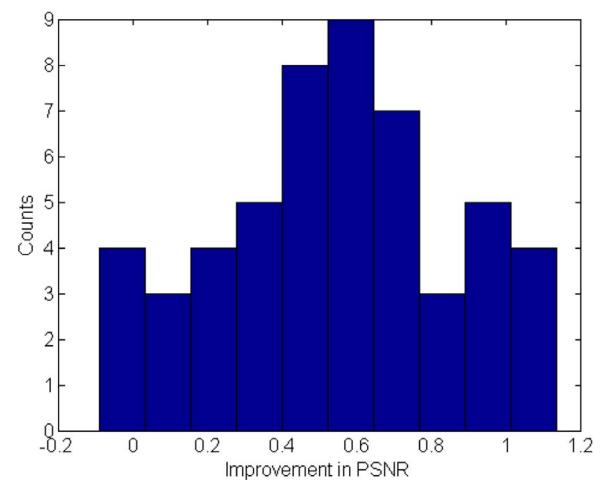


**Fig. 24.** Img4 recovery results across different subsampling ratios. Each data point is obtained using the average of five different trials.

directionality of Scale 1 over what is proposed by default curvelets. The number of angular decompositions in the outer most scale decreased. This later decrease can be related to the marginal level of Fourier transform activity in this scale.

### 5.3. CS recovery of subsampled data

Denoising-based adaptive curvelets provide a better representation of image features. The improved representation can be used in

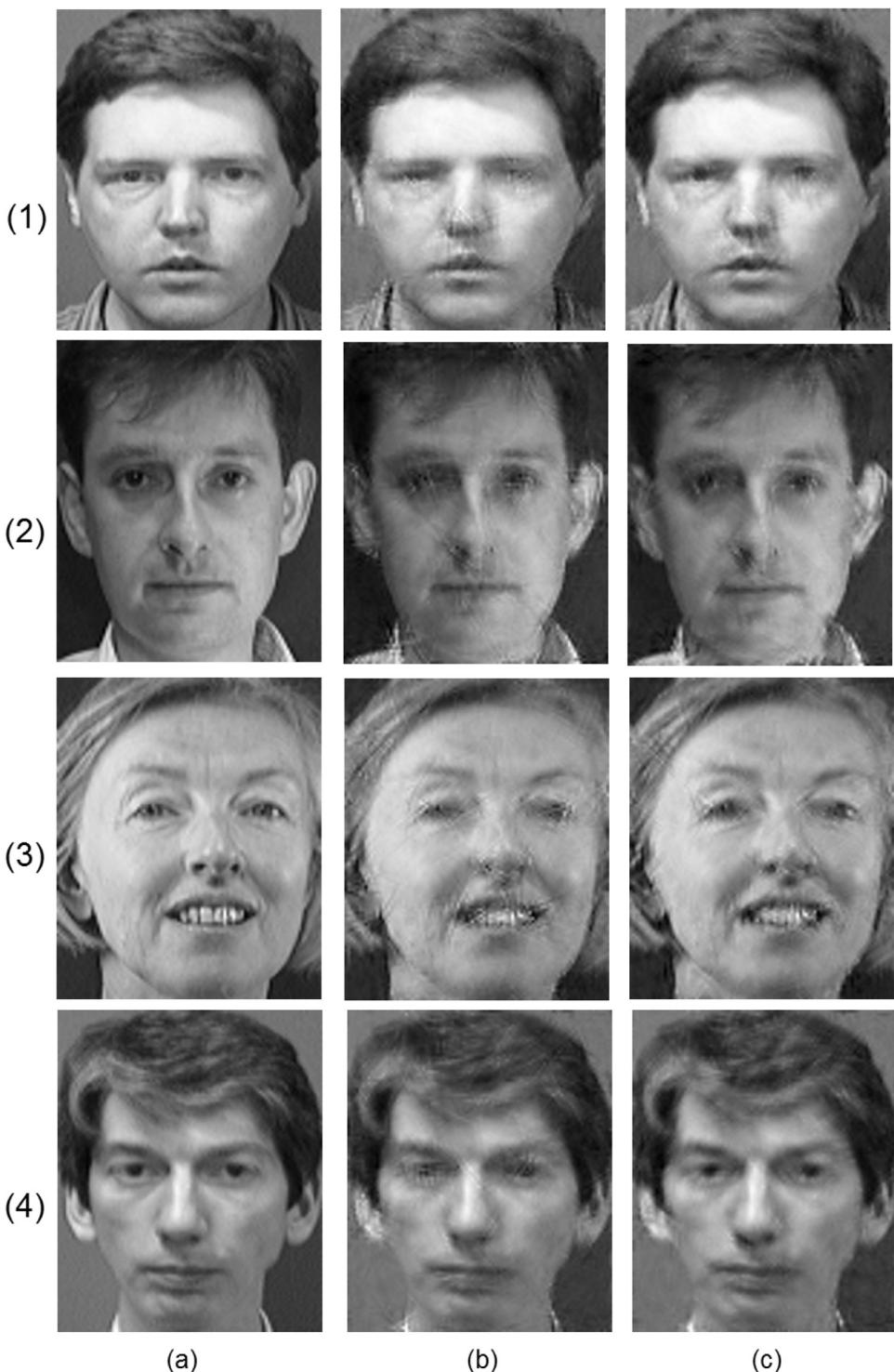


**Fig. 25.** Histogram plot of adaptive curvelet compressed sensing recovery improvements (PSNR) over default curvelets.

applications other than denoising. In [44], denoising-based adaptive curvelets were used successfully in solving texture retrieval problems. In this section, we present examples showing the advantages of using adaptive curvelets in the sparse recovery of subsampled data. Two applications for such recovery are presented in this section. This is followed by experiments on the sparse reconstruction of face images.

#### 5.3.1. Seismic data acquisition

Seismic reflection is an essential method for exploring subsurface



**Fig. 26.** Visual comparison of sparse recovery performance ( $\delta = 0.4$ ) between default and adaptive curvelets: (a) original image, (b) recovery using default curvelets and (c) recovery using adaptive curvelets.

layers of Earth. It plays an important role in the hydrocarbon industry, where it is used to locate and analyze hydrocarbon reservoirs. Acquisition of seismic data starts by generating seismic waves using a controlled active source of energy such as a dynamite explosion. The generated seismic waves interact with the subsurface layers of the Earth and generate reflections that are recorded using geophones at the surface. Each recorded wave is known as a trace. The recorded wave is processed using a structured collection of signal processing and mathematical tools to generate a map representing subsurface structure in the region of interest.

Sparsity was used in [16,45] to recover missing seismic traces from a subset of traces. The default curvelet transform was used as the sparsifying transform. Curvelet-based data recovery was shown to outperform wavelet and Fourier based recoveries. In what follows, improved recovery performance by using the adaptive curvelet transform is reported.

The problem of seismic data recovery from subsampled measurements can be mathematically described as follows. Assume that the data collected are in the form of vector  $\mathbf{b} = \mathbf{Af} \in R^n$ , where  $\mathbf{f} \in R^N$  is the vector of complete data points and  $\mathbf{A}$  is the sampling matrix that

randomly samples  $\delta = \frac{n}{N}$  seismic traces. Let  $\mathbf{x}$  be the unknown curvelet coefficients vector representing  $\mathbf{f}$ . The vector of complete measurements can be approximated by solving the following program:

$$\hat{\mathbf{f}} = C^{-1}(\hat{\mathbf{x}}) \quad \text{with } \hat{\mathbf{x}} = \arg_{\mathbf{x}} \min \| \mathbf{x} \|_1 \quad \text{subject to } \mathbf{A}C^{-1}(\mathbf{x}) = \mathbf{b}, \quad (11)$$

where  $C^{-1}(\mathbf{x})$  is the inverse curvelet transform of  $\mathbf{x}$ . It reshapes the 1D vector  $\mathbf{x}$  into a standard curvelet coefficients format. Next, it takes the inverse curvelet transform and reshapes the constructed image into a 1D vector. Numerous algorithms that solve the above optimization problem are available. We used the Spectral Projected Gradient algorithm (SPGL1) [16, 48] for its efficiency and to facilitate comparisons with previously published results.

The operation of SPGL1 requires functions that resemble the sampling operation  $AC^{-1}(\mathbf{x})$  and its adjoint. The sampling matrix  $\mathbf{A}$  can be very large in size. For example, the sampling matrix for a  $512 \times 512$  image is of size  $n \times 512^2$ , where  $n$  is the number of sampled traces. This issue is bypassed by introducing sampling functions that represents the operation of  $AC^{-1}(\mathbf{x})$  in an algorithmic manner.

The sampling operation  $AC^{-1}(\mathbf{x})$  takes the 1D vector  $\mathbf{x}$  as a vector of data points representing the curvelet transform of the reconstructed image. It works by taking the inverse curvelet transform of  $\mathbf{x}$ , after transforming it into a regular curvelet representation (where the coefficients are organized to represent curvelet tiles). Next, the image is reshaped into a 1D vector and  $n$  traces are sampled. The adjoint operation takes the vector representing the sampled data points  $\mathbf{b}$  and returns vector  $\mathbf{x}$  the curvelet representation of the reconstructed image. It works by reversing the sampling operation. Values of zero were inserted for missing seismic traces. Vectors  $\mathbf{b}$  and  $\mathbf{x}$  keep changing until convergence of the algorithm.

Eq. (11) was solved using default curvelets and adaptive curvelets. Adaptive curvelet tiles were learned using the procedure described earlier in Section 4, where 25 seismic datasets of size  $500 \times 100$  were contaminated with AWGN to find the optimal adaptive curvelet tiles. Reconstruction performance, using a sampling ratio  $\delta = 0.3$ , is demonstrated using four different seismic datasets and is shown in Fig. 22. Table 2 presents the mean squared error (MSE) values between original and reconstructed images obtained using default and adaptive curvelets. Adaptive curvelets succeed in generating recovery improvements in all cases.

Img4 reconstructions using default and adaptive curvelets are shown in Fig. 23. The subsampling ratio was set to  $\delta = 0.3$ . Adaptive curvelets results are smoother and with fewer distortions and discontinuities.

A comparison between adaptive and default curvelet recovery performance for different values of  $\delta$  is shown in Fig. 24. The figure was generated using Img4. The figure shows adaptive curvelets outperforming default curvelets over all subsampling ratios.

### 5.3.2. Reconstruction of face images

In this section, adaptive curvelets are used in the sparse recovery of photographic images from subsampled measurements. The AT & T face image database [49] was used in image recovery experiments. Adaptive curvelet tiles were learned using the procedure described earlier in Section 4. A total of 200 images were used as a training dataset. The set includes face images of 20 subjects taken at 10 different settings (e.g. different poses, open or closed eyes). All images were of size  $112 \times 92$ . Adaptive curvelet tiles were learned using a noise standard deviation  $\sigma$  value equal to 15. Recovery results were demonstrated using a testing set of 52 images. These images were not part of the training dataset.

Adaptive curvelets improved the recovery performance of default curvelets. Using a subsampling ratio  $\delta = 0.4$ , the average improvement in PSNR is 0.54 dB. A histogram plot of the improvements achieved by using adaptive curvelets is shown in Fig. 25. The improvements achieved by adaptive curvelets are established in the figure, however these improvements occupy a wide range with peaks covering the

**Table 3**

Sparse recovery results in PSNR for the face images shown in Fig. 26,  $\delta = 0.4$ .

Test image	Default curvelets	Adaptive curvelets
Face image (1)	29.25	<b>30.00</b>
Face image (2)	27.99	<b>28.40</b>
Face image (3)	25.92	<b>26.54</b>
Face image (4)	31.41	<b>31.89</b>

interval from  $-0.1$  to  $1.1$  dB. This behavior is likely caused by the non-uniformity of the dataset used. The performance of adaptive curvelets can be further improved by training over face images exhibiting a higher degree of similarity. For example, the training and testing datasets can be selected to be from the same face setting in the AT & T database. Images of subjects wearing sunglasses or with beards can be trained separately. Four sample images along with recovery results are shown in Fig. 26. Sample images reconstructed using adaptive curvelets are higher in quality and show a significant reduction of the artifacts generated in images reconstructed using default curvelets. For example, first subject eyes, nose, and lips are cleaner. A similar behavior is also observed with the third subject. Reconstruction performance in PSNR for these images are shown in Table 3.

## 6. Conclusions

A learning-based algorithm for adapting curvelet tiles was introduced in this paper. It involves choosing an optimal number of decomposition levels, angular divisions per scale/quadrant, and scale locations. Optimal choices were computed using denoising performance as a cost function. The tiling that generates the best separation between image structure and noise is considered optimal. Experimental results were performed on sets of synthetic, seismic and face images. Application areas included image denoising and partial reconstruction error. Furthermore, the problem of recovering data samples from incomplete measurements was solved using the adaptive approach in a compressed sensing framework. Our results indicate consistent performance improvements over default curvelets. In the future, we plan to extend the approach used in this paper to other wavelet and wavelet-like transforms. Determining optimal curvelet domain tilings using individual image content remains an open problem. Initial steps in this direction indicate the feasibility of maximizing the coefficient of variation ( $C_v$ ), where  $C_v = \frac{\sigma}{\mu}$ , in finding improved curvelet representations for seismic data [35].

Following the success of default curvelets in retrieving textured images [19,20], we plan to use adaptive curvelets in solving texture-retrieval problems. Popular texture retrieval algorithms rely on the similarity of frequency domain representations of reference and test images that contain the same texture. Initial results in [44] indicate promising performance.

An interesting research problem is the design of general free-form transforms where the frequency domain activity of the signal is broken down into different segments with minimal structural and transform dependent parameters.

## References

- [1] E. Candes, L. Demanet, D. Donoho, L. Ying, Fast discrete curvelet transforms, *Multiscale Model. Simul.* 5 (3) (2006) 861–899.
- [2] M.N. Do, M. Vetterli, The contourlet transform: an efficient directional multi-resolution image representation, *IEEE Trans. Image Process.* 14 (12) (2005) 2091–2106.
- [3] K. Guo, D. Labate, Optimally sparse multidimensional representation using shearlets, *SIAM J. Math. Anal.* 39 (1) (2007) 298–318.
- [4] Gitta Kutyniok, Wang-Q. Lim, Rafael Reisenhofer, Shearlab 3d: faithful digital shearlet transforms based on compactly supported shearlets, *ACM Trans. Math. Softw. (TOMS)* 42 (1) (2016) 5.
- [5] K. Engan, S.O. Aase, J. Hakon Husoy, Method of optimal directions for frame design, in: Proceedings of 1999 IEEE International Conference on Acoustics, Speech,

- and Signal Processing, 1999, vol. 5, pp. 2443–2446.
- [6] M. Aharon, M. Elad, A. Bruckstein, K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation, *IEEE Trans. Signal Process.* 54 (November (11)) (2006) 4311–4322.
  - [7] Ronald R. Coifman, M. Victor Wickerhauser, Entropy-based algorithms for best basis selection, *IEEE Trans. Inf. Theory* 38 (2) (1992) 713–718.
  - [8] Kannan Ramchandran, Martin Vetterli, Best wavelet packet bases in a rate-distortion sense, *IEEE Trans. Image Process.* 2 (2) (1993) 160–175.
  - [9] Yue M. Lu, Minh N. Do, Multidimensional directional filter banks and surfacelets, *IEEE Trans. Image Process.* 16 (4) (2007) 918–931.
  - [10] M. Unser, N. Chenouard, D. Van De Ville, Steerable pyramids and tight wavelet frames in, *IEEE Trans. Image Process.* 20 (October (10)) (2011) 2705–2721.
  - [11] Jacques Laurent, Duval Laurent, Chaux Caroline, Peyré Gabriel, A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity, *Signal Process.* 91 (12) (2011) 2699–2730.
  - [12] J.L. Starck, E.J. Candès, D.L. Donoho, The curvelet transform for image denoising, *IEEE Trans. Image Process.* 11 (6) (2002) 670–684.
  - [13] M.O. Ulfarsson, J.R. Sveinsson, J.A. Benediktsson, Speckle reduction of SAR images in the curvelet domain, in: 2002 IEEE International Geoscience and Remote Sensing Symposium, 2002. IGARSS'02, IEEE, Toronto, Canada, June 24–28, vol. 1, pp. 315–317.
  - [14] M. Choi, R.Y. Kim, M.R. Nam, H.O. Kim, Fusion of multispectral and panchromatic satellite images using the curvelet transform, *IEEE Geosci. Remote Sens. Lett.* 2 (2) (2005) 136–140.
  - [15] M.S. Miri, A. Mahloojifar, Retinal image analysis using curvelet transform and multistructure elements morphology by reconstruction, *IEEE Trans. Biomed. Eng.* 58 (5) (2011) 1183–1192.
  - [16] F.J. Herrmann, G. Hennenfent, Non-parametric seismic data recovery with curvelet frames, *Geophys. J. Int.* 173 (1) (2008) 233–248.
  - [17] S. Elaiwat, M. Bennamoun, F. Boussaid, A. El-Sallam, 3D face recognition using curvelet local features, *IEEE Signal Process. Lett.* 99 (2013) 1.
  - [18] T. Mandal, Q.M. Jonathan Wu, Y. Yuan, Curvelet based face recognition via dimension reduction, *Signal Process.* 89 (12) (2009) 2345–2353.
  - [19] Ishrat Jahan Sumana, Md Monirul Islam, Dengsheng Zhang, Guojun Lu, Content based image retrieval using curvelet transform, in: 2008 IEEE 10th Workshop on Multimedia Signal Processing, IEEE, Cairns, Australia, pp. 11–16.
  - [20] F. Gómez, E. Romero, Rotation invariant texture characterization using a curvelet based descriptor, *Pattern Recognit. Lett.* 32 (16) (2011) 2178–2186.
  - [21] Gerlind Plonka, Stefanie Tenorth, Daniela Rosca, A new hybrid method for image approximation using the easy path wavelet transform, *IEEE Trans. Image Process.* 20 (2) (2011) 372–381.
  - [22] Chang Chuo-Ling, B. Girod, Direction-adaptive discrete wavelet transform for image compression, *IEEE Trans. Image Process.* 16 (May (5)) (2007) 1289–1302.
  - [23] Pierre Vandergheynst, Pascal Frossard, Efficient image representation by anisotropic refinement in matching pursuit, in: Proceedings of 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001 (ICASSP'01), vol. 3, IEEE, Salt Lake City, Utah, USA, pp. 1757–1760.
  - [24] B. Ophir, M. Lustig, M. Elad, Multi-scale dictionary learning using wavelets, *IEEE J. Sel. Top. Signal Process.* 5 (September (5)) (2011) 1014–1024.
  - [25] Felix J. Herrmann, Deli Wang, Dirk J. Verschuur, Adaptive curvelet-domain primary-multiple separation, *Geophysics* 73 (3) (2008) A17–A21.
  - [26] Xiang Wu, Barry Hung, High-fidelity adaptive curvelet domain primary-multiple separation, *First Break* 33 (1) (2015) 53–59.
  - [27] Preeti D. Swami, Alok Jain, Image denoising by supervised adaptive fusion of decomposed images restored using wave atom, curvelet and wavelet transform, *Signal, Image Video Process.* 8 (3) (2014) 443–459.
  - [28] Hasan Al-Marzouqi, Ghassan AlRegib, Curvelet transform with adaptive tiling, in: IS & T/SPIE Electronic Imaging, International Society for Optics and Photonics, Burlingame, California, United States, p. 82950F.
  - [29] Hasan Al-Marzouqi, Ghassan AlRegib, Searching for the optimal curvelet tiling, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, Vancouver, BC, Canada, pp. 1626–1630.
  - [30] Stéphane Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1999.
  - [31] Emmanuel J. Candès, Michael B. Wakin, An introduction to compressive sampling, *IEEE Signal Process. Mag.* 25 (2) (2008) 21–30.
  - [32] Michael Wakin, Jason Laska, Marco Duarte, Dror Baron, Shriram Sarvotham, Dharmpal Takhar, Kevin Kelly, Richard G. Baraniuk, Compressive imaging for video representation and coding, in: Picture Coding Symposium, Beijing, China, vol. 1, 2006.
  - [33] Bruno A. Olshausen, David J. Field, Sparse coding of sensory inputs, *Curr. Opin. Neurobiol.* 14 (4) (2004) 481–487.
  - [34] Niall Hurley, Scott Rickard, Comparing measures of sparsity, *IEEE Trans. Inf. Theory* 55 (10) (2009) 4723–4741.
  - [35] Hasan Al-Marzouqi, Ghassan AlRegib, Using the coefficient of variation to improve the sparsity of seismic data, in: 1st IEEE Global Conference on Signal and Information Processing, 2013.
  - [36] Nikolay Ponomarenko, Federica Battisti, Karen Egiazarian, Jaakko Astola, Vladimir Lukin, Metrics performance comparison for color image database, in: Fourth International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, Arizona, USA, vol. 27, 14–16 January 2009.
  - [37] John A. Nelder, Roger Mead, A simplex method for function minimization, *Comput. J.* 7 (4) (1965) 308–313.
  - [38] Andrew R. Conn, Katya Scheinberg, Luis N. Vicente, *Introduction to Derivative-Free Optimization*, vol. 8, SIAM, 2009.
  - [39] Francisco P.M. Oliveira, João Manuel R.S. Tavares, Medical image registration: a review, *Comput. Methods Biomed. Eng.* 17 (2) (2014) 73–93.
  - [40] Lin Xu, Justin W.L. Wan, Tianjian Bian, A continuous method for reducing interpolation artifacts in mutual information-based rigid image registration, *IEEE Trans. Image Process.* 22 (8) (2013) 2995–3007.
  - [41] Hua-me Chen, Pramod K. Varshney, Mutual information-based ct-mr brain image registration using generalized partial volume joint histogram estimation, *IEEE Trans. Med. Imaging* 22 (9) (2003) 1111–1119.
  - [42] Thomas Gregor, William Bialek, Rob R. de Ruyter van Steveninck, David W. Tank, Eric F. Wieschaus, Diffusion and scaling during early embryonic pattern formation, *Proc. Natl. Acad. Sci. U.S.A.* 102 (51) (2005) 18403–18407.
  - [43] Tatiana Levina, Yuri Levin, Jeff McGill, Mikhail Nediak, Dynamic pricing with online learning and strategic consumers: an application of the aggregating algorithm, *Oper. Res.* 57 (2) (2009) 327–341. <http://dx.doi.org/10.1287/opre.1080.0577>.
  - [44] H. Al-Marzouqi, G. AlRegib, Texture similarity using periodically extended and adaptive curvelets, in: The IEEE Global Conference on Signal and Information Processing (GlobalSIP), 2014.
  - [45] J. Herrmann, Felix, Randomized sampling and sparsity: getting more information from fewer samples, *Geophysics* 75 (6) (2010) WB173–WB187.
  - [46] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005, vol. 2, pp. 60–65.
  - [47] Antoni Buades, Bartomeu Coll, Jean-Michel Morel, Non-local means denoising, *Image Process. On Line* 1 (2011).
  - [48] E. van den Berg, M.P. Friedlander, SPGL1: A Solver for Large-scale Sparse Reconstruction, Online: <<http://www.cs.ubc.ca/labs/scl/spgl1>>, 2007.
  - [49] F.S. Samaria, A.C. Harter, Parameterisation of a stochastic model for human face identification, in: Proceedings of the Second IEEE Workshop on Applications of Computer Vision, 1994, pp. 138–142.