EFFICIENT DETERMINISTIC COMPRESSED SENSING FOR IMAGES WITH CHIRPS AND REED-MULLER SEQUENCES*

KANGYU NI†, SOMANTIKA DATTA‡, PRASUN MAHANTI§, SVETLANA ROUDENKO¶, AND DOUGLAS COCHRAN§

Abstract. A recent approach to compressed sensing using deterministic sensing matrices formed from discrete frequency-modulated chirps or from Reed-Muller sequences is extended to support efficient deterministic reconstruction of signals that are less sparse than envisioned in the original work. In particular, this allows the application of this approach in imaging. The reconstruction algorithm developed for images incorporates several new elements to improve computational complexity and reconstruction fidelity in this application regime.

Key words. Compressed sensing, Reed-Muller sequences, FM chirps, Image reconstruction, Medical imaging

AMS subject classifications. 92C55 (Biomedical Imaging and Signal Processing); 94A08 (Image Processing); 94A12 (Signal Theory); 65R32 (Inverse Problems)

1. Introduction. In the few years since the foundational ideas of compressed sensing were set forth by Donoho [17] and Candès and Tao [12, 8], the methodology had inspired a substantial body of research seeking to exploit sparsity in various classes of signals to enable efficient measurement approaches. While most of the emphasis has been connected with the use of stochastic measurement matrices, a few researchers have sought to develop deterministic measurement strategies. Among these is an approach introduced by Applebaum et al. [1] and extended by Howard et al. [21] in which the columns of the measurement matrix consist of sampled linear frequency-modulated chirps or the closely related second-order Reed-Muller sequences. This approach includes a deterministic algorithm for reconstructing the original sparse signal from measurements that is shown to compare favorably to reconstruction methods used in other compressed sensing contexts, particularly so when the signal is extremely sparse. The performance of this algorithm deteriorates significantly, both in speed and fidelity, when the signal is less sparse. In most image processing applications, intrinsic compressibility of classes of images using a suitable basis (e.g., wavelets) typically makes high-accuracy approximation by a sparse image possible only if the sparsity is 5%-20% or higher. Thus, the reconstruction algorithm for chirp compressed sensing given in [1] and the closely related algorithm for Reed-Muller compressed sensing given in [21] are not suitable for use with images. The primary contribution of this paper is to describe and demonstrate a reconstruction algorithm that extends the utility of compressed sensing with chirp and Reed-Muller measurement matrices into a regime of less sparsity, thereby supporting imaging applications.

Following this introductory section, the paper continues in Section 2 with a synopsis of background material, focusing on the use of chirps and Reed-Muller sequences in compressed sensing. Background and notation used throughout the remainder of the

*This work was partially supported by NSF-DMS grant #0652833, NSF-DUE #0633033, and ONR-BRC grant #N00014-08-1-1110.

†School of Mathematical and Statistical Sciences, Arizona State University, Tempe AZ 85287-1804 USA

‡Department of Mathematics, University of Idaho, Moscow, ID 83844-3025 USA

§School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe AZ 85287-5706 USA

¶Department of Mathematics, The George Washington University, Washington, DC 20052 USA

paper is introduced in the course of this synopsis. Section 3 is the heart of the paper. It begins by explaining the shortcomings of the reconstruction algorithms of [1] and [21] that manifest when the signal is not highly sparse. It then proceeds to describe the approach taken to extend compressed sensing with chirps and Reed-Muller sequences to support imaging. Appropriate construction of the sensing matrix, with attention to satisfaction of the statistical restricted isometry property (StRIP) [7], is addressed in this section along with a description of the corresponding reconstruction algorithms for both chirp and Reed-Muller measurement matrices. In fact, three variants of the basic algorithms that incorporate various techniques for numerical implementation of the key steps are presented. Section 4 provides analytical and experimental results pertaining to the performance of the method. The paper concludes in Section 5 with a brief discussion of the work presented and some thoughts regarding future directions for this vein of research.

2. Background. The essential elements of compressed sensing are rather well documented in recent research literature. The papers by Baraniuk [2] and by Candès and Wakin [14], for example, present clear synopses of the basics. Consequently, this section provides only a concise description of the general compressed sensing problem to establish notation, and then presents enough background on the deterministic approaches of [1] and [21] to enable subsequent analysis of their shortcomings in imaging applications.

Table 2.1 summarizes notation used throughout the remainder of this paper.

TABLE 2.1
Notation used throughout this paper

N	dimension of the signal space
x	signal N-vector
k	sparsity of x
n	# of measurements (is prime for chirp / a power of 2 for RM)
Φ	$n \times N$ measurement matrix
y	n-vector of measurements
A	$n \times j$ matrix formed from columns from Φ, $0 \leq j \leq k$
z	length-j vector

2.1. Compressed sensing. A signal $x \in \mathbb{C}^N$ is k-sparse in a basis $\mathbf{\Psi} = \{\psi_i\}_{i=1}^N$ if x is a weighted superposition of at most k elements of $\mathbf{\Psi}$. Compressed sensing broadly refers to the inverse problem of reconstructing such a signal x from linear measurements $\{y_j = \langle x, \varphi_j \rangle | j = 1, \ldots n\}$ with $n < N$, ideally with $n \ll N$. Denoting by Ψ the $N \times N$ matrix having the basis elements ψ_i as its columns and by Φ the $n \times N$ matrix having the measurement vectors φ_j as its rows yields

$$y = \Phi x = \Phi \Psi s. \tag{2.1}$$

In this expression, y is an n-vector of measurements and s is an N-vector in which at most k components are nonzero. If Ψ is known, then reconstructing x from y is equivalent to reconstructing s from measurements $\Phi\Psi s$. For this reason, it is common to simplify notation by assuming Ψ is the standard basis and x itself contains no more than k nonzero components. Except in a few places where the sparsifying basis $\mathbf{\Psi}$ is explicitly needed, this assumption will be used throughout the rest of this paper.

Much of the compressed sensing literature is concerned with conditions on the measurement matrix Φ under which the k-sparse assumption on x regularizes, either

with certainty or very high probability, the ill-posed problem of recovering it from y. An important sufficient condition in this regard is the Restricted Isometry Property (RIP) [10], which stipulates that, for some $\varepsilon > 0$,

$$(1-\varepsilon)\|v\|_2^2 \leqslant \|\Phi_{\text{sub}} v\|_2^2 \leqslant (1+\varepsilon)\|v\|_2^2 \,, \qquad \forall v \in \mathbb{C}^k \,, \tag{2.2}$$

must hold for every $n \times k$ matrix Φ_{sub} formed from k distinct columns of Φ. While verifying the RIP property for a given measurement matrix Φ is combinatorially complex, it has been established that certain classes of randomly generated matrices (e.g., whose entries are realized from independent Gaussian random variables) manifest the RIP property with very high probability. One should note that (2.2) is equivalent to the following relation [12] between the minimum and maximum eigenvalues of the Grammian $\Phi_{\text{sub}}^* \Phi_{\text{sub}}$,

$$1-\varepsilon \leq \lambda_{\min}(\Phi_{\text{sub}}^* \Phi_{\text{sub}}) \leq \lambda_{\max}(\Phi_{\text{sub}}^* \Phi_{\text{sub}}) \leq 1+\varepsilon.$$

As noted above, the use of randomly generated matrices has become prevalent in compressed sensing. For a Gaussian measurement matrix, reconstruction of x from y is generally possible if $n > ck\log(N/k)$ [12, 10, 2], with reconstruction entailing a convex ℓ_1 optimization problem, such as

$$\min \|\tilde{x}\|_1 \quad \text{such that} \quad y = \Phi\tilde{x} \,. \tag{2.3}$$

Algorithms to efficiently solve such problems have received much recent attention specifically in connection with their utility in compressed sensing [19, 30, 20, 4]. The popular basis pursuit algorithm [11] has computational complexity $\mathcal{O}(N^3)$; alternatives to basis pursuit (e.g., greedy matching pursuit) also have computational complexities that depend on N.

2.2. Statistical RIP. In [7], Calderbank et al. set forth criteria on Φ that ensure a high probability that the mapping taking the k-sparse signal vector x to the measurement vector y is injective, assuming a specific probability distribution on the unit-magnitude k-sparse vectors in $\mathbb{C}^N$. They say that Φ has the Statistical Restricted Isometry Property (StRIP) with respect to parameters ε and δ if

$$(1-\varepsilon)\|v\|_2^2 \leq \|\Phi v\|_2^2 \leq (1+\varepsilon)\|v\|_2^2 \tag{2.4}$$

holds with probability exceeding 1-δ when v is assumed to be uniformly distributed among k-sparse vectors in $\mathbb{C}^N$ of some fixed norm (e.g., unit norm). They further say that Φ has the Uniqueness-guaranteed Statistical Restricted Isometry Property (UStRIP) with parameters ε and δ if, in addition to having the StRIP property for ε and δ,

$$\{\hat{v} \text{ } k\text{-sparse in } \mathbb{C}^N \;:\; \Phi\hat{v} = \Phi v\} = \{v\}$$

with probability exceeding 1-δ.

The implication of Φ possessing the UStRIP property on recovery of x from $y = \Phi x$ is evident. A further result in [7] provides a set of three conditions that are sufficient to establish that a matrix Φ possesses the UStRIP property. Specifically, suppose

(p1) The rows of Φ are orthogonal and all the row sums are zero;

(p2) The columns of Φ form a group under "pointwise multiplication"; and

(p3) For all $j \in \{2, ..., N\}$, $|\sum_x \phi_j(x)|^2 \leq n^{1-\eta}$.
Then if $k < 1 + (N-1)\varepsilon$ and $\eta > 1/2$, there exists a constant c such that, if $n \geq (c\frac{k \log N}{\varepsilon^2})^{\frac{1}{\eta}}$, Φ possesses the UStRIP property with parameters ε and

$$\delta = 2 \exp\left(-\frac{[\varepsilon - (k-1)/(N-1)]^2 n^\eta}{32k}\right) \tag{2.5}$$

The deterministic compressed sensing matrices proposed in [1, 21] and discussed in Sections 2.3 and 2.4 below satisfy the UStRIP property. In later sections of this paper, the above criteria will be used to show that certain related deterministic matrices also have the UStRIP property.

2.3. Compressed Sensing with Chirps. Applebaum et al. [1] proposed and demonstrated a deterministic compressed sensing scheme using matrices of discrete "chirps;" i.e., frequency modulated discrete sinusoids. Specifically, a discrete chirp of length n with *chirp rate* r and *base frequency* m has the form

$$\phi_{r,m}(\ell) = \frac{1}{\sqrt{n}} e^{\frac{2\pi i}{n} r\ell^2 + \frac{2\pi i}{n} m\ell}, \qquad r, m, \ell \in \mathbb{Z}_n. \tag{2.6}$$

Note that the coefficient $1/\sqrt{n}$ is present in order for the vector to have a unit ℓ_2 norm. For a fixed n, there are n^2 possible pairs (r, m). The full chirp sensing matrix Φ thus has size $n \times n^2$ and can be written as

$$\Phi_{\text{chirp}} = \begin{bmatrix} U_{r_1} & U_{r_2} & \cdots & U_{r_n} \end{bmatrix}. \tag{2.7}$$

Each U_{r_j} is an $n \times n$ matrix with columns given by chirp signals having a fixed chirp rate r_j with m varying from 0 to $n-1$. The chirp rate also varies from 0 to $n-1$. In [1], the suitability of Φ_{chirp} for compressed sensing was demonstrated empirically by comparing the eigenvalues of the Grammians of matrices consisting of k columns chosen uniformly at random from the chirp matrix with those from Gaussian matrices. Φ_{chirp} was shown to possess the UStRIP property in [7]. To do so, each sub-matrix U_{r_j} was multiplied by a unit-magnitude scalar $e^{\frac{2\pi i}{n} r_j}$ to achieve the row-sum condition described in the preceding section.

A key advantage of compressed sensing with chirp matrices is that it admits a fast reconstruction algorithm whose complexity, $\mathcal{O}(kn^2 \log n)$ (see [1]) versus $\mathcal{O}(knN)$ for basis pursuit with matching pursuit, depends only on the number of measurements n and not on the signal length N. This algorithm and a closely related one for reconstruction from compressed measurements made with Reed-Muller matrices are discussed further in Section 2.5.

2.4. Reed-Muller (RM) sensing matrix. In [21], Howard et al. introduced the idea of deterministic compressed sensing using a matrix of second-order Reed-Muller (RM) sequences [24]. The set of such sequences with length 2^m is parameterized by $m \times m$ binary symmetric matrices P and binary m-vectors $b \in \mathbb{Z}_2^m$. In terms of these parameters, a second-order RM function is given by

$$\phi_{P,b}(a) = \frac{1}{\sqrt{2^m}} i^{(2b + Pa)^\intercal a}. \tag{2.8}$$

In this expression, $a \in \mathbb{Z}_2^m$ indexes the 2^m components of the sequence. In analogy with chirps, the vector b in the linear term of this expression and the matrix P

in the quadratic term may be regarded as the "frequency" and "chirp rate" of the sequence, respectively. If P is zero on its main diagonal, then $\phi_{P,b}$ is real-valued. The compressed sensing matrix proposed in [21] has the form

$$\Phi_{\mathrm{RM}} = \begin{bmatrix} U_{P_1} & U_{P_2} & \cdots & U_{P_{2^{m(m-1)/2}}} \end{bmatrix}, \tag{2.9}$$

where each U_{P_j} is a $2^m \times 2^m$ orthogonal matrix whose columns are $\phi_{P_j,b}$ with b going through all binary m-vectors. In addition, each $\phi_{P,b}$ is multiplied with a phase factor $(-1)^{\mathrm{wt}(b)}$, where $\mathrm{wt}(b)$ is the Hamming weight of b, i.e., the number of ones in b. The extra phase factor ensures that the total number of plus and minus signs of the inner products of any two columns are the same. For convenience, P_1 is chosen to be the zero matrix, and therefore, without the phase factor, U_{P_1} is the Hadamard matrix up to a scaling. Consequently, multiplication by U_{P_1} is the Walsh-Hadamard transform which, up to a scaling, is its own inverse. In this paper, the scalings are chosen so that each column in Φ_{RM} has unit norm.

Since there are $2^{m(m-1)/2}$ possible $m \times m$ zero-diagonal binary symmetric matrices, the size of the RM matrix is $2^m \times 2^{m(m+1)/2}$. This is substantially larger than the chirp matrix discussed in the previous section. Defining $N = 2^{m(m+1)/2}$ and $n = 2^m$, a k-sparse signal $x \in \mathbb{C}^N$ yields a measurement $y = \Phi_{\mathrm{RM}} x \in \mathbb{C}^n$, which is the superposition of k RM functions

$$y(a) = z_1 \phi_{P_1,b_1}(a) + z_2 \phi_{P_2,b_2}(a) + \cdots + z_k \phi_{P_k,b_k}(a). \tag{2.10}$$

In (2.10), z_j are used instead of x in order to only write the nonzero terms, and P_j and b_j may individually repeat in the equation.

Comparisons between the condition numbers of the Grammians of randomly selected k-column sub-matrices of the RM matrix and of Gaussian matrices, shown in [21], indicate empirically that the average condition numbers are indistinguishable. This evidence supports their utility as compressed sensing matrices, despite the observation that they almost certainly do not have the RIP property. In [7], compressed sensing matrices formed from RM sequences were shown to have the UStRIP property.

2.5. Quadratic reconstruction algorithm. A key advantage of compressed sensing with chirps and RM sequences is that they admit fast deterministic reconstruction algorithms that perform very well for highly sparse signals. These algorithms, which are closely related, are summarized here to support discussion in Section 3 about how their performance can be improved when the signals to be reconstructed are less sparse.

To recover the sparse signal x from the chirp measurement vector y, Applebaum et al. [1] use a fast Fourier transform (FFT) based algorithm to identify the (r_j, m_j) values corresponding to the locations of the non-zero components of x. The total computational complexity of identifying all k nonzero locations is $\mathcal{O}(kn^2 \log n)$. The magnitudes z_j of the nonzero components are found by solving an associated least-squares problem. In summary, their chirp reconstruction algorithm repeats the following steps. Initially, let $y_0 = y$ denote the residual of the measurement.

Algorithm for signal reconstruction via chirp sensing matrix:

1. For each $T \neq 0$, calculate $w_T =$ the peak location of $\mathrm{FFT}\Big\{\overline{y_0(\ell)} y_0(\ell + T)\Big\}$. Find r_j by solving $w_T = 2r_j T \pmod n$.
2. Let $m_j =$ the peak location of $\mathrm{FFT}\Big\{y_0(\ell) e^{-\frac{2\pi i r_j \ell^2}{n}}\Big\}$.
3. Determine z_l by minimizing $\|y_0(\ell) - \sum_{l=1}^{j} z_l e^{\frac{2\pi i r_l \ell^2}{n} + \frac{2\pi i m_l \ell}{n}}\|_2$

4. Define $y_0(\ell) = y_0(\ell) - \sum_{l=1}^{j} z_l e^{\frac{2\pi i r_l \ell^2}{n} + \frac{2\pi i m_l \ell}{n}}$. Terminate if $\|y_0\|_2$ is sufficiently small.

Step 1 finds the chirp rate associated with the nonzero location having the largest coefficient. Multiplication of a shifted replicate of $y(\ell + T)$ with its conjugate $\overline{y(\ell)}$ results in the chirp rates appearing linearly rather than quadratically in the complex exponentials, and thus chirp rates are mapped to frequency components by the FFT in Step 1. Note that for Step 1 to work properly, n has to be a prime number to uniquely determine r_j. Moreover, all $T = 1, .., n-1$ are used to enhance robustness to noise and cross-term interference that is problematic with less sparse data. Once the chirp rate of the largest peak is found, Step 2 de-chirps the measurement and then applies the FFT to find the frequency associated with the peak. In Step 3, the values of the non-zero coefficients are found by solving a least-squares problem. Finally, the residual is computed before detecting the next largest peak. According to [1], this algorithm is more efficient than matching pursuit with random matrices for sufficiently sparse signals.

With the Reed-Muller sensing matrix, the fast Walsh-Hadamard transform (FHT) is used to detect the nonzero locations, (P_j, b_j) pairs. The total computational complexity of this reconstruction is $O(kn(\log n)^2)$. Initially, let $y_0 = y$ be the residual measurement. In this case, the quadratic reconstruction algorithm repeats the following steps until the residual y_0 is small:

Algorithm for signal reconstruction via RM sensing matrix:

1. For each e_i, $i = 1, .., m$, calculate $w_i =$ peak location of $\mathrm{FHT}\Big\{\overline{y_0(a)} y_0(a+e_i)\Big\}$. The i^{th} column of P_j is the w_i^{th} element in $\mathbb{Z}_2^m$.
2. Let $b_j =$ the w^{th} element in $\mathbb{Z}_2^m$, where $w =$ the peak location of $\mathrm{FHT}\Big\{y_0(a)\overline{\phi_{P_j,0}(a)}\Big\}$.
3. Determine z_l by minimizing $\|y_0(a) - \sum_{l=1}^{j} z_l \phi_{P_j,b_j}(a)\|_2$.
4. Define $y_0(a) = y_0(a) - \sum_{l=1}^{j} z_l \phi_{P_j,b_j}(a)$. Terminate if $\|y_0\|_2$ is sufficiently small.

The idea of this algorithm is similar to the one for chirps. The most intricate part is to form the P matrix in the first step, which is a scalar in the case of the chirp matrix. For reconstructing sparse signals, in terms of reconstruction speed and fidelity, this method is also more efficient than matching pursuit with random matrices, see [7].

3. Approach. Despite the success for accurate reconstruction of very sparse one-dimensional (1D) signals with the algorithm described in Section 2.5, application of it to real two-dimensional (2D) images is impractical. This is because, in general, real images are not as sparse in any transform domain as the one-dimensional signals used in [1] or [21]. For instance, for good reconstruction of a chirp signal of length $N = 67^2$ using $n = 67$ measurements, the sparsity is around $k = 8$. The sparsity for successful reconstruction of an RM signal of length $N = 2^{55}$ with $n = 2^{10}$ measurements is around $k = 20$. If sparsity is much larger than considered in these examples, then reconstruction is not guaranteed and the errors become very large.

A good approximation of a 256×256 pixel image is typically obtained by retaining the largest 10% wavelet coefficients in some suitably chosen wavelet domain. In particular, many medical images are well approximated by transform coding using 10–20% of their wavelet coefficients, but begin to show appreciable degradation as the percentage of coefficients retained falls below these levels. However, a 256×256 image with 10% sparsity has $6,554$ nonzero coefficients, which is much larger than the sparsity considered for the 1D signals in [1] and [21]. A rule of thumb, see [3, The-

orem 1], for the number of measurements in the standard compressed sensing using the Gaussian random matrices with ℓ_1 minimization is given by

$$n > k \log_2(1 + N/k). \tag{3.1}$$

This rule guarantees successful reconstruction with high probability if the number of measurements n is large compared to the sparsity and signal size. Using (3.1) in the above example, at least 22,672 measurements are needed for the correct reconstruction. The ratio N/n is 2.89, and this implies that roughly only three chirp rates or P matrices are needed to form the sensing matrix. Therefore, the efficiency of finding nonzero locations and their coefficients using quadratic reconstruction algorithm is not fully utilized.

This section continues by describing the construction of sensing matrices adapted to the 2D case in Section 3.1. Section 3.2 shows that these matrices satisfy the UStRIP property. Section 3.3 explains the first of the improvements introduced in this paper: detection of the 'bulk' of a signal in the first step of the approximation. This step is termed "initial best approximation," since it precedes the iteration process entailed in the two algorithms summarized in Section 2.5. It is both fast and valuable to make an initial approximation that resembles a low-pass filtered version of the signal. The second new feature introduced here detects several (largest) peaks at once, thus significantly decreasing processing time. This is described in Section 3.4. A third improvement is in the least-squares step and is described in Sections 3.5 and 3.6. Finally, it is demonstrated how these new features are combined in the numerical implementation and compare the improvement of each step in Section 3.7.

3.1. Construction of an effective sensing matrix for image reconstruction. As explained above, due to the nature of sparsity of images and the rule of thumb (3.1), a few sub-matrices of Φ_{chirp} or Φ_{RM} can be used as the sensing matrix, with the ratio $N/n = 2.89$ for 10%-sparse images. In practice, a larger ratio can be used, such as 4 for 10%-sparse images, which will be analyzed later. Consequently, there is more freedom in the choice of the chirp rates, r or P matrices, when constructing the sensing matrices.

Construction of the chirp sensing matrix. The inner product of any pair of distinct chirp vectors is as follows:

$$\left|\left\langle \phi_{r_i,m_j}, \phi_{r_{i'},m_{j'}} \right\rangle\right| = \begin{cases} \frac{1}{\sqrt{n}}, & \text{if } r_i \neq r_{i'} \\ 0, & \text{if } r_i = r_{i'} \text{ and } m_j \neq m_{j'}. \end{cases} \tag{3.2}$$

Therefore, a submatrix should use as few chirp rates as possible and the choice of the chirp rates can be arbitrary. For example, the submatrix can be

$$\Phi_{\text{chirp}} = \begin{bmatrix} U_{r_1} & U_{r_2} & U_{r_3} & \widetilde{U}_{r_4} \end{bmatrix}, \tag{3.3}$$

where $r_1 = 1, r_2 = 2, r_3 = 3$, $r_4 = 4$, and $\widetilde{U}_{r_4}$ is a submatrix of U_{r_4} so that the column number of Φ_{chirp} matches the signal size. For instance, the sensing matrix for an 256×256 ($N = 65536$) image may be taken to be of size $n \times N = 16411 \times 65536$. Note that 16411 is the next prime number larger than 25% of 65536. Here, the 25% ratio comes from the fact that four chirp rates are used.

Construction of the RM sensing matrix. When forming a sensing matrix from submatrices of the RM matrix, the choice of the submatrix cannot be arbitrary. From

[24], the inner product of two columns of Φ_{RM}, one taken from U_{P_i} and another from $U_{P_{i'}}$, $i \neq i'$, is given by

$$\left|\left\langle \phi_{P_i,b_j}, \phi_{P_{i'},b_{j'}} \right\rangle\right| = \begin{cases} \frac{1}{\sqrt{2^l}}, & 2^l \text{ times}, \\ 0, & 2^m - 2^l \text{ times}, \end{cases} \tag{3.4}$$

where $l = \mathrm{rank}(P_i - P_{i'})$. If $l = m$, the inner product is always $1/\sqrt{2^m}$, which is smaller than the inner product in other cases, $l < m$. Since the nonzero locations of the signal are unknown, it is desirable that the inner products between any two columns are as small as possible, thus making the columns of the resulting sensing matrix close to orthogonal. Taking $l = m$ gives the best situation. The set of zero-diagonal symmetric binary matrices such that the rank between any two matrices is equal to m is called the Kerdock set, see [22]. There are 2^{m-1} elements in the Kerdock set, and the construction of the Kerdock set can be found, for example, in [6]. For example, a sensing matrix can be constructed in the form

$$\Phi_{\mathrm{RM}} = \begin{bmatrix} U_{P_1} & U_{P_2} & U_{P_3} & U_{P_4} \end{bmatrix}, \tag{3.5}$$

where P_1, P_2, P_3, and P_4 are any matrices from the Kerdock set. For example, the sensing matrix for a 256×256 ($N = 2^{16}$) image with 10% sparsity is of size $n \times N = 2^{14} \times 2^{16}$ which means that only 25% of the signal entries are sampled. Note that for images with sparsity much smaller than 10%, fewer measurements are needed, and therefore, more chirp rates or P matrices can be used, since the ratio N/n becomes larger.

3.2. UStRIP property. This section presents a proof that the chirp and RM submatrices in Section 3.1 satisfy the UStRIP property. This is achieved by modifying the proof of Theorem 2.3 in [7], which guarantees the UStRIP property for a class of deterministic matrices. The proof of UStRIP for these matrices proceeds in two steps. First, the expectation of $\|\Phi x\|^2$ is shown to be close to $\|x\|^2$. Then, a modification of McDiarmid's inequality gives an upper bound on the probability that a multivariate function deviates within a fixed range from its expected value. The random variables of the multivariate function in the modified McDiarmid's inequality are distinct instead of independent because they are associated with the nonzero locations, which do not repeat. In [7], the proof for deterministic matrices satisfying the sufficient condition of the modified McDiarmid's inequality only relies on property **(p3)** in Section 2.2 and $k \ll N$, thus, can be extended to submatrices. For the first part, the following Lemma is shown in [7]:

LEMMA 3.1. *For $\pi =: (\pi_j)_{j\in\{1,\dots,N\}}$ a random permutation,*

$$\left(1 - \frac{k-1}{N-1}\right)\|x\|_2^2 \leq \mathbb{E}_\pi\left[\|\Phi x\|_2^2\right] \leq \left(1 + \frac{1}{N-1}\right)\|x\|_2^2. \tag{3.6}$$

The proof is based on **(p1-p3)** in Section 2.2.

For the submatrices used in this paper, **(p1)** and **(p2)** are not satisfied, and thus, (3.6) is not directly implied. Therefore, what is needed is a similar inequality for submatrices. As discussed in [7], a submatrix constructed by randomly choosing columns from the full deterministic matrix can be used in compressed sensing for signals with large sparsity. However, this does not guarantee that the submatrix satisfies the UStRIP property and the associated reconstruction algorithm may not be optimally efficient. Instead, the submatrices used in this paper are of the form

(3.3) and (3.5), which use a few chirp rates r or P. Denote the submatrix as $\Phi = \begin{bmatrix} U_1 & -U_2 & U_3 & -U_4 \end{bmatrix}$, where presence of the extra minus signs (or phase factors) ensure that the row sums of Φ equal zero. Then, the expectation $\mathbb{E}_\pi[\|\Phi x\|_2^2]$ based on this choice of the phase factors is shown in the following:

LEMMA 3.2. *For the submatrix defined above,* $\Phi = \begin{bmatrix} U_1 & -U_2 & U_3 & -U_4 \end{bmatrix}$,

- $\sum_\ell \phi_j(\ell) = 0, \quad \forall\ 1 \le j \le n$
- $\mathbb{E}_\pi \left[\sum_\ell \langle \phi_i(\ell) \mid \phi_j(\ell)\rangle\right] = -\frac{1}{\sqrt{n}} \frac{n}{N-1}, \ \forall\ i \ne j.$

Proof. The row sum of U_1 is zero for each row except for the first row because all entries in the first row are $1/\sqrt{n}$ by construction. Consequently, the row sums of U_j, for $j = 2, 3, 4$, are also zero except for the first row, in which all entries are $1/\sqrt{n}$ as well. Therefore, with the phase factors on U_2 and U_4, all row sums of Φ are equal to zero.

The next step is to find the expected inner product between any pair ϕ_i and ϕ_j with $i \ne j$. For each column ϕ_j from U_1, the collection formed by pointwise multiplication with the complex conjugate of all other columns of Φ except with itself is $\tilde{\Phi} = \begin{bmatrix} \tilde{U}_1 & -U_a & U_b & -U_c \end{bmatrix}$ for some chirp rates a, b, and c. $\tilde{U}_1$ is the same as U_1 but without the first column. Even though a, b, and c may not come from chirp rates of Φ, they do exist because of **(p2)**. Therefore, the row sum for each row of $\tilde{\Phi}$ is $-1/\sqrt{n}$. Similarly, for each column from U_2, U_3, or U_4, the row sum of the collection of the inner products is also $-1/\sqrt{n}$. Therefore,

$$\mathbb{E}_\pi \left[\sum_\ell \langle \phi_i(\ell) \mid \phi_j(\ell)\rangle\right] = \frac{N\frac{-1}{\sqrt{n}}}{N(N-1)} = -\frac{1}{\sqrt{n}} \frac{n}{N-1}, \tag{3.7}$$

because there are $4n = N$ columns in Φ and the total number of inner products is $4n(4n-1) = N(N-1)$. ∎

Lemma 3.1 can now be proved using Lemma 3.2, and thus the following result holds for submatrices:

THEOREM 3.3. *If a submatrix has the form* $\Phi = \begin{bmatrix} c_1 U_1 & c_2 U_2 & \cdots & c_j U_j \end{bmatrix}$, *where* $c_1, \ldots c_j$ *are real or complex-valued scalars with* $|c_j| = 1$ *and* $c_1 + \ldots + c_j = 0$, *then* Φ *satisfies* (k, ε, δ)*-UStRIP, with* δ *described in (2.5).*

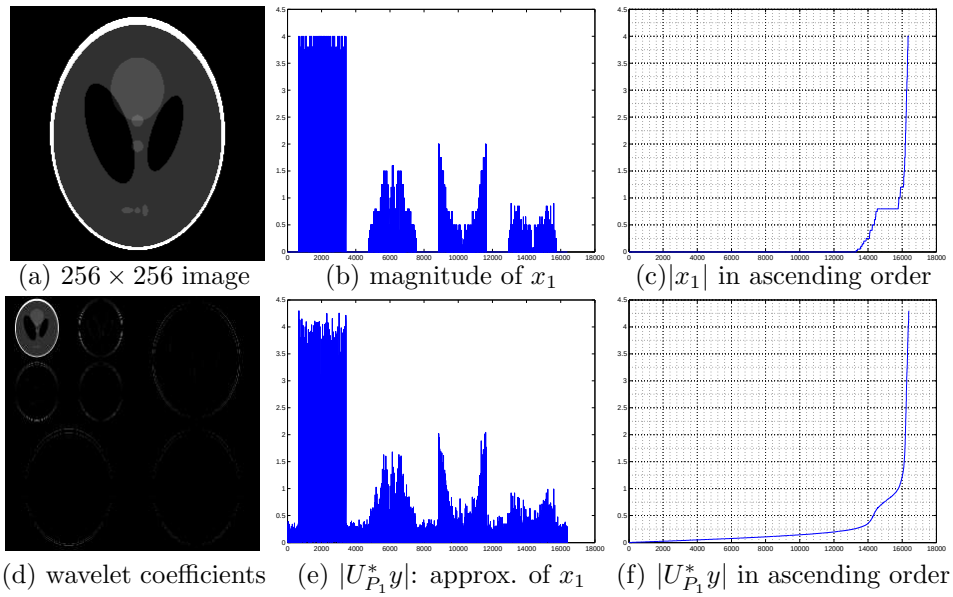
3.3. Initial best approximation. This section proposes a new approach to detect a large portion of the nonzero locations in one step. The approach is based on the observation that many important classes of images, including medical images, are low-pass in nature. Consequently, most of their energy is captured by the coarse-scale wavelet coefficients (i.e., in the upper-left region of a multiscale analysis diagram). This general structure can be exploited without more explicit *a priori* knowledge of individual images. Although the idea is explained here in the context of the RM sensing matrix, the same concept works for the chirp sensing matrix as well. The measurements can be written as

$$\Phi x = \begin{bmatrix} U_{P_1} & U_{P_2} & U_{P_3} & U_{P_4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = U_{P_1} x_1 + U_{P_2} x_2 + U_{P_3} x_3 + U_{P_4} x_4, \tag{3.8}$$

where x_1, x_2, x_3, and x_4 are vectors of the upper-left, lower-left, upper-right, and lower-right coefficients, respectively. It is possible to estimate x_1 by

$$U_{P_1}^* y = x_1 + U_{P_1}^* U_{P_2} x_2 + U_{P_1}^* U_{P_3} x_3 + U_{P_1}^* U_{P_4} x_4, \tag{3.9}$$

where * denotes the conjugate transpose. The last three terms are small because x_2, x_3, and x_4 are much sparser and smaller than x_1, and furthermore $U_{P_1}^* U_{P_2}, U_{P_1}^* U_{P_3}$, and $U_{P_1}^* U_{P_4}$ are small as well, as discussed in Section 3.1. Therefore, $U_{P_1}^* y \approx x_1$. In the case when all nonzero locations are in the upper-left region, i.e., x_2, x_3, and x_4 are zero, $U_{P_1}^* y$ is equal to x_1, which automatically completes the image reconstruction.



(a) 256×256 image (b) magnitude of x_1 (c) $|x_1|$ in ascending order

(d) wavelet coefficients (e) $|U_{P_1}^* y|$: approx. of x_1 (f) $|U_{P_1}^* y|$ in ascending order

FIG. 3.1. *(a) is the standard Shepp-Logan image and (d) demonstrates that the energy of wavelet coefficients is concentrated in the upper-left region; (b) shows the absolute value of x_1 entries and (c) is the plot of $|x_1|$ sorted in the ascending order; similarly, (e) shows the magnitude of $|U_{P_1}^* y|$ and then (f) orders $|U_{P_1}^* y|$ in the ascending order; comparing (c) and (f), it follows that $|U_{P_1}^* y|$ approximates $|x_1|$ well.*

Figure 3.1 shows an example of this method with RM sensing matrix, where (a) is the Shepp-Logan phantom image of pixel resolution 256×256 and (d) shows its Haar wavelet coefficients. From (b) and (e), it is evident that $U_{P_1}^* y$ is a good approximation of x_1 (with added cross terms in $U_{P_1}^* y$ from the last three terms of (3.9)). The nonzero locations in x_1 are then found by thresholding.

The threshold is determined by the following method. First, $|U_{P_1}^* y|$ is sorted in ascending order, as shown in (f). Observe that the graph in (f) is a smooth version of the graph in (c). The threshold is picked as the critical point[1] closest to the origin. In (f), such a critical point is around 14000. Denoting the detected locations by $(P_l, b_l), l = 1, ..., j$, the magnitudes z_l can be well estimated by solving the least-squares problem, $\min_z \|Az - y\|$, where A is the matrix whose columns consist of ϕ_{P_l, b_l} and $z = [z_1, .., z_j]^T$. The solution is $z_{sol} = (A^*A)^{-1}A^*y$. Since $P_l = P_1$ in the initial step and U_{P_1} is orthogonal, the solution is $z = A^*y$. This is the *initial best approximation.*

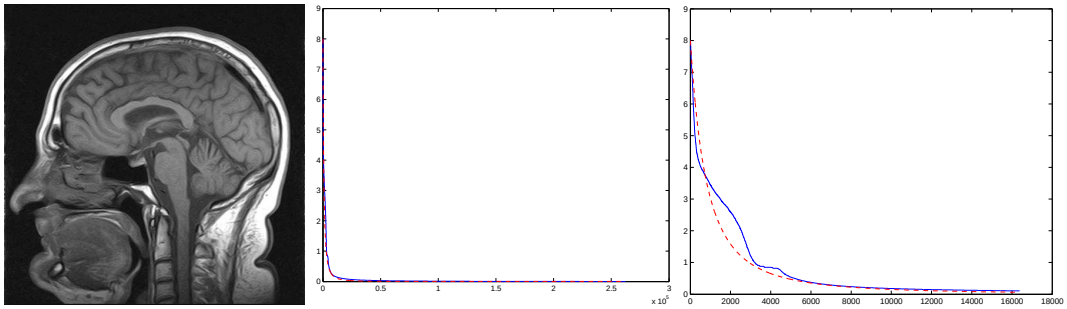
This method entails low computational cost, since it only requires one length-n FFT or FHT and sorting of one n-vector. Empirically, the initial best approximation

[1] In the continuous case it would be exactly the critical point, i.e., where the derivative does not exist; however, this is a discrete setting and so the term "critical point" is used loosely.

step already gives small reconstruction error, around -18 dB, where the error is defined as:

$$\text{Error(dB)} = 10\log_{10}\left[\frac{||x_{\text{actual}} - x_{\text{reconstructed}}||^2}{||x_{\text{actual}}||^2}\right]. \tag{3.10}$$

Note that the negative of the above error is known as the signal-to-noise ratio (SNR).



(a) $N = 512 \times 512$ (b) decay plot (c) decay plot zoom-in

FIG. 3.2. *(a) is the initial MRI image; in (b) the blue solid curve represents the absolute values of the wavelet coefficients of image (a) in the descending order. The red dashed curve approximates the blue curve quite well and is (3.11) with* $s = 2, D = 400\frac{N}{2^{16}}$, *and Const = maximum absolute values of wavelet coefficients.*

Error Estimation. To estimate the error of the initial approximation method, the coefficients of x are arranged in decreasing order, $|x_{(1)}| \geq |x_{(2)}| \geq ... \geq |x_{(N)}|$. Assume that the wavelet coefficients have the following decay property (this is consistent, for example, with [18, 13]):

$$|x_{(n)}| \approx \text{Const} \cdot (n + D)^{-s}, \tag{3.11}$$

for some $s \geq 1$ and $D > 0$. The approximation of $\|x\|$ is given by

$$\|x\|_2 = \left\|\sum_{n=1}^{N} x_{(n)}\phi_{(n)}\right\|_2 = \sqrt{\left|\sum_{n=1}^{N} x_{(n)}\phi_{(n)}\right|^2} \approx \sqrt{\sum_{n=1}^{N}\left|x_{(n)}\right|^2} \approx \text{Const}\cdot\sqrt{\sum_{n=1}^{N}(n+D)^{-2s}}. \tag{3.12}$$

If the largest M coefficients are detected, the error between the initial best approximation $x_M = \sum_{n=1}^{M} x_{(n)}\phi_{(n)}$ and the actual x is

$$\|x_M - x\|_2 = \left\|\sum_{n=M+1}^{N} x_n\phi_{(n)}\right\|_2 \approx \text{Const} \cdot \sqrt{\sum_{n=M+1}^{N}(n+D)^{-2s}}. \tag{3.13}$$

Therefore, the error of this initial approximation method is about

$$\text{Error}_{\text{init}} \approx 10\log_{10}\left[\frac{\sum_{n=M+1}^{N}(n+D)^{-2s}}{\sum_{n=1}^{N}(n+D)^{-2s}}\right]. \tag{3.14}$$

For example, let N be the image size and suppose that the largest 2% coefficients are detected, i.e., $M = 2\%$ of N. The wavelet coefficients of several images are described by the decay (3.11) with $s = 2$ and $D = 400\frac{N}{2^{16}}$. Then for $N = 256^2, 512^2$, and 1024^2 the calculation of (3.14) gives the initial error

$$\text{Error}_{\text{init}} \approx -18.9 \text{ dB}.$$

3.4. Detecting nonzero locations with DCFT and DCHT. The discrete chirp Fourier transform (DCFT) of an n-point signal x, defined in [31], can be written as

$$X_c(r,m) = \frac{1}{\sqrt{n}} \sum_{\ell=0}^{n-1} x(\ell)(W_n)^{r\ell^2+m\ell}, \tag{3.15}$$

where $W_n = e^{-2\pi i/n}$. In the above definition, r and m are the chirp rate and frequency, respectively. For a fixed r_j,

$$X_c(r_j,m) = \frac{1}{\sqrt{n}} \sum_{\ell=0}^{n-1} x(\ell)(W_n)^{r_j\ell^2+m\ell}. \tag{3.16}$$

Defining

$$x_{r_j}(\ell) = x(\ell)(W_n)^{r_j\ell^2}, \tag{3.17}$$

yields

$$X_c(r_j,m) = \text{DFT}\left\{x_{r_j}(\ell)\right\}, \qquad j = 1,2,3,4. \tag{3.18}$$

Note that only four n-point DFTs need to be evaluated, since only four chirp rates are used. Of the $4n$ DCFT coefficients computed, the ones with largest absolute values are chosen. The corresponding (r,m) pairs from the DCFT plane (see Figure 3.3) are used to recover the chirp sensing matrix columns. In this setting, applying the DCFT to detect nonzero locations is the same as de-chirping with all four chirp rates and then applying the DFT

$$w(r_j,\ell) = \underset{n}{\text{DFT}}\left\{y_0(\ell)\overline{\Phi_{r_j,0}(\ell)}\right\}, \qquad j = 1,2,3,4. \tag{3.19}$$

The first d largest coefficients $|w(r_j,\ell)|$ are then selected, giving d pairs (r_j,ℓ). In the experiments presented in Section 4.2, $d \approx 100$.

This selection procedure can be visualized as follows. The DCFT plane is an $n \times n$ plane for an n-point signal. For the Shepp-Logan phantom image of 64×64 size and 1% sparsity the DCFT plane is shown in Figure 3.3. The chirp rate axis has only four points per frequency cycle due to the selection of only four chirp matrices (in the example, the rates are 0, 400, 600, 800) while the chirp frequency axis is dense due to the true non-zero coefficients from the sparse signal as well as from the side-lobes introduced during the de-chirping step.

3.5. Updated pseudo inverse solution. The third step finds the values z_j by solving the linear least-squares problem:

$$\min_z \|Az - y\| , \tag{3.20}$$

where A is a submatrix of the sensing matrix and $z = [z_1,..,z_j]^{\mathsf{T}}$ is a vector. Two methods are proposed for this step. This section describes the first method, while the second method is discussed in the next subsection.

Note that the matrix A in the current step can be expressed as $A = [\widetilde{A} \quad c]$, where $\widetilde{A}$ is the matrix in the previous least-squares problem and c is the newly found column. To solve these least-squares problems without treating each problem (iteration)

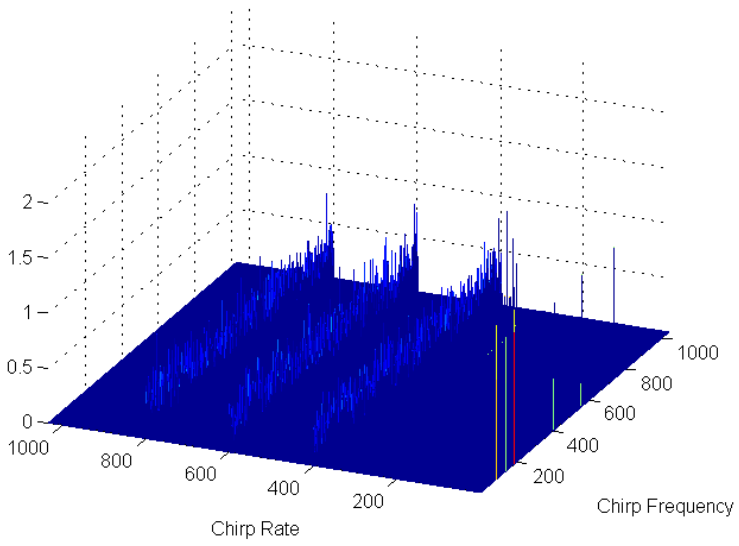


FIG. 3.3. *DCFT plane: chirp rate r_j vs. chirp frequency. Here, only 4 rates in X_c from* (3.18) *are relevant: $r_1 = 0, r_2 = 200, r_3 = 400, r_4 = 800$, the height here is $|X_c|$.*

independently, it is possible to use an updated pseudo-inverse solution method whose computation is based on previous calculations. The pseudo-inverse solution of (3.20) is

$$z_{sol} = (A^*A)^{-1}A^*y \; . \tag{3.21}$$

The inverse of

$$A^*A = \begin{bmatrix} \widetilde{A}^*\widetilde{A} & \widetilde{A}^*c \\ c^*\widetilde{A} & c^*c \end{bmatrix} \tag{3.22}$$

can be computed efficiently by the Schur-Banachiewicz blockwise inversion formula [5]:

$$\begin{bmatrix} D & E \\ F & G \end{bmatrix}^{-1} = \begin{bmatrix} D^{-1} + D^{-1}EVFD^{-1} & -D^{-1}EV \\ -VFD^{-1} & V \end{bmatrix}, \tag{3.23}$$

where $V = (G - FD^{-1}E)^{-1}$. Since $D^{-1} = (\widetilde{A}^*\widetilde{A})^{-1}$ is known from the previous iteration and the size of $V = (G - FD^{-1}E)^{-1}$ is small, the calculation is very efficient. The calculation of A^*y can be done using previous steps by

$$A^*y = \begin{bmatrix} \widetilde{A}^*y \\ c^*y \end{bmatrix}, \tag{3.24}$$

where the size of c is much smaller than the size of A.

3.6. Fast methods for DCFT and DCHT with an arbitrary subset of columns. The second method for solving the least-squares problem (3.20) is by the LSQR algorithm of Paige and Saunders [27], replacing the above pseudo-inverse solution approach. The LSQR algorithm is an efficient iterative method for solving the least-squares problem if the matrix A is sparse or the matrix-vector multiplications by A and A^* can be done efficiently. Therefore, this section begins by showing how to efficiently apply Φ and Φ^* to length-N and length-n vectors, respectively, without directly calculating matrix-vector products. Subsequently, an efficient method for multiplying A and A^* is described, where A is an arbitrary submatrix of Φ, formed by concatenating a set of columns of Φ.

First, write the matrix multiplication Φx as the blockwise matrix multiplications:

$$\Phi x = \begin{bmatrix} U_1 & U_2 & U_3 & U_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{3.25}$$

Recall that U_1 is the discrete inverse Fourier or Walsh-Hadamard transform, up to a scaling. A key step here is to express U_j for $j \neq 1$ as the product of a diagonal matrix D_{v_j} with diagonal $v_j = \phi_{r_j,0}$ or $v_j = \phi_{P_j,0}$ and matrix U_1 as follows:

$$U_j = D_{v_j} U_1. \tag{3.26}$$

Then,

$$\left(D_{v_j} U_1\right) x_j = D_{v_j} \left(U_1 x_j\right), \tag{3.27}$$

Therefore, instead of directly multiplying the matrix U_j with x_j, one can apply the inverse fast Fourier (or Walsh-Hadamard) transform after pointwise vector-vector multiplication of v_j and x_j. So, Φx in (3.25) can be calculated with only four FFTs or FHTs.

The calculation of Az can be efficiently carried out by a slight modification of the above method for Φx. Consider a length-N zero vector x_z. The coefficients of z are inserted at those locations in x_z that correspond to the columns of Φ that make up the matrix A. Then, $Az = \Phi x_z$ can be efficiently calculated as described above.

For efficient calculation of $\Phi^* y$, one can start by expressing it as a blockwise matrix multiplication:

$$\Phi^* y = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \begin{bmatrix} y \end{bmatrix} = \begin{bmatrix} U_1^* y \\ U_2^* y \\ U_3^* y \\ U_4^* y \end{bmatrix} \tag{3.28}$$

Therefore, $\Phi^* y$ can be obtained by concatenating four vectors $U_j^* y$, for $j = 1, 2, 3, 4$, into a long vector. The matrix U_1^* is the Fourier matrix or the Walsh-Hadamard matrix. Each U_j^* can be written as the product of U_1^* and $D_{\overline{v_j}}$, where $D_{\overline{v_j}}$ is a diagonal matrix with diagonal $\overline{v_j}$:

$$U_j^* = U_1^* D_{\overline{v_j}}\,. \tag{3.29}$$

Plugging in this expression for $U_j^* y$, the calculation can be carried out by first applying the Fourier or Walsh-Hadamard transform and then the pointwise vector-vector multiplication:

$$U_j^* y = \left(U_1^* D_{\overline{v_j}}\right) y = U_1^* \left(D_{\overline{v_j}} y\right). \tag{3.30}$$

Finally, the calculation of $A^* y$ can be efficiently carried out by a slight modification of the method for $\Phi^* y$. First, calculate $\Phi^* y$ as above and get a length-N vector, say x. Then, from x, only choose the positions that have corresponding columns in A to form a length-j vector, x_A. This is because A^* is a submatrix of Φ^* with selected rows from Φ^*. Therefore, $A^* y$ can also be efficiently calculated.

Using these efficient methods, the least-squares problem (3.20) may be solved via a LSQR algorithm instead of the updated pseudo-inverse solutions method of Section 3.5.

3.7. Three versions of the reconstruction algorithm. This section describes how the features introduced in the preceding sections are incorporated into the image reconstruction algorithm. The reconstruction algorithm is built up step-by-step with the most efficient reconstruction (called version 3, or v3) with the goal of comparing how each modification improves the reconstruction.

Version one (v1) consists of three iterative steps: detecting the non-zero locations, finding the corresponding coefficients and getting the residuals. These steps are repeated until the residual is sufficiently small. Specifically, nonzero location detection is done using DCFT or DCHT as described in Section 3.4 instead of using the shift-and-multiply method and then applying FFT or FHT ([1, 21]). Since only a few chirp rates are used, multiple nonzero locations can be found in each iteration of the algorithm. Then, v1 uses the updated least-squares method with the pseudo inverse solution, as described in Section 3.5, to find the coefficients without treating each iteration independently. This method was applied with the chirp sensing matrix in [26].

In version two (v2), the initial approximation step explained in Section 3.3, is added before the iterative steps to detect a significant number of the nonzero coefficients. This was applied with the Reed-Muller matrix in [25]. The additional approximation step speeds up the algorithm and, more importantly, the reconstruction fidelity is improved when the image sparsity is increased.

The version three (v3) also uses the initial approximation step. The improvement largely comes from the reconstruction time and storage in the second iterative step. Now the coefficients are found by using the fast DCFT or DCHT method introduced in Section 3.6 along with the LSQR algorithm for solving the least-squares problem.

The comparison is given in Table 3.1 with the following steps in each version:

- Step 0: approximation of the solution
- Step 1: detecting of nonzero locations
- Step 2: finding the coefficients using least squares
- Step 3: getting the residual and repeating steps 1 & 2 if the residual is not sufficiently small

4. Performance Analysis and Experiments. In this section, the performance of the approach described in the preceding section is analyzed in terms of computational complexity and memory use. Results from a number of experiments involving both natural scene and medical images are also summarized.

TABLE 3.1
Three versions of algorithms

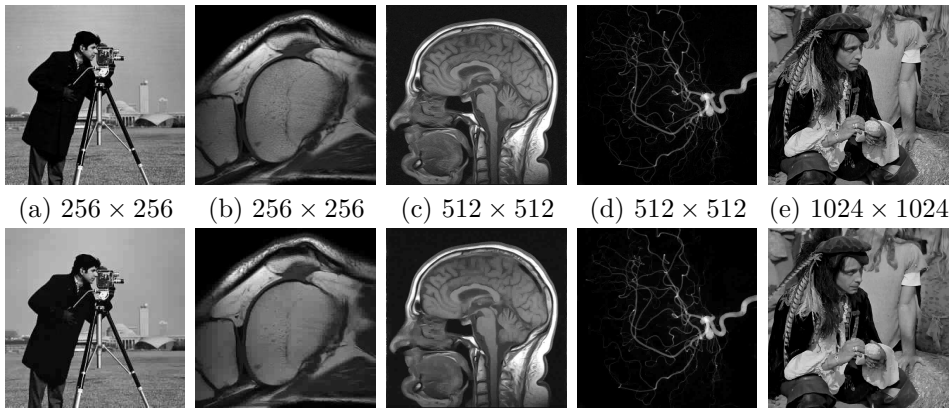
	Step 0: Approx.	Step 1: Detection	Step 2: Least squares
v1	no	DCFT or DCHT Section 3.4	Updated pseudo inverse solutions Section 3.5
v2	Sec. 3.2	DCFT or DCHT Sec. 3.4	Updated pseudo inverse solutions Sec. 3.5
v3	Sec. 3.2	DCFT or DCHT Sec. 3.4	LSQR with fast DCFT or DCHT Sec. 3.6

4.1. Computational complexity and memory. The chirp quadratic reconstruction algorithm (QRA) detects the nonzero locations one-by-one with $n-1$ length-n FFTs. Therefore, the total computational complexity for Step 1 is $\mathcal{O}(kn^2 \log n)$. For RM QRA, m $(= \log n)$ length-n FHTs are used for nonzero location detection. The computational complexity is thus $\mathcal{O}(kn(\log n)^2)$. For QRA with both chirp and RM, Step 2 uses a pseudo-inverse method to solve the least-squares problem. Specifically, since the matrix A is of size $n \times j$, $n > j$, this is usually done with the QR factorization. Therefore, the overall computational complexity is $\mathcal{O}(\sum_{j=1}^{k} j^3) = \mathcal{O}(k^4)$. Therefore, if the sparsity k is significantly small (i.e., $k \ll n \ll N$), this method is very efficient. However, when k is not so small, which is usually the case for real images, it becomes inefficient. For chirp and RM algorithms v1, v2, and v3, Step 1 uses only a few (for example, 4) length-n FFTs or FHTs to detect d nonzero locations in one iteration. Therefore, the total computational complexity is $\mathcal{O}(\frac{1}{d}kn \log n)$. In all the experiments presented here, $d = 100$. For chirp and RM, in both v1 and v2, Step 2 uses the updated pseudo-inverse method to find the nonzero coefficients, wherein multiplications with the matrix A of size $n \times j$ is carried out. Therefore, the overall computational complexity is $\mathcal{O}(\sum_{j=1}^{k} nj) = \mathcal{O}(nk^2)$. The memory is $\mathcal{O}(k^2)$ because $(A^*A)^{-1}$ needs to be stored and has size $j \times j$. In Step 2 of chirp and RM for v3, because of the use of the fast DCFT or DCHT algorithms that need a few length-n FFTs or FHTs, the total computational complexity is $\mathcal{O}(tkn \log n)$, where t is the iteration number of convergence. The memory cost in this case is only $\mathcal{O}(k)$ because only the j chirp-frequency pairs related to the matrix A need to be stored. The comparison is given in Table 4.1.

TABLE 4.1
Computational complexity of each step for different versions of the algorithm

		chirp QRA	RM QRA	Chirp/RM v1/v2	Chirp/RM v3
Complexity	Step 1 Step 2	$\mathcal{O}(kn^2 \log n)$ $\mathcal{O}(k^4)$	$\mathcal{O}(kn(\log n)^2)$ $\mathcal{O}(k^4)$	$\mathcal{O}(\frac{1}{d}kn \log n)$ $\mathcal{O}(nk^2)$	$\mathcal{O}(\frac{1}{d}kn \log n)$ $\mathcal{O}(tkn \log n)$
Memory	Step 2	$\mathcal{O}(nk)$	$\mathcal{O}(nk)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k)$

4.2. Experimental Results and Comparisons. For the experiments, each original image was sparsified by computing its Haar wavelet transform and then retaining a pre-determined fraction of its wavelet coefficients, keeping the largest coefficients and setting the rest to zero. The image data were compressively sensed (measured) with chirp, Reed-Muller, and noiselet matrices of the same size (for chirp, the closest prime number) and then reconstructed by their respective algorithms. Figure 4.1 show the original images with sizes ranging from 256×256 to 1024×1024 and their corresponding sparsified images with the pre-determined sparsity.



(a) 256×256 (b) 256×256 (c) 512×512 (d) 512×512 (e) 1024×1024

(f) 14% (g) 10% (h) 7% (i) 5% (j) 3%

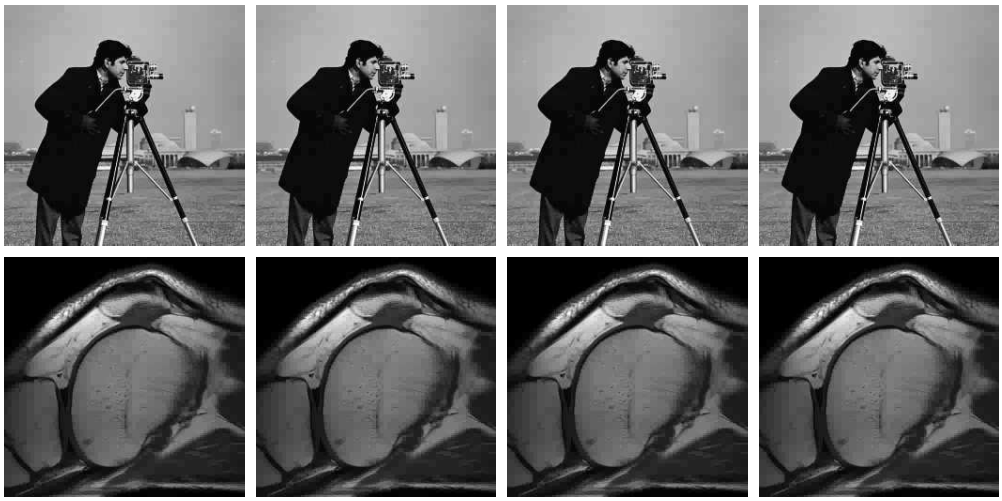
FIG. 4.1. *Top row shows original images. Bottom row shows the sparsified images.*

Table 4.2 shows the reconstruction errors and times for all three versions of the chirp reconstruction algorithm. Note that the experiments were done in Matlab and the codes are not optimized, so run-time comparisons should be interpreted accordingly. For reconstruction fidelity, v2 is slightly better than v1. Table 4.3 shows the reconstruction errors and times for all three versions of the RM reconstruction algorithm. For reconstruction fidelity, v2 is much better than v1 because the initial approximation step is able to correctly detect a large number of nonzero locations. Among all experiments performed (including many not shown in the tables), the highest sparsity level for which chirp v2 and RM v2 were able to accurately reconstruct is larger than that for chirp v1 and RM v1. In terms of reconstruction time, v3 is better than v2 because v3 uses the fast DCFT or DCHT method to find the coefficients. In the table, s, m, and h stand for seconds, minutes, and hours, respectively. The reconstruction results by chirp v3 and RM v3 are better than the other versions. The error is about -110 dB, which is larger than the error for v2 because v2 directly computes the closed-form solution of the least-squares problem and v3 approaches the solution iteratively. The reconstructed images by chirp v1, v2, and v3 in Table 4.2 is shown in Figure 4.2, in which all the reconstructed images look essentially identical to the reference images.

Table 4.4 and Figure 4.3 show experimental results obtained with chirp v3 and RM v3 algorithms, with a comparison to the results of using random noiselet measurements with ℓ_1 minimization. A real-valued noiselet transform is used (fast code provided by Romberg) and YALL1 [32]. Note that Candès and Romberg have used noiselets for compressed sensing in [9]. The measurements are again operated on the wavelet domain of each sparsified image. The reference images with pre-determined sparsities

TABLE 4.2
Reconstruction errors and times from experiments with all three versions of chirp compressed sensing.

	Chirp v1		Chirp v2		Chirp v3	
	error	time	error	time	error	time
cameraman, 256 × 256, 14%	-239 dB	7.3 h	-249 dB	12 h	-109 dB	28 s
knee, 256 × 256, 10%	-245 dB	152 m	-260 dB	80 m	-119 dB	7 s



(a) reference (b) Chirp v1 (c) Chirp v2 (d) Chirp v3

FIG. 4.2. *Reconstructed images with Chirp v1, v2, and v3 look merely identical to the (sparsified) reference images.*

are shown in column (a). The reconstructed images by noiselets, chirp, and RM are shown in columns (b), (c), and (d), respectively. The results of column (c) is by chirp v3 algorithm and (d) is by RM v3 algorithm. Both chirp v3 and RM v3 provide better reconstruction fidelity than reconstruction using random noiselet measurements.

5. Discussion and Conclusions. This paper has described an approach to improving the utility of compressed sensing using deterministic matrices consisting of chirps or second-order Reed-Muller sequences. Specifically, a reconstruction algorithm and some variations that extend the utility of the algorithms described in [1] and [21] to less sparse signals and, in particular, images. Several examples illustrating the performance of this approach with images with varying modest degrees of sparsity and having different characteristics in other respects were also presented.

The authors believe it is likely to be possible to make further improvements in both reconstruction efficiency and accuracy using chirps and RM sequences in compressed sensing of signals having limited sparsity by improving the process of peak detection entailed in identifying the locations of non-zero values in the sparse signal.

Additionally, it is noted that the method used here, although applied to images, is essentially one-dimensional. Ongoing work is investigation of more natural formulations for multi-dimensional signals. Finally, the set of RM sequences is naturally partitioned into a hierarchy of sub-families, beginning with the Kerdock sequences that have very good correlation properties and continuing through Delsarte-Goethals

TABLE 4.3
Reconstruction errors and times from experiments with all three versions of RM compressed sensing.

	RM v1		RM v2		RM v3	
	error	time	error	time	error	time
cameraman, 256 × 256, 14%	failed	-	failed	-	-43.7 dB	33 s
knee, 256 × 256, 10%	failed	-	-284 dB	31 m	-108 dB	18 s

TABLE 4.4
Reconstruction errors and times using noiselets, chirp v3, and RM v3 algorithms.

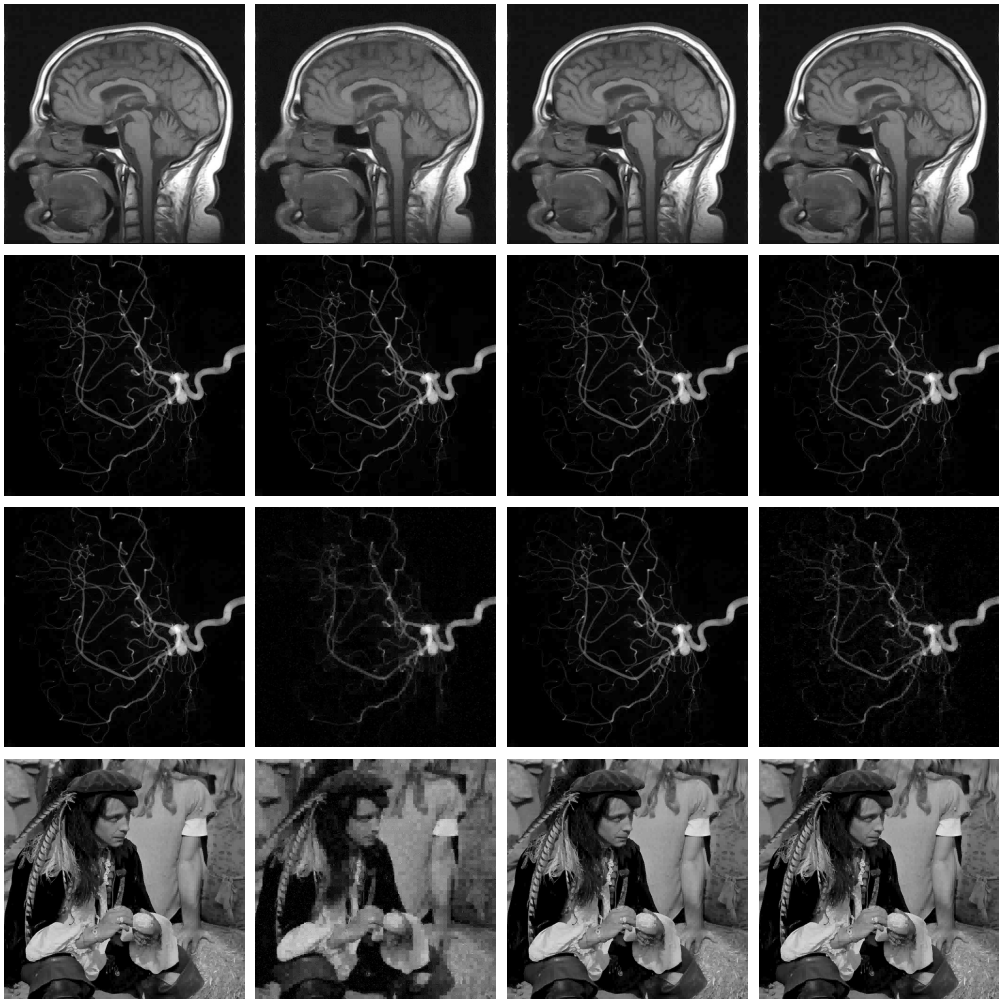
image size, sparsity	n/N	noiselets	Chirp v3	RM v3
brain 512 × 512, 7%	25%	-25.2 dB	-123 dB	-119 dB
vessel 512 × 512, 5%	25%	-38.9 dB	-129 dB	-125 dB
vessel 512 × 512, 5%	12.5%	-10.1 dB	-49.9 dB	-10.6 dB
man 1024 × 1024, 2.38%	6.25%	-14.5 dB	-112 dB	-109 dB

families in which the inner product magnitudes between sequences increase according to orderly bounds. It may be possible to exploit this structure in the compressed sensing process, particularly when some a priori information is available about the distribution of the locations of the non-zero components. This is often the case with images, many classes of which are known to possess a low-pass structure in addition to being compressible in certain bases (e.g., wavelets). They admit approximations that are not only somewhat sparse, but in which the locations of the non-zero coefficients are not uniformly distributed.

Acknowledgments. The authors wish to thank Robert Calderbank, Sina Jafarpour, Stephen Howard, and Stephen Searle for extensive discussions of their work in deterministic compressed sensing, particularly with regard to one-dimensional chirp and RM reconstruction algorithms. Justin Romberg's advice about noiselets and ℓ_1 algorithms helped us significantly, and we also appreciate his guiding us to sources for fast implementations of ℓ_1 minimization techniques. Jim Pipe has provided tireless guidance about medical imaging as well as the knee, vessel, and brain images that appear in this paper.

REFERENCES

[1] LORNE APPLEBAUM, STEPHEN D. HOWARD, STEPHEN SEARLE, AND ROBERT CALDERBANK, *Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery*, Applied and Computational Harmonic Analysis, 26 (2009), pp. 283 – 290.



(a) reference (b) noiselets (c) chirp v3 (d) RM v3

FIG. 4.3. *Reconstructed images with noiselets, chirp, and RM corresponding to Table 4.4.*

[2] RICHARD BARANIUK, *Compressive sensing*, IEEE Signal Processing Magazine, 24 (2007), pp. 118 – 121.
[3] DROR BARON, MARCO F. DUARTE, SHRIRAM SARVOTHAM, MICHAEL B. WAKIN, AND RICHARD G. BARANIUK, *An information-theoretic approach to distributed compressed sensing*, in Proceedings of the 43rd Allerton Conference on Communications, Control, and Computing, September 2005.
[4] AMIR BECK AND MARC TEBOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Science, 2 (2009), pp. 183 – 202.
[5] ÅKE BJÖRCK, *Numerical methods for least squares problems*, SIAM Press, 1996.
[6] ROBERT CALDERBANK, 2009. Personal communication.
[7] ROBERT CALDERBANK, STEPHEN HOWARD, AND SINA JAFARPOUR, *Cconstruction of a large class of deterministic matrices that satisfy a statistical isometry property*, IEEE Journal on Selected Topics in Signal Processing, 29 (2009).
[8] EMMANUEL CANDÈS, *Compressive sampling*, in Proceedings of the International Congress of Mathematicians, vol. III, European Mathematical Society, Madrid, 2006, pp. 1433–1452.
[9] EMMANUAL CANDÈS AND JUSTIN ROMBERG, *Sparsity and incoherence in compressive sampling*, Inverse Problems, 23 (2007), pp. 969 – 985.

[10] Emmanuel Candès, Justin Romberg, and Terence Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Transactions on Information Theory, 52 (2006), pp. 489 – 509.

[11] ———, *Stable signal recovery from incomplete and combinatorics: a unified approach to sparse signal recovery*, Communications on Pure and Applied Mathematics, 59 (2006), pp. 1207 – 1223.

[12] Emmanuel Candès and Terence Tao, *Near optimal signal recovery from random projections: Universal encoding strategies?*, IEEE Transactions on Information Theory, 52 (2006), pp. 5406 – 5425.

[13] Emmanuel J. Candès and Justin K. Romberg, *Signal recovery from random projections*, in Proceedings of SPIE, vol. 5674, 2005, pp. 76 – 86.

[14] Emmanuel J. Candès and Michael B. Wakin, *An introduction to compressive sampling*, IEEE Signal Processing Magazine, 25 (2008), pp. 21 – 30.

[15] Emmanuel Candès et al., *ℓ_1-magic home page*. http://www.acm.caltech.edu/l1magic/.

[16] Ronald DeVore, *Deterministic constructions of compressed sensing matrices*, Journal of Complexity, 23 (2007), pp. 918 – 925.

[17] David L. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory, 52 (2006), pp. 1289 – 1306.

[18] David L. Donoho, Martin Vetterli, R. A. DeVore, and Ingrid Daubechies, *Data compression and harmonic analysis*, IEEE Transactions on Information Theory, 44 (1998), pp. 2435 – 2476.

[19] Mário A. T. Figueiredo, Robert D. Nowak, and Stephen J. Wright, *Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems*, IEEE Journal of Selected Topics in Signal Processing, 1 (2007), pp. 586 – 597.

[20] Elaine T. Hale, Wotao Yin, and Yin Zhang, *Fixed-point continuation for ℓ_1-minimization: Methodology and convergence*, SIAM Journal on Optimization, 19 (2008), pp. 1107 – 1130.

[21] Stephen D. Howard, A. Robert Calderbank, and Stephen J. Searle, *A fast reconstruction algorithm for deterministic compressive sensing using second order Reed-Muller codes*, in Proceedings of the Conference on Information Sciences and Systems, Princeton, New Jersey, March 2008, pp. 11 – 15.

[22] A. M. Kerdock, *A class of low rate nonlinear binary codes*, Information and Control, 20 (1972), pp. 182 – 187.

[23] Michael Lustig, David Donoho, and John M. Pauly, *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magnetic Resonance in Medicine, 58 (2007), pp. 1182 – 1195.

[24] F. J. MacWilliams and N. J. A. Sloane, *The theory of error correcting codes*, vol. 16, North-Holland Mathematical Library, 1976.

[25] Kangyu Ni, Somantika Datta, Prasun Mahanti, Svetlana Roudenko, and Douglas Cochran, *Using reed-muller codes as deterministic compressive sensing matrices for image reconstruction*, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Dallas, March 2010.

[26] Kangyu Ni, Prasun Mahanti, Somantika Datta, Svetlana Roudenko, and Douglas Cochran, *Image reconstruction by deterministic compressive sensing with chirp matrices*, in Proceedings of SPIE, vol. 7497, October 2009.

[27] Christopher C. Paige and Michael A. Saunders, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Transactions on Mathematical Software, 8 (1982), pp. 43 – 71.

[28] Justin Romberg, *Imaging via compressive sampling*, IEEE Signal Processing Magazine, 25 (2008), pp. 14 – 20.

[29] Dharmpal Takhar, Jason N. Laska, Michael B. Wakin, Marco F. Duarte, Dror Baron, Shriram Sarvotham, Kevin F. Kelly, and Richard G. Baraniuk, *A new compressive imaging camera architecture using optical-domain compression*, in Proceedings of SPIE, January 2006, pp. 43 – 52.

[30] Ewout van den Berg and Michael P. Friedlander, *Probing the Pareto frontier for basis pursuit solutions*, SIAM Journal on Scientific Computing, 31 (2008), pp. 890 – 912.

[31] Xiang-Gen Xia, *Discrete chirp-Fourier transform and its application to chirp rate estimation*, IEEE Transactions on Signal Processing, 48 (2000), pp. 3122 – 3133.

[32] Yin Zhang, *User's guide for YALL1: Your agorithms for L1*, Tech. Report CAAM TR09-17, 2009. Available at: http://www.caam.rice.edu/∼yzhang/YALL1/.