Curvelets and Wave Atoms for Mirror-Extended Images

Laurent Demanet† and Lexing Ying‡

† Department of Mathematics, Stanford University, Stanford CA94305
‡ Department of Mathematics, University of Texas at Austin, Austin, TX 78712

July 2007

Abstract

We present variants of both the digital curvelet transform, and the digital wave atom transform, which handle the image boundaries by mirror extension. Previous versions of these transforms treated image boundaries by periodization. The main ideas of the modifications are 1) to tile the discrete cosine domain instead of the discrete Fourier domain, and 2) to adequately reorganize the in-tile data. In their shift-invariant versions, the new constructions come with no penalty on the redundancy or computational complexity. For shift-variant wave atoms, the penalty is a factor 2 instead of the naive factor 4.

These various modifications have been included in the Curvelab and WaveAtom toolboxes, and extend the range of applicability of curvelets (good for edges and bandlimited wavefronts) and wave atoms (good for oscillatory patterns and textures) to situations where periodization at the boundaries is uncalled for. The new variants are dubbed ME-curvelets and ME-wave atoms, where ME stands for mirror-extended.

1 Introduction

Curvelets [4, 5, 3] and wave atoms [9, 29] are two multiscale geometric transforms that have revealed themselves quite useful over the past few years, in such diverse fields as image processing [12, 13, 22, 27], seismic imaging [17, 18, 19], numerical analysis [8, 10], and analysis of partial differential equations [1, 2]. Many useful alternative architectures have also been investigated in the literature, and go by the names of contourlets [11], bandlets [25], shearlets [16], brushlets [24], dual-tree wavelets and wavelet packets [6, 14, 21, 30], multiselective wavelets [20], etc.

The definition of discrete curvelets and wave atoms, as in the toolboxes Curvelab [3] and WaveAtom [9], makes a central use of the discrete Fourier transform. This choice has one side effect: the FFT implicitly treats the image on $[0,1]^2$ as periodic across its boundaries. As a result, the basis functions that are localized near the image edges wrap around to the opposite edge by periodicity. This behavior may pose a problem for image processing applications where the contrast difference at opposite edges is unphysical and should play no role in the sparsity of a transform expansion.

One typical remedy is to work on the mirror-extended image, defined on the square $[-1,1]^2$ as

$$\tilde{u}(x_1, x_2) = u(|x_1|, |x_2|). \tag{1}$$

Applying the curvelet or wave atom algorithm naively to the extended image will result in (at least) a fourfold increase in computational complexity and redundancy. The overhead is easily identified: it is due to redundant translation steps in the mirrored portions of the image.

The main contribution of this paper is that there is, in most cases, a clean way of avoiding this unwanted redundancy by working directly in the *discrete cosine* domain of the original image.

This paper is organized as follows. In Section 2, we rehearse the construction of curvelets and wave atoms following [3, 9]. In Section 3, we introduce curvelets with mirror extension at the image boundaries. In Section 4, we generalize the results to wave atoms. We conclude by showing numerical experiments involving the new transforms. Our convention for the continuous Fourier transform in $\mathbb{R}^d$ is

$$\hat{f}(\boldsymbol{\omega}) = \int e^{-i\mathbf{x}\cdot\boldsymbol{\omega}} f(\mathbf{x})\, d\mathbf{x}, \qquad f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int e^{i\mathbf{x}\cdot\boldsymbol{\omega}} \hat{f}(\boldsymbol{\omega})\, d\boldsymbol{\omega}.$$

2 Basic transform architectures

2.1 Curvelets

This section is a summary of the exposition of discrete curvelets in [3]. In this paper, we focus on the "wrapping" version of the curvelet transform and we let $\mathbf{x} \in \mathbb{R}^2$.

In continuous frequency $\boldsymbol{\omega}$, and with a continuous translation parameter $\mathbf{b}$, the definition of curvelet coefficients of data $f(\mathbf{x})$ is

$$c_{j,\ell,\mathbf{b}} = \int \hat{f}(\boldsymbol{\omega}) U_{j,\theta_\ell}(\boldsymbol{\omega}) e^{i\mathbf{b}\cdot\boldsymbol{\omega}}\, d\boldsymbol{\omega}, \tag{2}$$

where U_{j,θ_ℓ} is a real-valued wedge window dilated to scale j and parabolically localized to angles near θ_ℓ by a shearing operation. To give an idea, when $j > 3$ and $|\omega_2| < .8 \cdot \omega_1$, say, we can write

$$U_{j,\theta_\ell}(\boldsymbol{\omega}) = W(2^{-j}\boldsymbol{\omega}) V\left(2^{\lfloor j/2 \rfloor}\omega_2/\omega_1 - \ell\right), \qquad \tan\theta_\ell = \ell \cdot 2^{-\lfloor j/2 \rfloor}, \tag{3}$$

where V and W are fixed—see [3] for details. The collection of U_{j,θ_ℓ} gives rise to the well-known SDD tiling of the frequency plane as in Figure 1.

An application of Plancherel's formula shows that $c_{j,\ell,\mathbf{b}}$ is exactly the expected inner product of $f(\mathbf{x})$ with curvelets at different scales, angles and positions:

$$c_{j,\ell,\mathbf{b}} = \int f(\mathbf{x}) \overline{\varphi}_{j,\ell}(\mathbf{x} - \mathbf{b})\, d\mathbf{x}, \tag{4}$$

where $\varphi_{j,\ell}(\mathbf{x})$ is the inverse Fourier transform of $U_{j,\theta_\ell}(\boldsymbol{\omega})$.

To obtain curvelets with discrete parameters, $\mathbf{b}$ appearing in equation (2) should be sampled so that the resulting collection of curvelets is still a tight frame:

$$f(\mathbf{x}) = \sum_{j,\ell,\mathbf{k}} \langle f, \varphi_{j,\ell,\mathbf{k}} \rangle \varphi_{j,\ell,\mathbf{k}}(\mathbf{x}), \qquad (\text{conv. in } L^2).$$

One possible choice when $-\pi/4 \le \theta_\ell \le \pi/4$, corresponding to the "wrapping" variant of the transform, is

$$\mathbf{b}_\mathbf{k} = \left(\delta_1 k_1 \cdot 2^{-j}, \delta_2 k_2 \cdot 2^{-\lfloor j/2 \rfloor}\right), \tag{5}$$

with adequate constants δ_1, δ_2. Observe how the parabolic scaling enters the construction of the window U_{j,θ_ℓ} in (3), and by duality the sampling of $\mathbf{b}$ in (5). In what follows, we may abbreviate $\mu = (j, \ell, \mathbf{k})$.

To obtain a fully discrete curvelet transform, the frequency variable $\boldsymbol{\omega}$ should be discretized by equispaced samples and limited to a square $[-\pi N, \pi N]^2$ corresponding to the bandlimit of the

data. As a result, the FFT can be invoked to compute Fourier transforms. In turn, (2) can be seen as a small inverse FFT, after an operation of reassignment by periodization called "wrapping" has been applied to the windowed data $\hat{f}(\boldsymbol{\omega})U_{j,\theta_\ell}(\boldsymbol{\omega})$. The algorithm for the fast discrete curvelet transform (FDCT) can then be summarized as follows:

1. Apply the 2D FFT and obtain Fourier samples $\hat{f}[\omega_1, \omega_2]$, where $\omega_1, \omega_2 \in 2\pi\mathbb{Z}$ and obey $-\pi N \leq \omega_1, \omega_2 < \pi N$.
2. For each scale j and angle ℓ, form the product $\tilde{U}_{j,\theta_\ell}[\omega_1, \omega_2]\hat{f}[\omega_1, \omega_2]$.
3. Wrap this product around the origin and obtain
$$\tilde{f}_{j,\ell}[\omega_1, \omega_2] = W(\tilde{U}_{j,\ell}\hat{f})[\omega_1, \omega_2].$$
4. Apply the inverse 2D FFT to each $\tilde{f}_{j,\ell}$, hence collecting the discrete coefficients $c^D(j, \ell, k)$.

This algorithm for the FDCT has computational complexity $O(N^2 \log N)$ and in practice, its computational cost does not exceed that of 6 to 10 two-dimensional FFTs. Because the FDCT preserves the ℓ_2 norm, it is invertible by the adjoint transform in essentially the same complexity.

Let us finish this short review by listing a few properties of the FDCT that are sometimes overlooked.

- Curvelets are complex-valued, but a real-valued construction exists as well [3].
- The transform has reasonable redundancy, about 7.5 if curvelets are used at the finest scale, or 2.5 otherwise.
- Curvelets are by construction *shift-invariant*, in the sense that *subspaces* corresponding to fixed j, ℓ are invariant under integer-pixel translations of the image $f(\mathbf{x})$. Individually, coefficients are of course not shift-invariant, and not even nearly so.
- Discrete curvelets are as faithful to the continuous transform (2) as the joint sampling and bandlimiting of the data allows (Proposition 6.1 in [3]).

2.2 Wave Atoms

The unabridged reference for this section is [9]. We apologize to the reader for overloading such common symbols as j, k in the current and previous sections.

Individually, wave atoms are tensor products of a special type of 1D wave packets due to Lars Villemoes [29]. In continuous frequency, we start by defining

$$\hat{\psi}^0_m(\omega) = e^{-i\omega/2}\left[e^{i\alpha_m} g(\epsilon_m(\omega - \pi(m + 1/2))) + e^{-i\alpha_m} g(\epsilon_{m+1}(\omega + \pi(m + 1/2)))\right], \qquad (6)$$

with $\alpha_m = \frac{\pi}{2}(m + \frac{1}{2})$ and $\epsilon_m = (-1)^m$. The function g is a particular real-valued, compactly-supported C^∞ bump function chosen such that

$$\sum_m |\hat{\psi}^0_m(\omega)|^2 = 1,$$

and such that the translates $\{\psi_m(t - k)\}$ form an orthonormal basis of $L^2(\mathbb{R})$. This construction provides a *uniform*, or Gabor, tiling of the frequency axis.

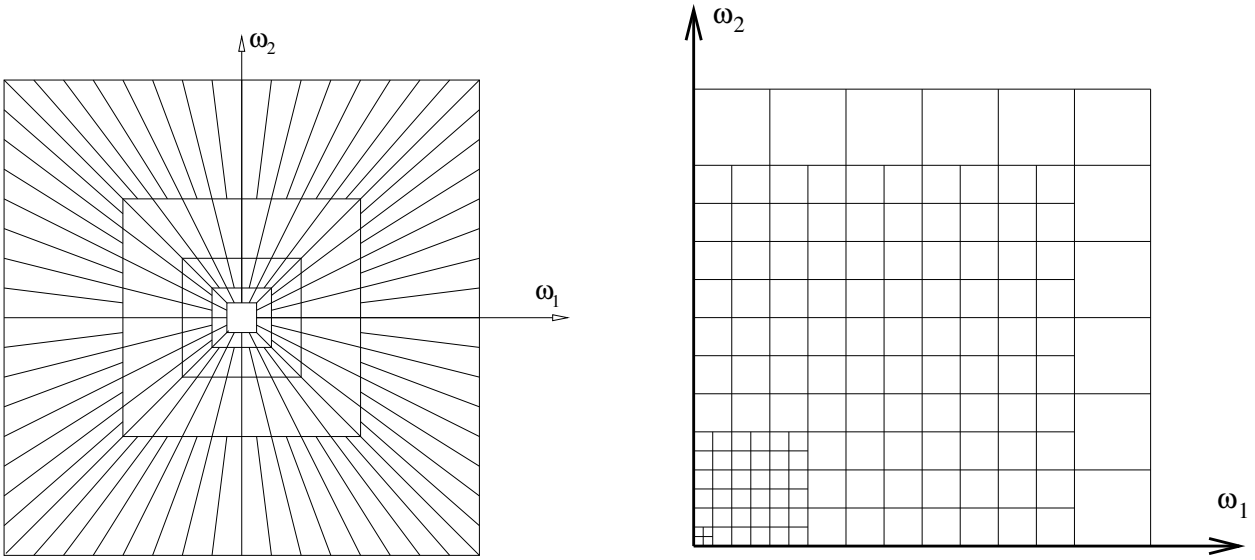

Figure 1: Left: Curvelet tiling, also called second dyadic decomposition (SDD) of the frequency plane. Right: wave atom tiling of the frequency plane — only the first quadrant is shown.

The novelty in Villemoes's construction is that dilations of the template ψ^0_m can generate non-Gabor, wavelet-packet-like orthobases with very good frequency localization [29]. Let us introduce a subscript j to index scale, and define

$$\psi^j_{m,n}(x) = \psi^j_m(x - 2^{-j}n) = 2^{j/2}\psi^0_m(2^j x - n).$$

The coefficients associated to $\psi^j_{m,n}(x)$ will naturally be denoted as $c_{j,m,n}$. Note that the couple (j, m) refers to a point on the wavelet packet tree; the depth at that point is $J - j$, where J is the maximum depth, and m can be interpreted as the number of nodes on the left at the same depth (nodes are not necessarily leaves). This convention is the standard indexing scheme for wavelet packets.

Within the limits of the Fourier uncertainty principle, a given atom $\psi^j_{m,n}(x)$ is centered in the (x, ω) space at

$$x_{j,n} = 2^{-j}n, \qquad \omega_{j,m} = \pi 2^j m.$$

The "wave atom" architecture is obtained when we let the subscript m be comparable to 2^j in absolute value. This choice implies a parabolic scaling law: the window width in frequency ($2^j 2\pi$) is asymptotically, up to a constant, the square root of the distance to the origin ($\sim 2^{2j}$). In a dual fashion, the essential width of the support in x ($\sim 2^{-j}$) is about the square root of the wavelength inside the envelope ($\sim 2^{-2j}$).

A second orthonormal basis is obtained by Hilbert-transforming the basis elements to define

$$H\hat{\psi}^0_m(\omega) = e^{-i\omega/2}\left[(-i)e^{i\alpha_m} g(\epsilon_m(\omega - \pi(m + 1/2))) + ie^{-i\alpha_m} g(\epsilon_{m+1}(\omega + \pi(m + 1/2)))\right], \qquad (7)$$

and adequate dilations thereof for other values of j. The coefficients in the Hilbert-transformed basis will be denoted by $c^H_{j,m,n}$.

There are essentially three ways to combine 1D wave atoms in order to define 2D wave atoms:

1. *An orthonormal basis.* Orthonormal basis functions with 4 bumps in the frequency plane can be formed by individually taking tensor products of 1D wave atoms in a "nonstandard"

fashion — where nonstandard refers to the fact that there is only one dilation parameter j. So we let $\mu = (j, \mathbf{m}, \mathbf{k})$, where $\mathbf{m} = (m_1, m_2)$ and $\mathbf{n} = (n_1, n_2)$, and write

$$\varphi^+_\mu(x_1, x_2) = \psi^j_{m_1}(x_1 - 2^{-j}n_1)\, \psi^j_{m_2}(x_2 - 2^{-j}n_2).$$

The Fourier transform is also separable, namely

$$\hat{\varphi}^+_\mu(\omega_1, \omega_2) = \hat{\psi}^j_{m_1}(\omega_1) e^{-i2^j n_1 \omega_1}\, \hat{\psi}^j_{m_2}(\omega_2)\, e^{-i2^j n_2 \omega_2}.$$

A dual orthonormal basis can be defined from the Hilbert-transformed wavelet packets,

$$\varphi^-_\mu(x_1, x_2) \equiv H\psi^j_{m_1}(x_1 - 2^{-j}n_1)\; H\psi^j_{m_2}(x_2 - 2^{-j}n_2).$$

2. *A tight frame with redundancy 2.* The main drawback of the above definition is that basis functions in the x space oscillate in two directions instead of one. This lack of directionality can be remedied in a well-known way [26] by combining the primal and dual (Hilbert-transformed) basis. More precisely, the recombination

$$\varphi^{(1)}_\mu = \frac{\varphi^+_\mu + \varphi^-_\mu}{2}, \qquad \varphi^{(2)}_\mu = \frac{\varphi^+_\mu - \varphi^-_\mu}{2}, \tag{8}$$

provides basis functions with *two* bumps in the frequency plane, symmetric with respect to the origin, hence purely directional wave atoms. Together, $\varphi^{(1)}_\mu$ and $\varphi^{(2)}_\mu$ form the wave atom frame and may be denoted jointly as φ_μ.

3. *A tight frame with redundancy four.* For data compression or numerical analysis, the orthobasis or tight frame with redundancy two should prove satisfactory. However for statistical estimation, where translation invariance matters, a different design is desirable. Shift-invariance of the subspaces for fixed j and $\mathbf{m}$ can be obtained by isolating bumps in the frequency plane by means of quadrant indicators, thereby constructing complex-valued basis functions. As a result, the phase factors in (6) are not necessary anymore, and the redundancy is quadrupled from the orthobasis construction. Details can be found in [9].

The frequency tiling of 2D wave atoms is in Figure 1 on the right. In contrast to curvelets, about $O(2^{j/2})$ wave atom windows would be needed to cover a SDD wedge at scale j.

The discretization of wave atoms closely follows the strategy of frequency sampling and wrapping used for curvelets. For 1D wave atoms, we define

$$c^D_{j,m,n} = \sum_{\omega \in 2\pi\mathbb{Z}} e^{i2^{-j}nk} \overline{\hat{\psi}^j_m(\omega)} \hat{f}(\omega). \tag{9}$$

Modulo aliasing questions that only complicate the design of basis functions at the smallest scale, we can let $\omega = 2\pi(-N/2+1 : 1 : N/2)$ in the sum above. The wrapping operation is a periodization of the frequency samples that allows to write

$$c^D_{j,m,n} = \sum_{\omega = 2\pi(-2^j/2+1:1:2^j/2)} e^{i2^{-j}n\omega} \sum_{p \in 2\pi\mathbb{Z}} \overline{\hat{\psi}^j_m(\omega + 2^j p)} \hat{u}(\omega + 2^j p).$$

The corresponding expressions in 2D are simple tensor products. The flowchart of the wave atom transform is very similar to that of curvelets and yields an $O(N \log N)$ algorithm in 1D — $O(N^2 \log N)$ in 2D. All three variants of the wave atom transform are isometries and therefore invertible in essentially the same complexity by the adjoint. All variants are implemented in the WaveAtom software package.

3 Mirror-Extended Curvelets

For the time being, let the data be a real function $u(\mathbf{x})$ of continuous $\mathbf{x} \equiv (x_1, x_2)$. (If the data are complex-valued, consider the real part and imaginary part separately.) As previously, we define its mirror extension $\tilde{u}$ by equation (1) posed in $[-1,1]^2$. The basic guideline we will follow for the construction of mirror-extended (ME) curvelets is to work on the extension $\tilde{u}$, and cut down redundant computations where possible. Let us detail the two most important points.

3.1 Cosine Instead of Fourier

The first step of the ME curvelet transform is to compute a cosine transform of u in place of the Fourier transform of $\tilde{u}$. In order to see why this is important, stick to continuous variables x_1, x_2 in the square $[-1,1]^2$ and consider the Fourier series of $\tilde{u}$:

$$\hat{\tilde{u}}(\boldsymbol{\omega}) = \int_{[-1,1]^2} e^{-i\mathbf{x}\cdot\boldsymbol{\omega}/2} \tilde{u}(\mathbf{x})\, d\mathbf{x}, \qquad \boldsymbol{\omega} \equiv (\omega_1, \omega_2) \in 2\pi\mathbb{Z} \times 2\pi\mathbb{Z}.$$

It is immediate to check that $\hat{\tilde{u}}$ is the cosine series[1] of the original function u, up to a factor 4:

$$\hat{\tilde{u}}(\omega_1, \omega_2) = 4 \int_{[0,1]^2} \cos(x_1\omega_1/2) \cos(\pi x_2\omega_2/2) u(x_1, x_2)\, dx_1 dx_2.$$

In particular, the coefficients are real-valued, and have the same even-even symmetry as $\tilde{u}$:

$$\hat{\tilde{u}}(\omega_1, \omega_2) = \hat{\tilde{u}}(|\omega_1|, |\omega_2|).$$

Due to this redundancy, the computation of $\hat{\tilde{u}}(\omega_1, \omega_2)$ should be limited to the first quadrant $\omega_1, \omega_2 > 0$, and this is precisely what the cosine series does.

In practice, x_1 and x_2 are discretized over n equispaced points, as $(x_{1j_1}, x_{2j_2}) = (j_1, j_2)/n$, $j_1, j_2 = 0, \ldots, n-1$. If we simply copy the image by two successive 1D mirror extensions like $[abcd] \to [abcddcba]$ along x_1 and x_2, then the Fourier transform of the mirrored image is unfortunately neither real nor even about the origin. To avoid this problem, another option would be to mirror the image according to $[abcd] \to [abcdcb]$, and identify the Fourier transform with the output of the type-I DCT transform. There are however issues[2] with the implementation of DCT-I. Instead, to best exploit existing fast routines[3], we discretize the cosine series by means of the type-II discrete cosine transform (DCT-II, or simply called the DCT). The definition of the DCT-II in 1D, with vector indexing starting at zero, is

$$\hat{u}[k] = w(k) \sum_{j=0}^{N-1} \cos\left[\frac{\pi}{N}\left(j + \frac{1}{2}\right) k\right] u[j], \qquad k = 0, \ldots, N-1,$$

with weights $w(k) = \sqrt{2/N}$ if $k \neq 0$, and $w(0) = 1/\sqrt{N}$. (Think of k as $\omega/(2\pi)$). The weight factors make the DCT unitary, hence the inverse DCT is simply the transpose. In 2D, the DCT is the succession of two DCT along each dimension.

[1] We could also call it the cosine transform on an interval (1D) or on a square (2D); it is the most obvious way of building an orthobasis of L^2 with cosine functions.

[2] The DCT-I is absent from Matlab's standard toolboxes, and is curiously difficult to implement efficiently. For instance, the DCT-I is present in the FFTW package [15], but the documentation warns us that it is often best computed via FFT, by going back to the mirror-extended vector. This would defeat our purpose.

[3] The DCT-II is present in the JPEG image compression standard and many good routines are available for it [15].

There exists a correspondence between DCT and FFT, but it is not as simple as in the continuous case. In short, if $[abcd]$ represents a vector of real numbers and $[ABCD]$ represent twice its DCT (computed without the weights $w(k)$), then

$$v = [0a0b0c0d0d0c0b0a] \quad \Rightarrow \quad \mathrm{FFT}(v) = [ABCD0(-D)(-C)(-B)(-A)(-B)(-C)(-D)0DCB].$$

The first element (A) is the zero in frequency. The portion of the FFT with sign flips is not interesting and corresponds to the high-frequency signature of the insertion of zeros in v. We will therefore discard this portion and deploy the wedge partitioning on the truncated Fourier domain, i.e., with sequences like $[ABCD0DCB]$ in 1D. Let us remark that this sequence is not exactly the Fourier transform of the mirrored data $[abcddcba]$ (as we remarked earlier, the latter would not be real-valued), but the substitute is a very good one.

In two dimensions, the truncation means that the frequency plane is now tiled with four mirror-flipped copies of the DCT data, instead of 16 when the sign-flipped regions were kept.

At this stage, let us recall that the algorithm has only computed the DCT of the input, in complexity $O(N^2 \log N)$.

3.2 Assignment to Symmetric Wedges

The second step of the ME curvelet transform is to window and wrap the data around the origin as before, from data in the frequency plane obtained by mirror extension of the DCT. We are naively brought back to the complexity of the curvelet transform on the mirror-extended images, with a factor-4 redundancy. The symmetries of the data $\hat{f}$ in the frequency plane, however, as well as the symmetries of the transform architecture, can be leveraged to identify these redundancies. For the data, by construction, we have

$$\hat{f}(\omega_1, \omega_2) = \hat{f}(-\omega_1, \omega_2) = \hat{f}(\omega_1, -\omega_2) = \hat{f}(-\omega_1, -\omega_2).$$

As for the wedge windows $W_{j,\theta_\ell}(\omega_1, \omega_2)$ of the curvelet partitioning with $0 < \theta_\ell < \pi/2$, we have

$$W_{j,\theta_\ell}(\omega_1, \omega_2) = W_{j,\pi-\theta_\ell}(-\omega_1, \omega_2) = W_{j,\pi+\theta_\ell}(-\omega_1, -\omega_2) = W_{j,-\theta_\ell}(\omega_1, -\omega_2).$$

(All angles are defined modulo 2π, of course.) At the level of the coefficients, denote by ℓ_{-+} the subscript corresponding to $\pi - \theta_\ell$, ℓ_{--} the subscript corresponding to $\pi + \theta_\ell$, and ℓ_{+-} the subscript corresponding to $-\theta_\ell$. Then, it is easy to see from equation (2) that

$$c_{j,\ell,b_1,b_2} = c_{j,\ell_{-+},(-b_1),b_2} = c_{j,\ell_{--},(-b_1,-b_2)} = c_{j,\ell_{+-},b_1,(-b_2)}.$$

The same symmetry relations hold for the "corner windows", as defined in [3]. As a result of this symmetry, we only need to consider angles $0 < \theta_\ell < \pi/2$, and obtain all the other ones through the relations above.

Let us explain quite concretely the impact of this symmetry on the architecture of the curvelet transform. As mentioned, the loop on angles will only cover $[0, \pi/2]$. Fix j and θ_ℓ. After wrapping the windowed data near the origin, and taking an inverse FFT, we obtain an array of numbers indexed by the position index (k_1, k_2), related to (b_1, b_2) through equation (5). The only difference with the periodic case is now that k_1 and k_2 also run over *negative* numbers, corresponding to coefficients in the mirror-extended portions of the image. Hence, the array of coefficients needs to be split in 4, and assigned to the different angles corresponding to the correspondences $\ell \to \theta_\ell$, $\ell_{-+} \to \pi - \theta_\ell$, $\ell_{--} \to \pi + \theta_\ell$, $\ell_{+-} \to -\theta_\ell$.

These are the only modification that make the ME curvelet transform different from the standard curvelet transform. Since the redundancies due to the mirroring have all been removed by construction, the complexity is essentially the same as that of regular curvelets.

4 Mirror-extended wave atoms

Wave atoms are also implemented in the frequency domain, and the basic idea of reducing computations by going to the DCT domain carries over. In particular, for the tight frame of wave atoms with redundancy four, the discussion of the previous section applies word for word and needs not be changed. This defines a redundancy-4 tight frame of *complex-valued ME wave atoms.*

There is a twist, however, for the less redundant versions of wave atoms (orthobasis and tight frame with redundancy 2). Since wave atoms are not real-valued in the frequency domain, the symmetry relations are not the same as in the case of curvelets. This difficulty is related to the fact that wave atoms are built from a critically decimated 1D orthobasis. The "delicate" phase factors of the Fourier transform of the 1D basis functions in equation (6) are essential for invertibility and tightness, as in any wavelet-like orthonormal basis.[4]

As before, the first step of the ME wave atom transform is to compute the DCT of the image.

The more complicated symmetry property of wave atoms is that coefficients centered at positions in the mirror-extended portion of the image are in fact coefficients in the dual, *Hilbert-transformed basis.* Let us explain this claim with a lemma for the 1D transform.

Lemma 1. *Let $\hat{\psi}_m^j(\omega)$ be a 1D wave atom basis function in frequency. Then*

$$\hat{\psi}_m^j(\omega) = i^m e^{-i2^{-j}\omega} \overline{H\hat{\psi}_m^j(\omega)}, \tag{10}$$

where $H\hat{\psi}_m^j(\omega)$ is the corresponding basis element in the Hilbert-transformed basis.

Proof. By definition, $\hat{\psi}_m^j(\omega) = 2^{-j/2}\hat{\psi}_m^0(2^{-j}\omega)$, so the case $j \neq 0$ in equation (10) follows directly from the case $j = 0$. Let us therefore fix $j = 0$, and consider the complex conjugate of the left-hand side, which by equation (6) equals

$$\overline{\hat{\psi}_m^0}(\omega) = e^{i\omega/2}\left[e^{-i\alpha_m} g(\epsilon_m(\omega - \pi(m+1/2))) + e^{i\alpha_m} g(\epsilon_{m+1}(\omega + \pi(m+1/2)))\right]. \tag{11}$$

This expression can be related to $H\hat{\psi}_m^j$ if and only if there exist scalars β_m such that

$$\begin{pmatrix} e^{-i\alpha_m} \\ e^{i\alpha_m} \end{pmatrix} = \beta_m \begin{pmatrix} e^{i\alpha_m} \\ -e^{-i\alpha_m} \end{pmatrix}. \tag{12}$$

We can check from inspection of the (generic) cases $m = 0, 1, 2, 3$ that this property holds if we choose $\beta_m = (-i)^m$. This is no coincidence; it is a direct consequence of the fact that $e^{-i\alpha_m}$ and $e^{i\alpha_m}$ are in quadrature (their arguments differ by $\pi/2$) and remain so after a rotation in the complex plane. (Note that this "quadrature" condition is an essential ingredient for building an orthobasis. It is also present in filter design in traditional wavelet analysis).

Obtaining equation (10) is now a matter of using (12) in (11), and comparing it with with (7). □

The corresponding symmetry relation for 1D wave atom coefficients is the following.

Proposition 1. *Let c^D denote the discrete wave atom coefficients, and c^{DH} the discrete wave atom coefficients in the Hilbert-transformed basis. Assume the frequency data $\hat{f}(\omega)$ is real and even. Then*

$$c^D_{j,m,(-n)} = i^m c^{DH}_{j,m,(n-1)}$$

[4]The construction of curvelets, for instance, does not have such phase factors. This comes with a drawback, namely a small increase in redundancy for curvelets, but also with an advantage: the lack of aliasing implies the shift-invariance property that we have alluded to earlier.

Proof. From the pre-wrapping formulation (9), we have

$$c^D_{j,m,n} = \sum_{\omega \in 2\pi\mathbb{Z}} e^{i2^{-j}n\omega} \overline{\hat{\psi}^j_m(\omega)} \hat{f}(\omega).$$

The wave atoms are not even in x, but they are real-valued hence obey $\overline{\hat{\psi}^j_m(-\omega)} = \hat{\psi}^j_m(\omega)$. Using this information, and after renaming ω to $-\omega$, we obtain for the coefficient at $j, m, (-n)$,

$$c^D_{j,m,(-n)} = \sum_{\omega \in 2\pi\mathbb{Z}} e^{i2^{-j}n\omega} \hat{\psi}^j_m(\omega) \hat{f}(\omega).$$

The only difference with the expression of $c^D_{j,m,n}$ is the complex conjugate over the basis function, which prevents a direct identification. If we invoke Lemma 1, however, we can write

$$c^D_{j,m,(-n)} = i^m \sum_{\omega \in 2\pi\mathbb{Z}} e^{i2^{-j}n\omega} e^{-i2^{-j}\omega} \overline{H\hat{\psi}^j_m(\omega)} \hat{f}(\omega),$$

which is none other than $i^m c^{DH}_{j,m,n-1}$, in the Hilbert-transformed basis.

□

The next step of the ME wave atom algorithm in 2D, after taking the DCT, is exactly the same as for the periodic wave atoms: wrapping the four bumps of the windowed data near the origin, and inverse FFT of the wrapped array. For each j and m, these operations output an array of numbers $d_{j,\mathbf{m},n_1,n_2}$ that contains four times as many entries as in the non-ME case — because n_1 and n_2 can take on negative values.

If we isolate the first quadrant ($n_1 \geq 0$, $n_2 \geq 0$), and put

$$c^{(++)}_{j,\mathbf{m},n_1,n_2} = d_{j,\mathbf{m},n_1,n_2}, \qquad n_1, n_2 \geq 0,$$

then the resulting coefficients correspond to wave packets in the non-mirrored portion of the image. We do not seek to isolate them to form a new transform: the basis functions would not be orthonormal. Instead, we accept the fourfold increase in redundancy and deal with the full arrays $d_{j,\mathbf{m},n_1,n_2}$. In this respect, the only computational relief compared to running the full wave atom orthobasis on the extended image is at the level of the initial FFT — now a DCT.

The redundancy-2 tight frame of fully-directional wave atoms, however, has an interesting mirror-extended counterpart. If we were to apply the naive transform on mirror-extended images, we would have a redundancy-8 transform. We propose to lower the redundancy to four.

By definition, the fully directional wave atom coefficients are combinations of coefficients in the standard orthobasis, with coefficients in the double-Hillbert-transformed orthobasis. This set of coefficients is also accessible from the array $d_{j,\mathbf{m},n_1,n_2}$; thanks to Proposition 1, they can be found in the third quadrant:

$$c^{(--)}_{j,\mathbf{m},n_1,n_2} = d_{j,\mathbf{m},-n_1-1,-n_2-1}, \qquad n_1, n_2 \geq 0.$$

Coefficients of fully directional basis functions can then be defined by combining $c^{(1)} = (c^{(++)} \pm c^{(--)})/2$, as before. In order to design an isometric transform (tight frame), we also need to consider the second and fourth quadrants:

$$c^{(+-)}_{j,\mathbf{m},n_1,n_2} = d_{j,\mathbf{m},n_1,-n_2-1}, \qquad n_1, n_2 \geq 0,$$

$$c^{(-+)}_{j,\mathbf{m},n_1,n_2} = d_{j,\mathbf{m},-n_1-1,n_2}, \qquad n_1, n_2 \geq 0,$$

and combine them in an analogous manner: $c^{(2)} = (c^{(+-)} \pm c^{(-+)})/2$. The collection of coefficients $c^{(1)}$ and $c^{(2)}$ for various wave vectors j and $\mathbf{m}$ defines our desired tight frame of fully directional ME wave atoms, with redundancy 4. We call them the *real-valued ME-wave atoms*, in contrast to the conceptually simpler complex-valued ME-wave atoms defined at the beginning of this section.

In conclusion, the ME wave atom algorithms yield the proper generalization of wave atoms to mirror-extended image, in complexity and redundancy

- roughly four times that of the wave atom orthobasis,
- roughly twice that of the wave atom tight frame with redundancy two, and
- roughly the same as the shift-invariant wave atom tight frame with redundancy four.

5 DST variant

Let us mention in passing that a *DST variant* of the curvelet and wave atom transforms can also be defined, where DST stands for Discrete Sine Transform. It would be a properly decimated version of either transform, defined on the mirror-extended image with parity sign changes,

$$\tilde{u}(-x_1, x_2) = -u(x_1, x_2), \qquad \tilde{u}(x_1, -x_2) = -u(x_1, x_2), \qquad \tilde{u}(-x_1, -x_2) = u(x_1, x_2).$$

As its name indicates, the transform should be computed using the Discrete Sine Transform, in a manner entirely analogous to what was done in the DCT case.

This DST variant may be useful in numerical analysis when treating Dirichlet boundary conditions on a rectangle, i.e., when the solution of an equation is zero at the boundary. Similarly, the DCT-based ME transforms that we defined in this paper should be useful for encoding Neumann boundary condition on a rectange, i.e., when the normal derivative of the solution is zero at the boundary.

6 Numerical experiments

In Figure 2 we show curvelets and wave atoms near the edge of the image: instead of being extended by periodicity near the opposite edge, they are mirror-extended inside the image. This new feature may positively affect the sparsity of images which have high contrast discontinuities across periodized edges; we verify that sparsity is indeed enhanced by the ME transforms for the simple image

$$f(x_1, x_2) = x_1 + x_2 - 1, \qquad 0 \le x_1, x_2 \le 1.$$

We discretize f by sampling on a 512-by-512 grid. In this experiment, we plot the error of the best m-term approximant for the curvelet, ME-curvelet, wave atom and ME-wave atom transforms, measured in PSNR (dB). In Figure 3 we can see that inversion from the largest 128 ME-curvelet coefficients suffices to accurately recover the function (PSNR = 50.0dB), while the largest 128 regular curvelet coefficients synthesize a very poor approximant (PSNR = 9.8dB, this is not a typo). In Figure 4 the same experiment is carried out for wave atoms: the best 128-term ME-wave atom approximant gives PSNR = 47.7dB, while the best 128-term wave atom approximant gives PSNR = 24.7dB. The wave atom variant used for this experiment is the tight frame with redundancy 2.

Finally, our research is reproducible:

`http://www.curvelet.org`
`http://www.waveatom.org`

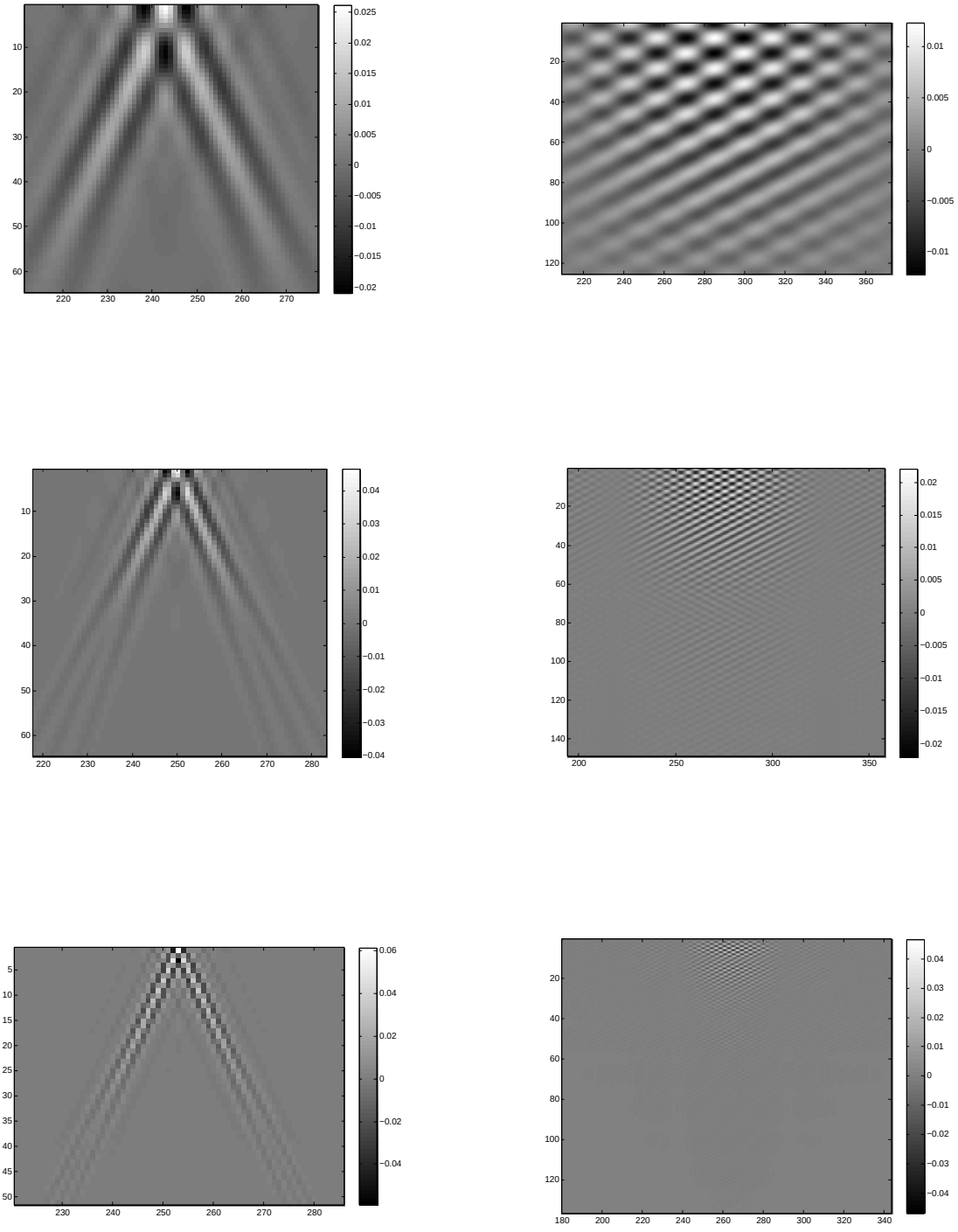

Figure 2: Left row: Zoom of ME curvelets at different scales, near the boundary (the top of the image). Right row: Zoom of ME wave atoms at different scales, near the boundary. ME curvelets or wave atoms centered in the interior of the image are very similar to their counterpart in the standard periodic implementations. The experienced reader will notice that the spatial decay is as good for the ME transforms as for the regular transforms.

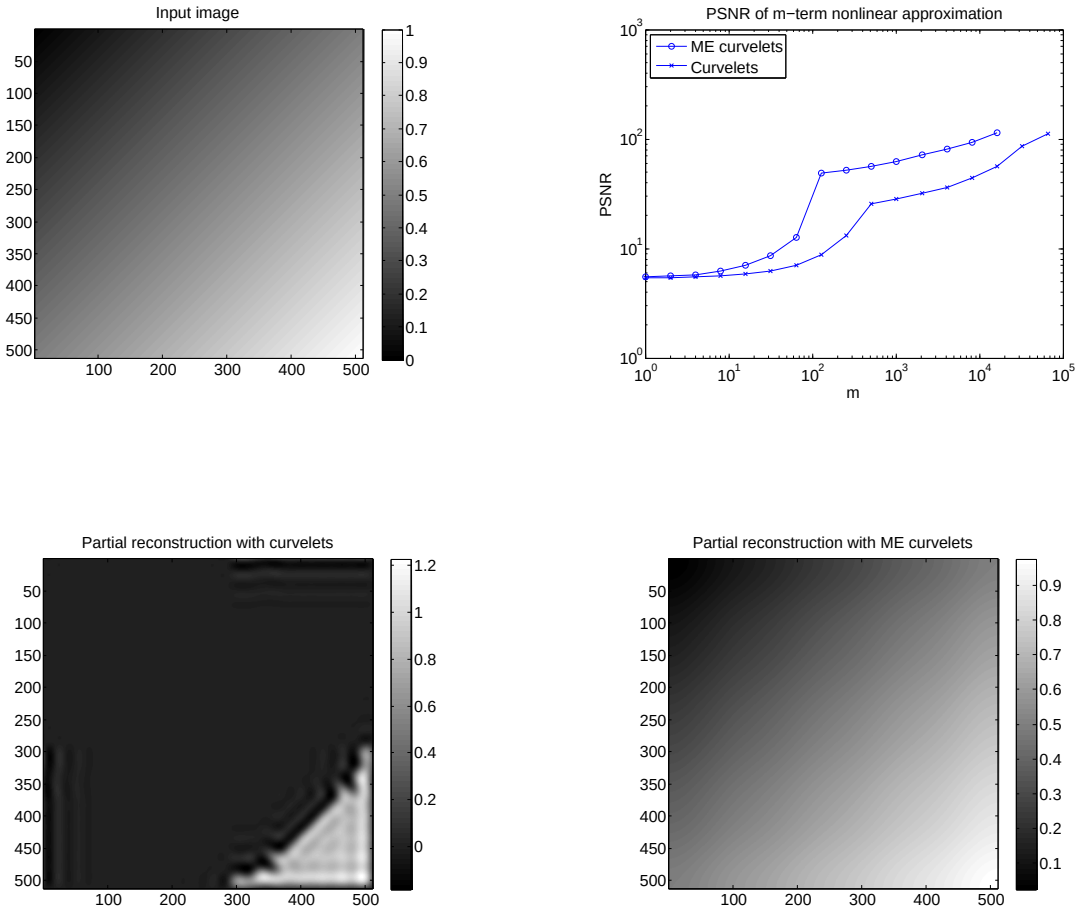

Figure 3: Top-left: original image. Top-right: PSNR curves of best m-term approximation for the curvelet and ME-curvelet transforms. (Regimes where the PSNR tops 50dB are interesting in numerical analysis, which is why we choose to plot the whole curve.) Bottom-left: partial reconstruction using the largest 512 curvelet coefficients (PSNR = 25.4). Bottom-right: partial reconstruction using the largest 512 ME-curvelet coefficients (PSNR = 56.6).

References

[1] E. J. Candès, L. Demanet, Curvelets and Fourier integral operators. *C. R. Acad. Sci. Paris, Ser. I* **336** (2003) 395–398.

[2] E. J. Candès and L. Demanet, The curvelet representation of wave propagators is optimally sparse, *Comm. Pure Appl. Math.*, **58-11** (2005) 1472–1528.

[3] E. J. Candès, L. Demanet, D. L. Donoho, L. Ying, Fast Discrete Curvelet Transforms, *SIAM Mult. Model. Sim.* **5-3** (2006) 861–899

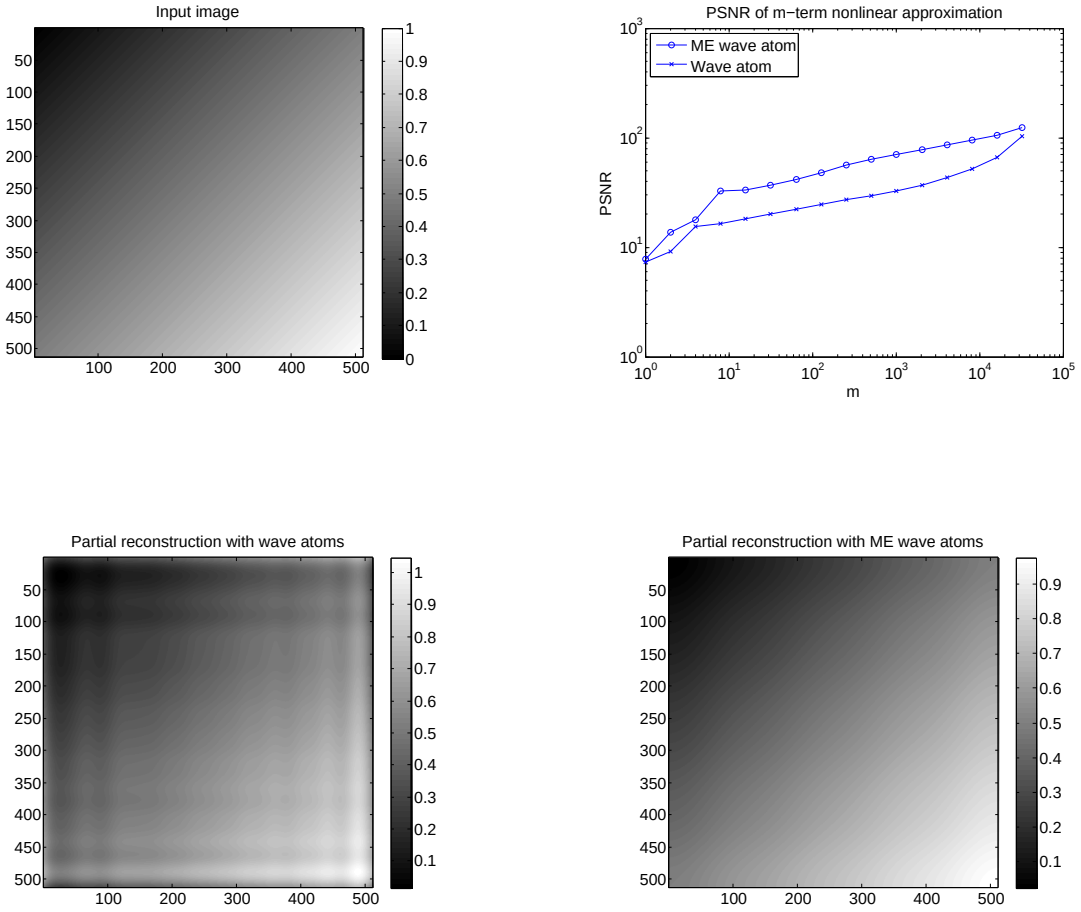

Figure 4: Top-left: original image. Top-right: PSNR curves of best m-term approximation for the wave atom and ME-wave atom transforms. Bottom-left: partial reconstruction using the largest 512 wave atom coefficients (PSNR = 60.1). Bottom-right: partial reconstruction using the largest 512 ME-wave atom coefficients (PSNR = 29.7).

[4] E. J. Candès and D. L. Donoho. Curvelets – a surprisingly effective nonadaptive representation for objects with edges. In C. Rabut A. Cohen and L. L. Schumaker, editors, *Curves and Surfaces*, pages 105–120, Vanderbilt University Press, 2000. Nashville, TN.

[5] E. J. Candès and D. L. Donoho. New tight frames of curvelets and optimal representations of objects with piecewise-C^2 singularities. *Comm. on Pure and Appl. Math.* **57** (2004) 219–266.

[6] C. Chaux, L. Duval, J. C. Pesquet, Image analysis using a dual-tree M-band wavelet transform, *to appear in IEEE Trans. Im. Proc.*

[7] H. Choi, J. Romberg, R. Baraniuk, and N. Kingsbury, Hidden Markov tree modeling of complex wavelet transforms, in *Proc. IEEE ICASSP* (2000).

[8] L. Demanet, *Curvelets, Wave Atoms and Wave Equations*, Ph. D. thesis, California Institute of Technology, 2006.

[9] L. Demanet and L. Ying, Wave atoms and sparsity of oscillatory patterns, *to appear in Appl. Comput. Harm. Anal.* (2007)

[10] L. Demanet and L. Ying, Wave atoms and time upscaling of wave equations, *submitted* (2007)

[11] M. Do and W. Vetterli, The contourlet transform: an efficient directional multiresolution image representation, *IEEE Trans. Im. Proc.* **14-12** (2005) 2091–2106

[12] M.J. Fadili and J.L. Starck, Sparse representation-based image deconvolution by iterative thresholding, *submitted* (2007)

[13] A. Federico and G. H. Kaufmann, Denoising in digital speckle pattern interferometry using wave atoms, *Opt. Lett.* **32** (2007) 1232–1234

[14] F. Fernandes, M. Wakin, R. Baraniuk, Non-redundant, linear-phase, semi-orthogonal, directional complex wavelets, in *Proc. IEEE ICASSP*, Montreal, Canada (2004)

[15] M. Frigo and S. G. Johnson, The Design and Implementation of FFTW3, in *Proceedings of the IEEE* **93-2** (2005) 216–231

[16] K. Guo and D. Labate, Optimally Sparse Multidimensional Representation using Shearlets. *SIAM J. Math Anal.* **39** (2007) 298–318

[17] G. Hennenfent and F. J. Herrmann. Seismic denoising with nonuniformly sampled curvelets, *Comput. in Sci. Eng.*, **8-3** (2006) 16–25

[18] F.J. Herrmann, U. Boeniger, and D.J. Verschuur. Nonlinear primary-multiple separation with directional curvelet frames, *Geophys. J. Int.* **170** (2007) 781–799

[19] F. J. Herrmann, P. P. Moghaddam, and C. C Stolk, Sparsity- and continuity-promoting seismic image recovery with curvelet frames, *to appear in Appl. Comput. Harmon. Anal.* (2007)

[20] L. Jacques and J.P. Antoine, Multiselective pyramidal decomposition of images: wavelets with adaptive angular selectivity, *to appear in Int. J. Wavelets, Multires. and Im. Proc.* (2007)

[21] N. Kingsbury, Image processing with complex wavelets, *Phil. Trans. Roy. Soc. A* **357-1760**, (1999), 2543–2560

[22] J. Ma, Characterization of textural surfaces using wave atoms, *Appl. Phys. Lett.* **90** 264101 (2007)

[23] Matlab documentation for the Signal processing toolbox and the Image processing toolbox. The Mathworks Inc.

[24] F. G. Meyer and R. R. Coifman, Brushlets: a tool for directional image analysis and image compression, *Applied Comput. Harmon. Anal.* **4** (1997) 147–187.

[25] G. Peyré and S. Mallat, Surface Compression With Geometric Bandelets, *ACM Trans. Graphics (Proc. SIGGRAPH 2005)*,**24-3** (2005) 601–608

[26] I. Selesnick, R. G. Baraniuk, N. G. Kingsbury, The dual-tree complex wavelet transform, *IEEE Sig. Proc. Mag.* **22(6)** (2005) 123–151

[27] J. L. Starck, E. J. Candès, and D. L. Donoho, The curvelet transform for image denoising. *IEEE Trans. Im. Proc.*, **11-6** (2002) 670–684

[28] P. Vandergheynst and J. F. Gobbers, Directional dyadic wavelet transforms: Design and algorithms. *IEEE Trans. Im. Proc.* **11-4** (2002) 363–372.

[29] L. Villemoes, Wavelet packets with uniform time-frequency localization, *Comptes-Rendus Mathematique*, **335-10** (2002) 793–796

[30] M. Wakin, J. Romberg, H. Choi, and R. Baraniuk, Wavelet-domain Approximation and Compression of Piecewise Smooth Images, *to appear in IEEE Trans. Im. Proc.*

[31] L. Ying, L. Demanet, and E. Candès, 3D discrete curvelet transform, *in Proc. SPIE wavelets XI*, San Diego (2005)