Coarse Quantization with the Fast Digital Shearlet Transform

Bernhard G. Bodmann[a], Gitta Kutyniok[b], and Xiaosheng Zhuang[b]

[a]Department of Mathematics, University of Houston, Texas, USA
[b]Institute of Mathematics, University of Osnabrück, Osnabrück, Germany

ABSTRACT

The fast digital shearlet transform (FDST) was recently introduced as a means to analyze natural images efficiently, owing to the fact that those are typically governed by cartoon-like structures. In this paper, we introduce and discuss a first-order hybrid sigma-delta quantization algorithm for coarsely quantizing the shearlet coefficients generated by the FDST. Radial oversampling in the frequency domain together with our choice for the quantization helps suppress the reconstruction error in a similar way as first-order sigma-delta quantization for finite frames. We provide a theoretical bound for the reconstruction error and confirm numerically that the error is in accordance with this theoretical decay.

Keywords: Shearlets, ShearLab, Hybrid Sigma-Delta Quantization, FDST, Tight Frame

1. INTRODUCTION

The benefits of coarse quantization include resilience against hardware imperfections and the idea of a democratic encoding of the signal content within the allocated bit budget. Democracy means in this context that different bits should not be distinguished by their significance. The notion of such a democratic encoding was examined by Calderbank and Daubechies, who showed that for the source coding of band-limited signals, the best achievable accuracy of such algorithms is inferior to that of fine quantization [1]. However, this notion of optimality does not include the possibility of losing part of the quantizer output. If random parts of a quantized signal are lost or corrupted during a transmission then having bits of different significance is undesirable, at least in the worst-case scenario. This motivates the search for robust, democratic encoding strategies associated with typical signal spaces.

In this paper, we investigate the encoding of images by applying a first-order sigma-delta quantizer to sequences of shearlet coefficients generated by the fast digital shearlet transform (FDST) recently introduced in [2]. This transform is the faithful digitization of the continuum domain shearlet transform, which provides optimally sparse approximations of natural images modeled by functions which are C^2 except for C^2 singularity curves [3,4] – so-called cartoon-like functions. Our sigma-delta quantization scheme uses a subband-decomposition in the frequency domain. Sigma-delta algorithms are naturally designed for coefficients of scaling functions, as opposed to those associated with wavelets or – here – shearlets. Therefore, we modify the typical frequency domain decomposition in the construction of shearlets to a decomposition into wedges. The number of wedges can be freely chosen, which provides an additional flexibility of our scheme. After the frequency domain is partitioned into wedges, the shearlet coefficients are computed for each wedge. The shearlets belonging to one wedge can be arranged into linearly ordered subsets. Adjusting the redundancy of the shearlet transform allows to control the difference between neighboring shearlet coefficients, in direct analogy with oversampling for bandlimited signals. The reconstruction error of the coarsely quantized shearlet coefficients is suppressed similarly as the redundancy ratio diverges, which we prove rigorously. Our numerical experiments then show that the error decay rate with growing redundancy behaves in accordance with the theoretical rate; they also show that increasing the directional selectivity is advantageous when images have edges in different orientations.

Further author information: (Send correspondence to X.Z.)
B.G.B.: E-mail: bgb@math.uh.edu
G.K.: E-mail: kutyniok@uos.de
X.Z.: E-mail: xzhuang@uos.de

2. THE FAST DIGITAL SHEARLET TRANSFORM

We start by giving a brief introduction to the fast digital shearlet transform (FDST). For further details, we refer the interested reader to [2].

A shearlet system is generated by parabolic scaling, shearing, and translation of a finite number of generating shearlets as follows.

DEFINITION 2.1. *For* $\phi, \psi^h, \psi^v \in L^2(\mathbf{R}^2)$, *the* shearlet system $\mathcal{SH}(\phi; \psi^h, \psi^v)$ *is defined by*

$$\mathcal{SH}(\phi; \psi^h, \psi^v) = \{\phi_m = \phi(\cdot - m) : m \in \mathbf{Z}^2\}$$
$$\cup \{\psi^h_{j,s,m} = 2^{\frac{3j}{2}} \psi^h(S_s A_j \cdot -m) \ : \ j \geq 0, |s| \leq 2^j, m \in \mathbf{Z}^2\}$$
$$\cup \{\psi^v_{j,s,m} = 2^{\frac{3j}{2}} \psi^v(S_s^T \tilde{A}_j \cdot -m) \ : \ j \geq 0, |s| \leq 2^j, m \in \mathbf{Z}^2\},$$

where

$$A_j = \begin{pmatrix} 4^j & 0 \\ 0 & 2^j \end{pmatrix}, \quad \tilde{A}_j = \begin{pmatrix} 2^j & 0 \\ 0 & 4^j \end{pmatrix}, \quad and \quad S_s = \begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix}.$$

The fast digital shearlet transform is based on shearlet systems $\mathcal{SH}(\phi; \psi^h, \psi^v)$, which are generated by band-limited functions $\phi, \psi^h, \psi^v \in L^2(\mathbf{R}^2)$. A typical choice for the two shearlet generators ψ^h and ψ^v is

$$\hat{\psi}^h(\xi) := \hat{\psi}^h(\xi_1, \xi_2) = \hat{\psi}_1(\xi_1)\hat{\psi}_2(\tfrac{\xi_2}{\xi_1}) \quad \text{and} \quad \hat{\psi}^v(\xi_1, \xi_2) = \hat{\psi}^h(\xi_2, \xi_1),$$

where ψ_1 is a wavelet and ψ_2 a 'bump' function. In the frequency domain, each function $\psi^h_{j,s,m}$ or $\psi^v_{j,s,m}$ generated by ψ^h or ψ^v, respectively, then exhibits a trapezoidal shaped support, which combines to a frequency tiling as shown in Figure 2.
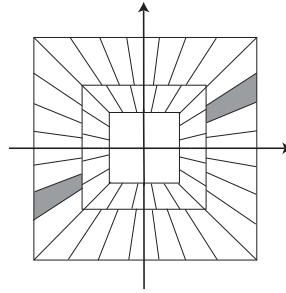

Figure 1. Frequency tiling by a shearlet system

The *discrete shearlet transform* is the map defined by:

$$L^2(\mathbf{R}^2) \ni f \mapsto \{\langle f, \phi_m \rangle, \langle f, \psi^\iota_{j,s,m} \rangle : \iota \in \{h, v\}, j \geq 0, |s| \leq 2^j, m \in \mathbf{Z}^2\}.$$

The FDST is designed to provide a faithful digitization of the discrete shearlet transform. It is based on the observation that the computation of the discrete shearlet transform consists of three main steps:

1. Continuous Fourier transform with change of variables from Cartesian to pseudo-polar coordinates.
2. Weighting by a radial 'density compensation' factor.
3. Decomposition into rectangular tiles and application of the 2D inverse Fourier transform to these tiles.

To ensure faithfulness, the digitization of the discrete shearlet transform, i.e., the algorithm FDST, also consists of three main steps which correspond to the three steps of the discrete shearlet transform. Thus the FDST comprises the following steps:

1. PPFT. Mapping of an image on the Cartesian grid to an image on a pseudo-polar grid by the fast pseudo-polar Fourier transform (PPFT).
2. WEIGHTING. Appropriate weighting of the pseudo-polar grid in order to provide isometry of PPFT.
3. WINDOWING. Decomposition of the pseudo-polar grid to rectangular tiles.

2.1 Weighted Pseudo-Polar Fourier Transform

We first discuss Steps 1 and 2 of the aforementioned three steps of the FDST. For this, let $I = \{I(u,v) : -L/2 \leq u, v \leq L/2 - 1\}$ be an image defined as samples on an $L \times L$ Cartesian grid. The PPFT computes the Fourier transform of this data on the so-called pseudo-polar grid. More precisely, it computes $\hat{I} = \{\hat{I}(\omega_x, \omega_y) : (\omega_x, \omega_y) \in \Omega_R\}$ with the pseudo-polar grid Ω_R defined by

$$\Omega_R := \Omega_R^v \cup \Omega_R^h,$$

where $R \geq 1$ is an oversampling factor along the radial direction, and the vertical and horizontal cone, Ω_R^v and Ω_R^h, are given by

$$\Omega_R^v = \left\{ \left(-\frac{2k}{R} \cdot \frac{2\ell}{L}, \frac{2k}{R} \right) : -L/2 \leq \ell \leq L/2, -RL/2 \leq k \leq RL/2 \right\},$$

$$\Omega_R^h = \left\{ \left(\frac{2k}{R}, -\frac{2k}{R} \cdot \frac{2\ell}{L} \right) : -L/2 \leq \ell \leq L/2, -RL/2 \leq k \leq RL/2 \right\}.$$

For an illustration of Ω_R, Ω_R^v, and Ω_R^h, we refer to Figure 2. The pseudo-polar Fourier transform $\hat{I}$ of I is then

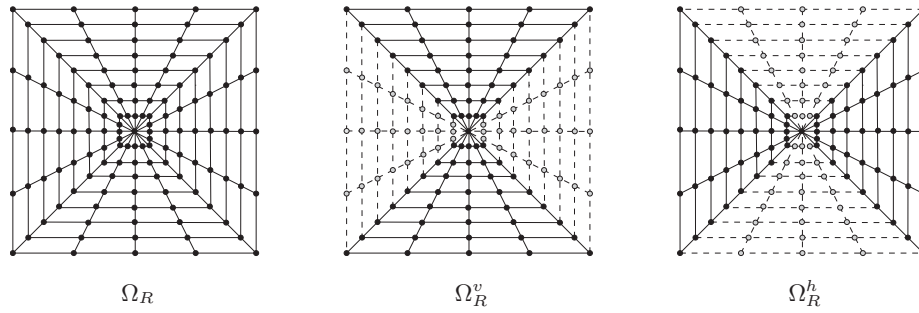$$\Omega_R \qquad\qquad \Omega_R^v \qquad\qquad \Omega_R^h$$

Figure 2. A pseudo-polar grid $\Omega_R = \Omega_R^v \cup \Omega_R^h$ with $L = 4$ and $R = 4$.

given by

$$\hat{I}(\omega_x, \omega_y) = \sum_{u,v=-L/2}^{L/2-1} I(u,v) e^{-\frac{2\pi i}{m_0}(u\omega_x + v\omega_y)}, \quad (\omega_x, \omega_y) \in \Omega_R, \tag{1}$$

where $m_0 \geq L$ is an integer.

Similar to the FFT which provides a fast algorithmic realization of the DFT, a fast algorithm to compute the PPFT was provided in [5]. The basic idea is that on each cone, e.g., the vertical cone Ω_R^v, $\hat{I}(\omega_x, \omega_y)$ with $\omega_x = -4k\ell/RL$ and $\omega_y = 2k/R$ can be rewritten as

$$\hat{I}(\omega_x, \omega_y) = \sum_{u=-L/2}^{L/2-1} \left(\sum_{v=-L/2}^{L/2-1} I(u,v) e^{-2\pi i \frac{vk}{m_0 R/2}} \right) e^{-2\pi i \cdot u\ell \cdot \frac{-k}{(m_0 R/2) \cdot L/2}}.$$

Thus $\hat{I}$ can be computed on Ω_R^v by first applying the discrete fractional Fourier transform (see [6]) to each vector of I along the direction v with a fixed fractional order $1/(m_0 R/2)$, and then applying the discrete fractional Fourier transform once again to the resulting image along the direction u with fractional order depending on k. Since the implementation of the discrete fractional Fourier transform is based on the FFT, the total complexity of the PPFT is $O(L^2 \log L)$.

Since the density of the sampling points given by the pseudo-polar grid is not uniform, it is intuitively clear that the PPFT cannot be an isometry. This is indeed the case, which however causes problems when utilizing it as an algorithmic means for a 'change of variables from Cartesian to pseudo-polar coordinates'. Another problem

is that the inverse PPFT is not simply its adjoint. To resolve this issue, in [2] a framework for deriving weight functions $w : \Omega_R \to \mathbf{R}^+$ satisfying the isometry condition

$$\sum_{u,v} |I(u,v)|^2 = \sum_{(\omega_x, \omega_y) \in \Omega_R} w(\omega_x, \omega_y) \cdot |\hat{I}(\omega_x, \omega_y)|^2 \tag{2}$$

was developed. Given such a weight function w, the weighted PPFT defined by

$$\hat{I}_w(\omega_x, \omega_y) = \sqrt{w(\omega_x, \omega_y)} \sum_{u,v=-L/2}^{L/2-1} I(u,v) e^{-\frac{2\pi i}{m_0}(u\omega_x + v\omega_y)}, \quad (\omega_x, \omega_y) \in \Omega_R \tag{3}$$

is an isometry, thereby allowing us to use the adjoint as a left inverse. Since weight functions w satisfying (2), i.e., providing exact isometry, are difficult to obtain due to the complexity of the associated linear system of equations, in [2] the 'exact isometry' condition was also relaxed to 'almost isometry' conditions, thereby providing more flexibility in the choice of weight functions.

2.2 Windowing and the Wedge-Cutting Algorithm

We next discuss Step 3 of the aforementioned three steps of FDST. For this, let $\hat{I}_w$ be the weighted PPFT of an image I as defined in (3). Step 3 requires the design of a sequence of digital shearlets $\{\varphi_m, \sigma^\iota_{j,s,m}\}$ on Ω_R, which digitize the shearlets $\{\phi_m, \psi^\iota_{j,s,m}\}$ on the pseudo-polar grid Ω_R. It then consists of computing inner products of $\hat{I}_w$ with each of the digital shearlets, i.e., computing the digital shearlet coefficients. Such a digitization has been given in [2] based on a partition of unity using the Meyer wavelet function and a smooth 'bump' function leading to the 'windowing' step used in the FDST.

In this paper, we are however 'only' interested in decomposing the pseudo-polar grid Ω_R into pieces of low-pass subbands. For such a decomposition, a smooth 'bump' function is sufficient, which though provides the flexibility to choose different numbers of wedges. To distinguish this modified Step 3 from Step 3 of FDST, we coin it 'wedge-cutting algorithm'.

For this, we let V be a function supported on $[-1, 1]$ such that

$$|V(\xi - 1)|^2 + |V(\xi)|^2 + |V(\xi + 1)|^2 = 1 \quad \forall |\xi| \leq 1, \quad \xi \in \mathbf{R},$$

which implies

$$\sum_{s=-2^j}^{2^j} |V(2^j \xi - s)|^2 = 1 \quad \forall |\xi| \leq 1, \quad \xi \in \mathbf{R}.$$

It is easy to construct such functions, e.g., $V(\xi) = \sqrt{\nu(1 + \xi) + \nu(1 - \xi)}$ with $\nu(x) = x^4(35 - 84x + 70x^2 - 20x^3)$ for $0 \leq x \leq 1$, $\nu(x) \equiv 0$ for $x < 0$, and $\nu(x) \equiv 1$ for $x > 1$ (cf. [2]).

Our means to parameterize the number of wedges into which the pseudo-polar grid Ω_R will be decomposed will be the scale $j \geq 0$. More precisely, for a given scale j and focussing exemplarily on the cone Ω_R^h and shear $s \in \{-2^j + 1, \ldots, 2^j - 1\}$, the function $(\omega_x, \omega_y) \mapsto V(s + 2^j \frac{\omega_x}{\omega_y})$ extracts from Ω_R^h the wedge-shaped tile

$$\{(k, \ell) \in \{-RL/2, \ldots, RL/2\} \times \{2^{-j-1}L(s-1), \ldots, 2^{-j-1}L(s+1)\}\}.$$

Notice that for the extreme cases $s = \pm 2^j$, the parameter ℓ ranges only over half of $\{2^{-j-1}L(s-1), \ldots, 2^{-j-1}L(s+1)\}$, i.e., either $\{L/2^{j+1}, \ldots, L/2\}$ or $\{-L/2^{j+1}, \ldots, -L/2\}$.

Now set

$$\varphi^v_{s,m}(\omega_x, \omega_y) := C(\omega_x, \omega_y) V(s + 2^j \frac{\omega_x}{\omega_y}) e^{2\pi i \frac{m_1 k}{RL+1}} e^{2\pi i \frac{m_2(\ell - s)}{L/2^j + 1}} \cdot \chi_{\Omega_R^v}(\omega_x, \omega_y),$$

$$\varphi^h_{s,m}(\omega_x, \omega_y) := C(\omega_x, \omega_y) V(s + 2^j \frac{\omega_y}{\omega_x}) e^{2\pi i \frac{m_1 k}{RL+1}} e^{2\pi i \frac{m_2(\ell - s)}{L/2^j + 1}} \cdot \chi_{\Omega_R^h}(\omega_x, \omega_y), \tag{4}$$

where $s = -2^j, \ldots, 2^j$, $m = (m_1, m_2)$ ranges over the same index set as (k, ℓ) cut by the 'bump' function $V(s + 2^j \cdot)$. In addition to the bump function, we introduce additional weights

$$
C(\omega_x, \omega_y) = \begin{cases} \frac{1}{\sqrt{2(N+1)}} & : \quad (\omega_x, \omega_y) = (0, 0), \\ \frac{1}{\sqrt{2}} & : \quad |\omega_x| = |\omega_y| \text{ and } (\omega_x, \omega_y) \neq (0, 0), \\ 1 & : \quad \text{else}. \end{cases}
$$

to compensate for multiple occurrences of some points on the pseudo-polar grid in several tiles.

To ensure uniform support sizes among each subband – advantageous for our quantization scheme – we replace $\varphi^v_{\pm 2^j, m}$ and $\varphi^h_{\pm 2^j, m}$ by the new "seamline" or "diagonal" elements defined by

$$
\begin{aligned}
\varphi^d_{2^j, m}(\omega_x, \omega_y) &:= \varphi^h_{2^j, m}(\omega_x, \omega_y) + \varphi^v_{2^j, m}(\omega_x, \omega_y), \\
\varphi^d_{-2^j, m}(\omega_x, \omega_y) &:= \varphi^h_{-2^j, m}(\omega_x, \omega_y) + \varphi^v_{-2^j, m}(\omega_x, \omega_y).
\end{aligned}
\tag{5}
$$

For $j < 0$, we need to choose $V \equiv 1$, which is the very special case of having just two wedges: Ω^v_R and Ω^h_R, and $\varphi^d_{\pm 1, m}$ are defined similarly as in (4) for each cone independently.

This system exhibits redundancy – essential for a sigma-delta quantization scheme to work –, but in a stable manner, i.e., it forms a tight frame. For the convenience of the reader, we first recall this notion in an abstract setting.

DEFINITION 2.2. *Let $\mathcal{H}$ be a D-dimensional Hilbert space. A sequence of vectors $\mathcal{F} = \{\varphi_1, \ldots, \varphi_N\}$ in $\mathcal{H}$ is called an A-tight frame, if for every $x \in \mathcal{H}$, the norm equality $\|x\|_2^2 = \frac{1}{A} \sum_{i=1}^N |\langle x, \varphi_i \rangle|^2$ holds.*

For later use, we also consider doubly indexed vectors and let a tight frame $\mathcal{F} = \cup_{n=1}^M \mathcal{F}_n$, be composed of M subsequences $\mathcal{F}_n = \{\varphi_{1,n}, \ldots, \varphi_{N,n}\}$. We remark that we can also interpret such a frame as a fusion frame, see [7]. The digital shearlet system we introduced in this subsection indeed forms such an A-tight frame as the following result shows. We specialize to $j \geq 0$ to avoid technicalities.

THEOREM 2.3. *Given integers $j \geq 0$ and $R, L \in \mathbf{N}$ such that 2^j divides L, the system $\mathcal{DSH} := \{\varphi^h_{s,m}, \varphi^v_{s,m} : s = -2^j + 1, \ldots, 2^j - 1; m\} \cup \{\varphi^d_{s,m} : s = \pm 2^j; m\}$ defined as in (4) and (5) forms an A-tight frame for functions $I : \Omega_R \to \mathbf{C}$, where $A = (RL + 1)(L/2^j + 1)$.*

Proof. We need to establish that for any function on Ω_R the Parseval-type identity for A-tight frames holds. Equivalently, it is sufficient to check this for each function $\delta_{\omega_x, \omega_y}$ which is equal to one at one point $(\omega_x, \omega_y) \in \Omega_R$ and vanishes elsewhere, we have

$$
\delta_{\omega_x, \omega_y} = \frac{1}{A} \sum_{\iota, s, m} \langle \delta_{\omega_x, \omega_y}, \varphi^\iota_{s,m} \rangle \varphi^\iota_{s,m}.
$$

This expression simplifies because the weights and the squares of the shifted bump functions implicit on the right hand side add to one. The remaining tightness constant A is then simply the normalization required of the 2D-iFFT which goes with the exponential term in the definition of $\varphi^\iota_{s,m}$.

The algorithmic realization of the wedge-cutting algorithm is now straightforward. Given an image $\hat{I}_w$ on the pseudo-polar grid Ω_R (already processed by Steps 1 and 2) and some $j \in \mathbf{Z}$, 'bump' functions $V(s + 2^j \cdot)$ are generated along the shearing direction of Ω_R. This splits the pseudo-polar domain into several overlapping wedges, see Figure 3. Then $\hat{I}_w$ is pointwise multiplied with each of these functions, followed by application of the 2D-iFFT to each wedge, thus producing the shearlet coefficients.

3. SIGMA-DELTA QUANTIZATION WITH THE FDST

We now turn to introducing our sigma-delta quantization scheme for coarsely quantizing the shearlet coefficients computed by weighted PPFT (Subsection 2.1) and wedge-cutting (Subsection 2.2).
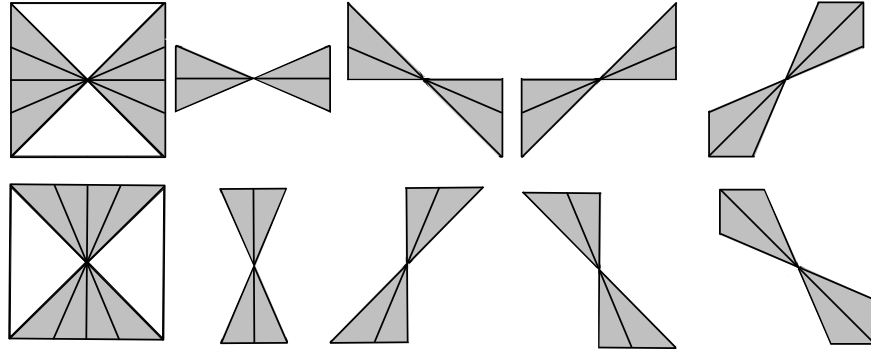
Figure 3. Wedge-cutting algorithm for $j = 1$. First column: horizontal and vertical cone. Middle three columns: cutted wedges inside each cone. Last column: seamline (diagonal) wedges.

3.1 Hybrid Sigma-Delta Quantization

Given a tight frame $\mathcal{F}_n = \{\varphi_{1,n}, \ldots, \varphi_{N,n}\}$, for any vector (signal) $x \in \mathcal{H}$, we can represent x as a sequence of frame coefficients $\{\langle x, \varphi_{i,n} \rangle : i = 1, \ldots, N, n = 1, \ldots, M\}$. We henceforth only consider frames for real Hilbert spaces. To quantize the coefficients, we first need an alphabet and a quantizer. Let $K \in \mathbf{N}$ and $\delta > 0$. The *midrise quantization alphabet* $\mathcal{A}_K^\delta$ is given by

$$\mathcal{A}_K^\delta := \{(-K + \frac{1}{2})\delta, \ldots, -\frac{1}{2}\delta, \frac{1}{2}\delta, \ldots, (K - \frac{1}{2})\delta\},$$

which consists of $2K$ elements. For $K = 1$, we have $\mathcal{A}^\delta = \{\pm\frac{1}{2}\delta\}$, which means we only use 1 bit to specify the output of the quantizer. The *scalar quantizer* Q with such an alphabet is given by

$$Q(u) := \mathrm{argmin}_{q \in \mathcal{A}_K^\delta} |u - q|.$$

With a sequence of frame coefficients, an alphabet, and a quantizer, we introduce a hybrid sigma-delta quantization scheme as follows:

DEFINITION 3.1. *Given $K \in \mathbf{N}$ and $\delta > 0$, let $\mathcal{A}_K^\delta$ and Q be defined as above. Then the* hybrid (first-order) sigma-delta quantization *of the sequence of frame coefficients $\{\langle x, \varphi_{i,n} \rangle : i = 1, \ldots, N; n = 1, \ldots, M\}$ is given by*

$$\begin{aligned} q_{i,n} &= Q(\langle x, \varphi_{i,n} \rangle + u_{i-1,n}) \\ u_{i,n} &= \langle x, \varphi_{i,n} \rangle - q_{i,n} + u_{i-1,n} \end{aligned} \quad \text{with} \quad u_{0,n} = 0; i = 1, \ldots, N.$$

The sigma-delta quantization produces a sequence of quantized coefficients $\{q_{i,n} : i = 1, \ldots, N; n = 1, \ldots, M\}$ and auxiliary sequences $\{u_{i,n} : i = 1, \ldots, N; n = 1, \ldots, M\}$. The hybrid nature of this quantization is that the quantization of a coefficient depends on the preceding ones belonging to the same n, whereas for different values of the index n, rounding is independent.

We can reconstruct a vector using the A-tight frame by defining

$$Q_\mathcal{F}(x) := \frac{1}{A} \sum_{i=1}^{N} \sum_{n=1}^{M} q_{i,n} \varphi_{i,n}.$$

Also, the sigma-delta quantization is stable [8–10] in the sense that $\{u_{i,n} : i = 1, \ldots, N, n = 1, \ldots, M\}$ is uniformly bounded if the input sequence $\{\langle x, \varphi_{i,n} \rangle : i = 1, \ldots, N, n = 1, \ldots, M\}$ is uniformly bounded. More precisely, we have

$$|u_{i,n}| \le \frac{\delta}{2} \quad \forall i = 1, \ldots, N, n = 1, \ldots, M \quad \text{provided} \quad |\langle x, \varphi_{i,n} \rangle| \le (K - \frac{1}{2})\delta \quad \forall i = 1, \ldots, N, n = 1, \ldots, M.$$

The reconstruction error is then bounded as follows.

THEOREM 3.2. *Let $\mathcal{F} = \{\varphi_{i,n} : i = 1, \ldots, N, n = 1, \ldots, M\}$ be an A-tight frame, and let $Q_{\mathcal{F}}(x)$ be defined as above, with $\max_{i,n} |\langle x, \varphi_{i,n}\rangle| \leq (K - 1/2)\delta$, then*

$$\|x - Q_{\mathcal{F}}(x)\|_2 \leq \frac{1}{A}\frac{\delta}{2}\sum_{n=1}^{M}\sigma(\mathcal{F}_n),$$

where $\sigma(\mathcal{F}_n) = \|\varphi_{1,n} - \varphi_{2,n}\|_2 + \cdots + \|\varphi_{N-1,n} - \varphi_{N,n}\|_2 + \|\varphi_{N,n}\|_2$.

Proof. By using Minkowski's inequality, together with the previous error bounds for first-order sigma-delta quantization [8, 9], we obtain

$$
\begin{aligned}
\|x - Q_{\mathcal{F}}(x)\|_2 &\leq \|\frac{1}{A}\sum_{n=1}^{M}\sum_{i=1}^{N}(u_{i,n} - u_{i-1,n})\varphi_{i,n}\|_2 \\
&\leq \frac{1}{A}\|\sum_{n=1}^{M}\sum_{i=1}^{N-1}u_{i,n}(\varphi_{i,n} - \varphi_{i+1,n}) + u_{N,n}\varphi_{N,n}\|_2 \\
&\leq \frac{1}{A}\frac{\delta}{2}\sum_{n=1}^{M}\sigma(\mathcal{F}_n).
\end{aligned}
$$

The theorem is proved. □

For many families of frames, the tightness constant A is usually proportional to the number of frame vectors, and each $\sigma(\mathcal{F}_n)$ is uniformly bounded, independent of N. Hence, the reconstruction error decays as N (or the redundancy ratio of the frame) increases.

We implement this type of hybrid quantization with the shearlet transform, where oversampling is only applied in the radial direction of the frequency domain, not in the directional selectivity. To this end, we let $i = m_1$ and $n = (s, m_2, \iota)$ where $\iota = h, v, d$ and (m_1, m_2) indexes the modulations on the respective wedges. This means

$$M = 2 \cdot (2 \cdot 2^j) \cdot (L/2^j + 1), \quad j \in \mathbf{N}, \tag{6}$$

where the first factor corresponds to the separation of the frequency domain into horizontal and vertical pieces, the second factor gives the number of wedges, and the third the number of directions in each wedge. We partition $\mathcal{DSH} = \cup_n \mathcal{F}_n$ and perform the hybrid quantization.

COROLLARY 3.1. *Let $j \geq 0$ and $R, L \in \mathbf{N}$ such that 2^j divides L, and the system $\mathcal{DSH} := \{\varphi_{s,m}^h, \varphi_{s,m}^v : s = -2^j + 1, \ldots, 2^j - 1; m\} \cup \{\varphi_{s,m}^d : s = \pm 2^j; m\}$ be defined as in (4) and (5), then the 1-bit hybrid sigma-delta quantization of an input vector $x \in \operatorname{span}\mathcal{DSH}$ with $|\langle x, \varphi_{s,m}^\iota\rangle| \leq \delta/2$ for all $\iota = \{h, v, d\}, s, m$, yields an error bounded by*

$$\|x - Q_{\mathcal{DSH}}(x)\| \leq \frac{\delta}{2} \cdot \frac{2^{j+2}}{RL + 1}\sigma_{\max}$$

where $\sigma_{\max} = \max_n \sigma(\mathcal{F}_n)$ maximizes the path length over all "radial" subsets of $\mathcal{DSH}$ in the partition.

Proof. Substituting $A = (RL + 1)(L/2^j + 1)$ and M in (6) into the result of Theorem 3.2 gives the claimed expression. □

The experimental portion of this paper examines whether the terms in this bound contribute indeed as suggested in the proof.

3.2 Implementation of the Quantization with the FDST

We are now ready to discuss the algorithmic realization of our hybrid sigma-delta quantization scheme as introduced in Subsection 3.1.

Given an image I of size $N \times N$, we first compute its weighted PPFT image $\hat{I}_w$ on a pseudo-polar grid Ω_{R_1}. To introduce redundancy for sigma-delta quantization, we further embed $\hat{I}_w$ into a larger pseudo-polar grid Ω_{R_2} with $R_2 \geq R_1$; that is, we define

$$J_w(\omega_x, \omega_y) = \begin{cases} \hat{I}_w(\omega_x, \omega_y) & : & (\omega_x, \omega_y) \in \Omega_{R_1} \cap \Omega_{R_2}, \\ 0 & : & \text{otherwise.} \end{cases}$$

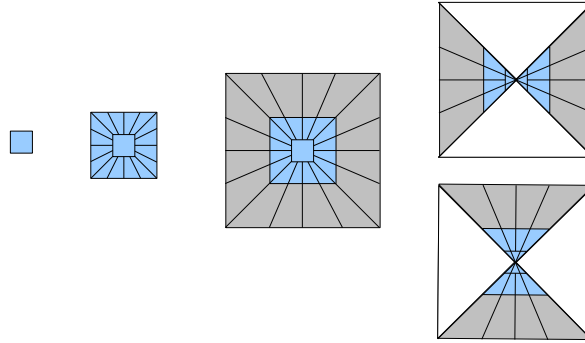This embedding is illustrated in Figure 4.

Figure 4. Weighted PPFT image $\hat{I}_w$ and embedding image J_w of $\hat{I}_w$. From left to right: original size image I, weighted PPFT image $\hat{I}_w$, embedding image J_w of $\hat{I}_w$, and vertical cone image J_w^v and horizontal cone image J_w^h of J_w.

Next, for a fixed $j \in \mathbf{Z}$, we decompose the embedded image J_w into small (overlapping) wedge-like pieces. Then we apply the 2D-iFFT to each wedge producing a sequence of coefficient matrices. Finally, we perform the hybrid sigma-delta quantization scheme on the coefficient matrix.

This procedure is detailed in Algorithm 1.

Algorithm 1 Hybrid Sigma-Delta Quantization with the FDST

(a) **Input**: Image I of size $L \times L$. Oversampling factors: R_1 for PPFT and R_2 for sigma-delta quantization. Number of wedges: determined by $j \in \mathbf{Z}$. Alphabet size: $2K, K \in \mathbf{N}$.

(b) **Output**: Reconstructed image $\tilde{I}$.

(c) **Hybrid Sigma-Delta Quantization with the FDST**:

1: $\hat{I}_w$: Application of the weighted PPFT to I to obtain an image $\hat{I}_w$ on the pseudo-polar grid Ω_{R_1} as shown in Subsection 2.1.

2: $J_w = J_w^h \cup J_w^v$: Embedding of $\hat{I}_w$ into a larger pseudo-polar grid Ω_{R_2}. This generates a new image J_w, which consists of a horizontal image J_w^h and a vertical image J_w^v.

3: $J_{w,s}^h, J_{w,s}^v, s = -2^j, \ldots, 2^j$: Decomposition of J_w into small pieces $J_{w,s}^h, J_{w,s}^v$ according to j as shown in Subsection 2.2.

4: $\hat{J}_{w,s}^h, \hat{J}_{w,s}^v, s = -2^j, \ldots, 2^j$: Application of 2D-iFFT to each piece $J_{w,s}^h, J_{w,s}^v$ to derive coefficient matrices $\hat{J}_{w,s}^h, \hat{J}_{w,s}^v$. Let c_{max} to be the absolute maximal coefficients among all coefficients matrices.

5: $Q_{w,s}^h, Q_{w,s}^v, s = -2^j, \ldots, 2^j$: Application of the hybrid sigma-delta quantization with $\delta = \frac{c_{max}}{K-1/2}$ as discussed in Subsection 3.1 to each coefficient matrix $\hat{J}_{w,s}^h, \hat{J}_{w,s}^v$, producing the quantized coefficient matrices $Q_{w,s}^h, Q_{w,s}^v$.

6: $\tilde{I}$: Reconstruction of the image $\tilde{I}$ by 'going backwards' and applying the adjoint of the above steps (4-3-2-1) to the quantized coefficient matrices $Q_{w,s}^h, Q_{w,s}^v$.

3.3 Performance

We finally discuss our numerical experiments to determine the decay rate of the error of our hybrid sigma-delta quantization scheme. For fixed $j \in \{-1, 0, 1\}$, i.e., a fixed number of wedges, for fixed $R_1 = 2$, and for each redundancy factor $R_2 = 2^{k_0}, k_0 = 3, \ldots, 10$, we generate 10 random images $I_i, i = 1, \ldots, 10$ of size 128×128 with normally distributed entries inside a fixed size disc (see Figure 5 for an example of such an image). Then we apply Algorithm 1 using only 1 bit, i.e., $\mathcal{A}_2^\delta = \{-\frac{1}{2}\delta, \frac{1}{2}\delta\}$, to each of these random images and compute the reconstruction error $\|I - Q_{\mathcal{F}}(I)\|_2$. Our hypothesis is that the reconstruction error behaves like $\|I - Q_{\mathcal{F}}(I)\|_2 = c \cdot R_2^\lambda$ for some $\lambda < 0$ and $c \in \mathbf{R}$ due to Corollary 3.1 with $\lambda \leq -1$.
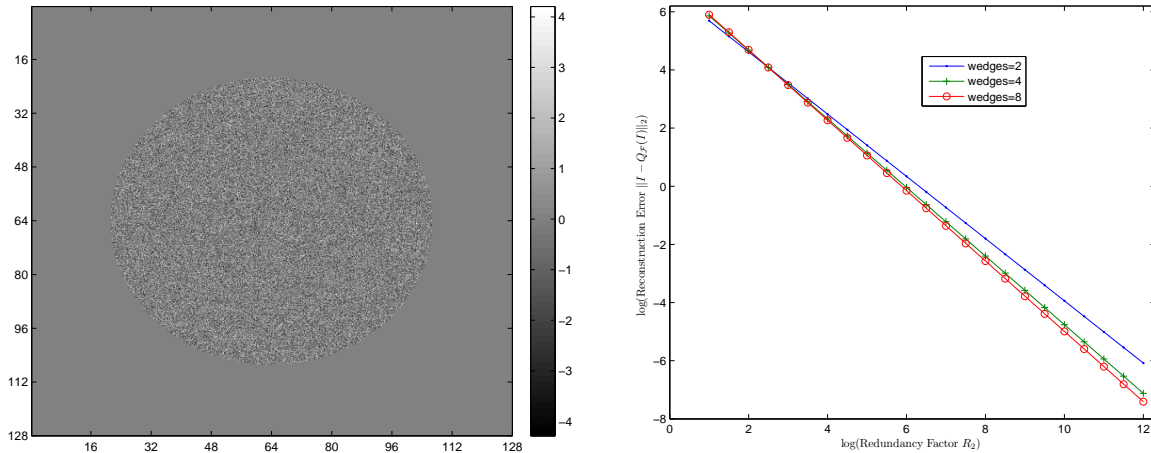

Figure 5. Left: test image. Right: log-log plot of lines of different decay rates with respect to the number of wedges (x-axis: log(Redundancy Factor R_2), y-axis: log(Reconstruction Error $\|I - Q_{\mathcal{F}}(I)\|_2$)).

Figure 5 shows the numerically determined decay rates for $2, 4$, and 8 wedges (corresponding to $j \in \{-1, 0, 1\}$), i.e., λ_{-1}, λ_0, and λ_1. For 2 wedges we obtain $\lambda_{-1} \approx -1.07$, for 4 wedges $\lambda_0 \approx -1.18$, and for 8 wedges $\lambda_1 \approx -1.21$. Firstly, this confirms Corollary 3.1 (or more generally Theorem 3.2) in its claim that the reconstruction error decays at least linearly in terms of redundancy ratio. Secondly, we observe that the estimated decay becomes significantly better as the number of wedges grows, thereby showing the advantage of having a directional based quantization scheme.

A more precise overview of the numerically determined data based on which Figure 5 (right) was generated is shown in Table 1.

Table 1. Numerical test data averaged over 10 simulated images.

$j \backslash R_2$	8	16	32	64	128	256	512	1024	intercept	slope(λ_j)
-1	105.77	41.07	20.34	9.66	5.24	2.94	1.15	0.47	6.76	-1.07
0	129.83	46.59	16.34	6.54	3.48	1.84	0.83	0.36	7.04	-1.18
1	131.24	51.46	17.77	6.44	3.04	1.49	0.68	0.40	7.11	-1.21

4. CONCLUSION

In this paper, we introduce a hybrid sigma-delta quantization scheme for quantizing digital shearlet coefficients. These coefficient are generated by a slight modification of the fast digital shearlet transform (FDST), since only a decomposing of the pseudo-polar grid into pieces of low-pass subbands is explored. The possibility of a variable redundancy necessary for sigma-delta quantization is already present in the FDST, since the shearlet coefficients are computed on a pseudo-polar grid. We provide numerical tests to estimate the decay rate of the reconstruction error as the redundancy increases. Our numerical results show that the reconstruction error decays at least linearly in terms of redundancy ratio. It also gives evidence to the fact that the decay rate increases with the number of wedges, showing the advantage of having a directional based quantization scheme.

ACKNOWLEDGMENTS

REFERENCES

[1] Calderbank, A. R. and Daubechies, I., "The pros and cons of democracy," *IEEE Trans. Inf. Theory* **48**, 1721–1725 (2002).

[2] Kutyniok, G., Shahram, M., and Zhuang, X., "Shearlab: A rational design of a digital parabolic scaling algorithm," *Preprint* (2011).

[3] Guo, K. and Labate, D., "Optimally sparse multidimensional representation using shearlets," *SIAM J. Math. Anal.* **39**, 298–318 (2007).

[4] Kutyniok, G. and Lim, W.-Q., "Compactly supported shearlets are optimally sparse," *J. Approx. Theory* (to appear).

[5] Averbuch, A., Coifman, R. R., Donoho, D. L., Israeli, M., and Shkolnisky, Y., "A framework for discrete integral transformations i – the pseudo-polar fourier transform," *SIAM J. Sci. Comput.* **30**, 764–784 (2008).

[6] Bailey, D. H. and Swarztrauber, P. N., "The fractional fourier transform and applications," *SIAM Rev.* **33**, 389–404 (1991).

[7] Casazza, P. G., Kutyniok, G., and Li, S., "Fusion frames and distributed processing," *Appl. Comput. Harmon. Anal.* **25**, 114–132 (2008).

[8] Benedetto, J. J., Yilmaz, O., and Powell, A. M., "Sigma-delta quantization and finite frames," *IEEE Trans. Info. Theory* **52**, 1990–2005. (2006).

[9] Bodmann, B. G. and Paulsen, V. I., "Frame paths and error bounds for sigma-delta quantization of finite-frame expansions," *Appl. Comput. Harmon. Anal.* **20**, 126–148 (2006).

[10] Daubechies, I. and DeVore, R. A., "Approximating a bandlimited function using very coarsely quantized-data: a family of stable sigma-delta modulators of arbitrary order," *Ann. Math.* **158**, 679–710 (2003).