USING REED-MULLER CODES AS DETERMINISTIC COMPRESSED SENSING MATRICES FOR IMAGE RECONSTRUCTION

Kangyu Ni[1]*, Somantika Datta*[2]*, Prasun Mahanti*[3]*, Svetlana Roudenko*[1]*, and Douglas Cochran*[1]

[1]School of Mathematical and Statistical Sciences, Arizona State University
[2]Program in Applied and Computational Mathematics, Princeton University
[3]School of Electrical, Computer, and Energy Engineering, Arizona State University

ABSTRACT

An image reconstruction algorithm by compressed sensing (CS) with deterministic matrices from Reed-Muller (RM) codes is introduced. An existing 1d signal reconstruction algorithm by Howard et al. [4] using CS with RM codes produces accurate results only for very sparse signals; its speed and accuracy suffer when the degree of sparsity is not high, making it inapplicable for 2d signals. We propose an efficient 2d CS with RM codes algorithm, provide medical image reconstruction examples and compare it with the original 2d CS (with noiselets) algorithm. Moreover, new steps are introduced: initial best approximation, a greedy algorithm for the nonzero locations, and a new approach in the least squares method, improving SNR, time and stability.

***Index Terms*—** compressed sensing, Reed-Muller codes, image reconstruction

1. INTRODUCTION

This paper is primarily concerned with using second order Reed-Muller (RM) codes for deterministic compressed sensing and designing an efficient reconstruction algorithm in the imaging regime, specifically for medical images. Such images have good sparsity properties compared to natural images). Let P be a $m \times m$ binary symmetric matrix, $a = (a_0, ..., a_{m-1})$, $b = (b_0, ..., b_{m-1}) \in \mathbb{Z}_2^m$ be binary vectors. A second order RM function of length 2^m is

$$\phi_{P,b}(a) = \frac{(-1)^{\mathrm{wt}(b)}}{\sqrt{2^m}} i^{(2b+Pa)^T a}, \tag{1}$$

where $\mathrm{wt}(b)$ is the weight of b, i.e., the number of 1s the vector contains. The vector b in the linear term and the matrix P in the quadratic term can be considered as the "frequency" and "chirp rate" of the binary world, respectively. Let P be zero-diagonal, then $\phi_{P,b}$ is real-valued. Howard et al. [4] proposed a compressed sensing matrix of the form

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

where each U_{P_j} is a $2^m \times 2^m$ orthonormal matrix whose columns are the second order RM functions for P_j, with b going through all possible binary vectors. Since there are $2^{m(m-1)/2}$ possible $m \times m$ binary zero-diagonal symmetric matrices, the associated full RM matrix is $2^m \times 2^{m(m+1)/2}$. Defining $N = 2^{m(m+1)/2}$ and $n = 2^m$, a k-sparse signal $x \in \mathbb{C}^N$ yields a measurement $y = \Phi_{RM} x \in \mathbb{C}^n$, which is the superposition of k RM functions

$$y(a) = z_1 \phi_{P_1,b_1}(a) + z_2 \phi_{P_2,b_2}(a) + \cdots + z_k \phi_{P_k,b_k}(a). \tag{3}$$

In (3), 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. To recover x, Howard et al. use the fast Hadamard transform (FHT) to detect one by one the nonzero locations, (P_j, b_j) pairs, whose total computational complexity is $O(kn(\log n)^2)$. The magnitudes z_j are then found by solving the associated least squares problem. For reconstructing sparse signals, in terms of reconstruction speed and fidelity, this is more efficient than $\ell 1$ minimization with random matrices, whose computational complexity is $O(knN)$.

Despite the success for accurately reconstructing sparse one-dimensional signals, directly applying this algorithm to real images is impractical. This is because real images are not as sparse in any transform domain as the one-dimensional signals in Howard et al.'s paper [4], and therefore, the reconstruction error becomes large. For example, a 128×128 image with 10% sparsity has 1638 nonzero coefficients. At least approximately 4000 measurements are needed in order for the reconstruction to be correct. This implies that only four P matrices in (2) are needed to solve $y = \Phi_{RM} x \in \mathbb{C}^{2^m}$, and therefore, the efficiency of finding nonzero locations is not fully utilized in the imaging regime. Moreover, their algorithm requires about 1638 iterations and the least squares computations become very large.

Section 2 shows our proposed method for compressive sensing and reconstruction of images. Section 3 shows experiments with 256×256 medical images and comparison with Candès and Romberg's method with noiselets [3]. Section 4 summarizes the results.

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

2. METHOD

2.1. Construction of the compressive sensing matrix

As explained in the introduction, due to the sparsity nature of images and rule of thumb in compressive sensing, only four P matrices are needed for the sensing matrix, instead of using the entire matrix given in (2). Therefore, there is flexibility to construct the sensing matrix according to the choice of P matrices. The statistics of inner products of two columns, each of which is taken from U_{P_i} and U_{P_j}, respectively, are [6]

$$|\langle \phi_{P_i,\cdot} | \phi_{P_j,\cdot} \rangle| = \begin{cases} 1/\sqrt{2^l}, & 2^l \text{ times}, \\ 0, & 2^m - 2^l \text{ times}, \end{cases} \quad (4)$$

where $l = \text{rank}(P_i - P_j)$. If $l = m$, the inner products are always $1/\sqrt{2^m}$, which is smaller than if $l < m$. This is desirable because the nonzero locations are not known and the inner products between any two columns are kept as small as possible. Since the rank between two zero-diagonal symmetric binary matrices is always even, m is even in this paper. 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 [5]. There are 2^{m-1} elements in the Kerdock set, and the construction of the Kerdock set can be found in [2]. The sensing matrices used in this paper are constructed in the form

$$\Phi = \begin{bmatrix} U_{P_1} & U_{P_2} & U_{P_3} & U_{P_4} \end{bmatrix}, \quad (5)$$

where P_1, P_2, P_3, and P_4 can be any matrices from the Kerdock set. For example, the sensing matrix for a 256×256 $(= 2^{16})$ image is of the size $2^{14} \times 2^{16}$. Therefore, only 25% of the measurements are sampled.

2.2. Initial best approximation of the solution

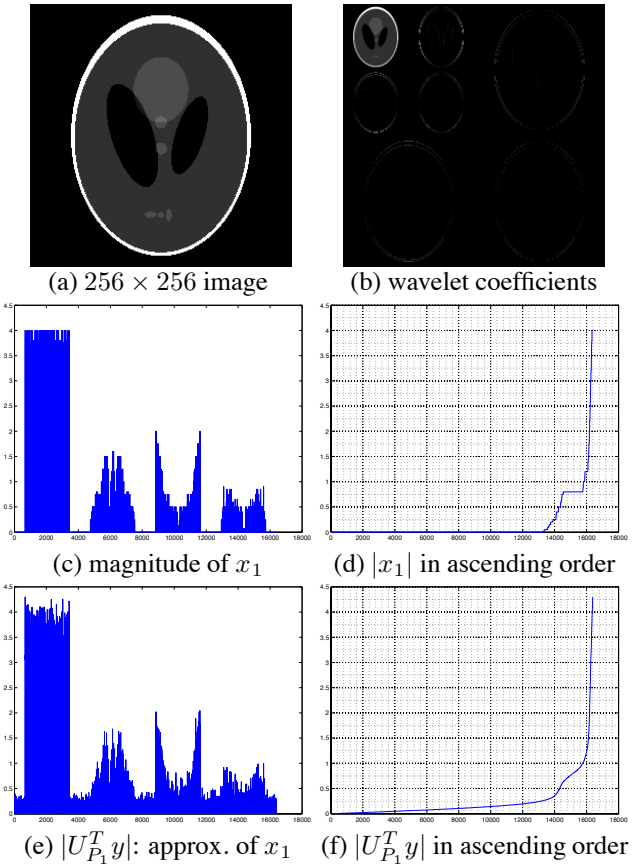
We propose a method to detect a large portion of the nonzero locations in one step, based on the knowledge from Fourier analysis that the energy of the wavelet coefficients is concentrated in the upper-left region. This does not require a priori knowledge of individual images. Let's begin by writing the measurements as

$$\begin{aligned} y &= \Phi x \\ &= \begin{bmatrix} U_{P_1} & U_{P_2} & U_{P_3} & U_{P_4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \\ &= U_{P_1} x_1 + U_{P_2} x_2 + U_{P_3} x_3 + U_{P_4} x_4, \end{aligned} \quad (6)$$

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. We propose to estimate x_1 by

$$U_{P_1}^T y = x_1 + U_{P_1}^T U_{P_2} x_2 + U_{P_1}^T U_{P_3} x_3 + U_{P_1}^T U_{P_4} x_4. \quad (7)$$

The last three terms are small because x_2, x_3, and x_4 are much sparser and smaller than x_1, and the rest of the terms $U_{P_1}^T U_{P_2}, U_{P_1}^T U_{P_3}$, and $U_{P_1}^T U_{P_4}$ are small, as discussed in Sec. 2.1. Therefore, $U_{P_1}^T y \approx x_1$. In the case when all nonzero locations are in the upper-left region, i.e., x_2, x_3, and x_4 are zero, $U_{P_1}^T y$ is equal to x_1, which automatically completes the image reconstruction.



(a) 256×256 image (b) wavelet coefficients

(c) magnitude of x_1 (d) $|x_1|$ in ascending order

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

Fig. 1. In (b) it is shown that the energy of wavelet coefficients is concentrated in the upper-left region; in (c) and (e) it is shown that $|U_{P_1}^T y|$ approximates $|x_1|$ well; (d) is the plot of $|x_1|$ sorted in ascending order, and finally, (f) is $|U_{P_1}^T y|$ in ascending order.

Fig. 1 shows an example of this method, where (a) is the Shepp-Logan phantom image of pixel resolution 256×256 and (b) shows its Haar wavelet coefficients. In (c) and (e), we see that $U_{P_1}^T y$ well approximates x_1. Therefore, even though x_1 is unknown, most of the nonzero locations in x_1 can be detected. The graph in (f) sorts the magnitudes of the approximation in ascending order. Most of the nonzero locations in x_1 can be detected by selecting the locations above the first critical point from 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 following linear squares problem:

$$\min_z \|Az - y\|, \tag{8}$$

where A is a matrix whose columns consist of ϕ_{P_l,b_l} and $z = [z_1, .., z_j]^T$ is a vector. Finally, subtracting the linear (their respective magnitudes z_l) sum of the found columns from y gives the residual, $y_0(\ell) = y(\ell) - \sum_{l=1}^{j} z_l \phi_{P_l,b_l}(\ell)$.

2.3. Reconstruction algorithm

This section describes the proposed method, which is summarized in the following steps:

1. Get initial best approximation (see Sec. 2.2).
2. Find multiple (P_l, b_l) pairs.
3. Determine z_j by updated linear least squares solutions.
4. Get residual $y_0(\ell) = y_0(\ell) - \sum_{l=1}^{j} z_l \phi_{P_l,b_l}(\ell)$.
5. Repeats steps 2–4 until y_0 is sufficiently small.

The second step first "de-chirps" the samples residual with all four P matrices in the measurement matrix, i.e., multiplication by $\overline{\phi_{P_j,0}}$ eliminates the quadratic term, and then use the FHT to detect the remaining nonzero locations:

$$w_{P_j}(\ell) = \left| \mathrm{FHT}\left(y_0(\ell)\overline{\phi_{P_j,0}(\ell)} \right) \right|, \qquad j = 1, 2, 3, 4. \tag{9}$$

Each obtained magnitude corresponds to a unique pair (P_j, ℓ). The largest magnitudes are selected and the corresponding pairs give the nonzero locations.

The third step solves the magnitudes z_j by the linear least squares problem in (8), with the associated matrix A. Notice 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 step and c is the newly found columns. To solve these least squares problems without treating each problem (iteration) independently, we use an updated pseudo inverse solution method whose computation is based on the previous calculations. The pseudo inverse solution of (8) is

$$z_{sol} = (A^*A)^{-1}A^*y, \tag{10}$$

where * indicates conjugate transpose. Finding the inverse of

$$A^*A = \begin{bmatrix} \widetilde{A}^*\widetilde{A} & \widetilde{A}^*c \\ c^*\widetilde{A} & c^*c \end{bmatrix} \tag{11}$$

can be made efficiently by the Schur-Banachiewicz blockwise inversion formula (e.g., see [1]):

$$\begin{bmatrix} D & E \\ F & G \end{bmatrix}^{-1} = \begin{bmatrix} D^{-1} + D^{-1}EVFD^{-1} & -D^{-1}EV \\ -VFD^{-1} & V \end{bmatrix}, \tag{12}$$

where $V = (G - D^{-1}E)^{-1}$. Note that D^{-1} is known from the previous iteration and the size of $V = (G - D^{-1}E)^{-1}$ is small. The calculation of A^*y can also be updated by

$$A^*y = \begin{bmatrix} \widetilde{A}^*y \\ c^*y \end{bmatrix}, \tag{13}$$

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

3. RESULTS

In the experiments, each original image was sparsified by computing its Haar wavelet transform and retaining a predetermined fraction of its wavelet coefficients, keeping the largest and setting the rest to zero. Then, 25% noiselet measurements and RM measurements were taken for noiselet and RM reconstruction, respectively. Fig. 2 (a) is the 256×256 vessel image with 10% sparsity. The reconstructed image using (our implementation of) Candès and Romberg's method with noiselets [3] is shown in (b), which has noticeable patches. The reconstructed image in (c) is by our method and is identical to the reference image in (a). In (b), (d), and (f), the respective horizontal slices at the center are shown and the slice by our method is identical to the slice in the reference image. The reconstruction error by our method (-283 dB), is far better than the error by noiselets (-19 dB), where the error is defined as:

$$\mathrm{Error(dB)} = 10\log_{10}\left[\frac{||x_{\mathrm{actual}} - x_{\mathrm{reconstructed}}||^2}{||x_{\mathrm{actual}}||^2} \right]. \tag{14}$$

Fig. 3 shows image reconstruction for the 256×256 MRI knee image. The true image is 10%-sparse and 25% samples were used for reconstruction. Our method in this case also outperformed the noiselet method. The error for RM was -284 dB, whereas the error for noiselets was -22 dB.

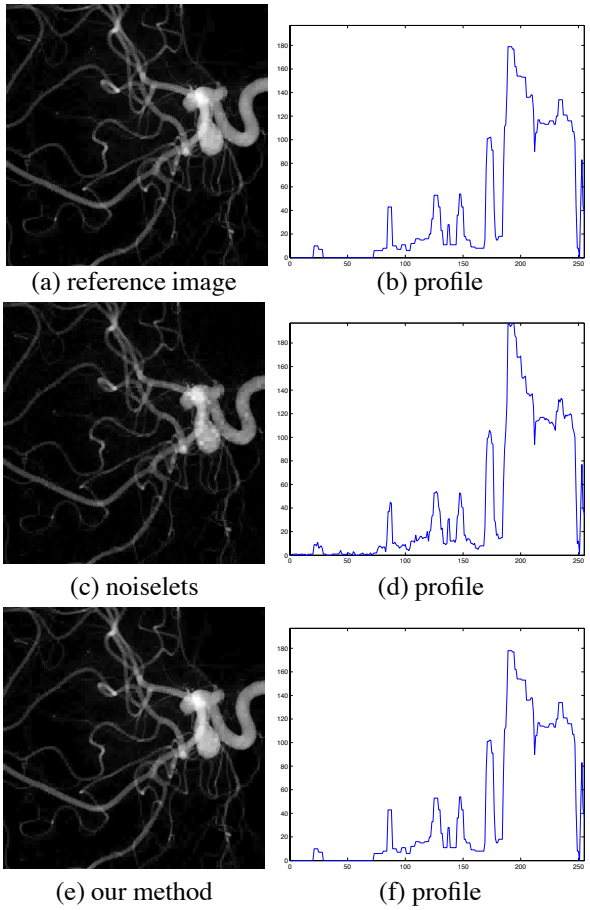
Table 1 shows reconstruction error in dB for various 256×256 images with sparsities from $6\% - 11\%$, using 25% measurements by the proposed algorithm. The reconstruction works well up to around 10%.

Sparsity	vessel	knee	phantom	cameraman
6%	-288	-289	-286	-289
7%	-287	-288	-286	-287
8%	-286	-286	-285*	-286
9%	-284	-285	N/A	-282
10%	- 283	-284	N/A	-32
11%	- 15	-281	N/A	-17

Table 1. Various experiments show our method is able to correctly reconstruct images with sparsity up to around 10% with only 25% measurements. *The phantom image is 7.27% sparse, thus no data is available greater than 8% sparsity.

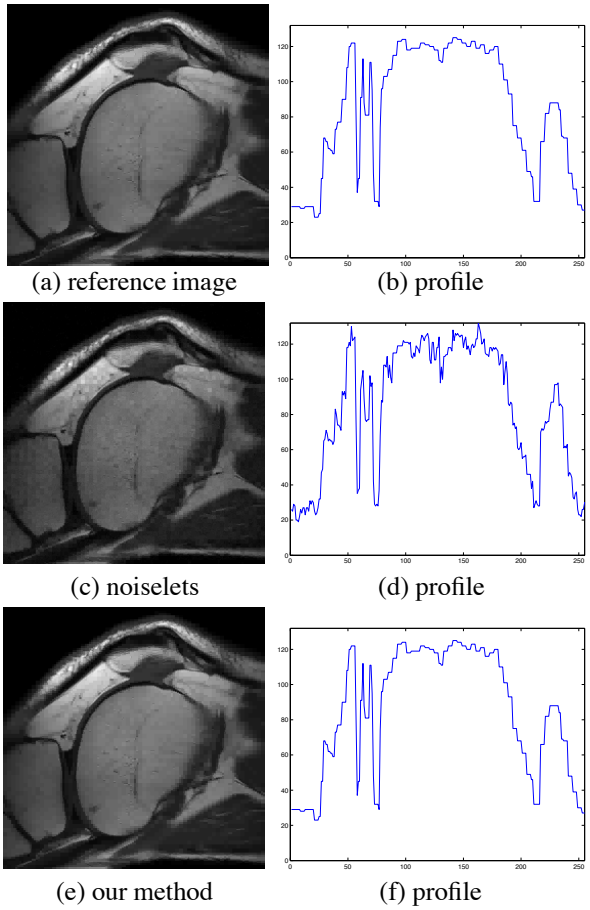
4. CONCLUSION

We propose a new image reconstruction algorithm by deterministic compressed sensing method especially suitable for medical images. Compared to the original compressed sensing reconstruction method by noiselets, our algorithm



(a) reference image (b) profile
(c) noiselets (d) profile
(e) our method (f) profile

Fig. 2. (a) 256×256 vessel image with 10% sparsity. (c) reconstructed image with noiselets, error is -19 dB. (e) reconstructed image with our method, whose error is smaller, -283 dB. The respective horizontal slices in (b), (d), and (f) also show our result is identical to the reference image.



(a) reference image (b) profile
(c) noiselets (d) profile
(e) our method (f) profile

Fig. 3. (a) 256×256 knee 10%-sparse image. (c) reconstructed image with noiselet, error = -22 dB. (e) reconstructed image with our method. The error is much smaller, -284 dB. The respective horizontal slices in (b), (d), and (f) also show our result is identical to the reference image.

provides more optimal reconstructions in terms of errors (or SNR) and computational efficiency. New features such as the initial best approximation method speeds up the solution and largely decreases the error (the proof is not shown due to space limitation) which can be observed in the examples shown. The total computational complexity of finding nonzero locations in the iterative step in our experiments is $O(\frac{1}{5}kn \log n)$, which is much smaller than $O(kn(\log n)^2)$ (compared with the Howard et al. same size algorithm). Finally, the 'updated' least squares method is incorporated to increase computational efficiency and stability.

5. ACKNOWLEDGMENTS

The authors wish to thank Robert Calderbank, Stephen Howard, and Stephen Searle for discussions on the Reed-Muller codes and Jim Pipe for providing the knee and vessel images.

6. REFERENCES

[1] A. Björck, *Numerical methods for least squares problems*, SIAM, 1996.

[2] A. R. Calderbank, personal communication, 2009.

[3] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems* **23**(3), 969 – 985, 2007.

[4] S. D. Howard, A. R. Calderbank, and S. J. Searle, "A fast reconstruction algorithm for deterministic compressive sensing using second order Reed-Muller codes," *Proc. of Conf. on Information Sciences and Systems*, 2008.

[5] A. M. Kerdock, "A class of low rate nonlinear binary codes," *Information and Control* **20**, 182 – 187, 1972.

[6] F. J. MacWilliams and N. J. A. Sloane, *The theory of error correcting codes*, Elsevier, 1976.