TENSOR AMPLITUDE EXTRACTION IN SENSOR ARRAY PROCESSING

Kyle R. Krueger, James H. McClellan, Waymond R. Scott Jr.

Georgia Institute of Technology
School of Electrical and Computer Engineering
Atlanta, GA 30332-0250

ABSTRACT

Sensor array measurements can be inverted to image a region containing targets. The resulting amplitude image is usually interpreted as target strength versus location, but often the imaged amplitude is a function of more parameters than just the location. Sparse target regions can be imaged with dictionary based modeling which relies on enumeration of each parameter with a dense grid. With many parameters, the dictionary becomes too large, which leads to computational complexity issues. This paper shows how additional parameters, such as target orientation and symmetry, can be represented by a tensor matrix instead of a simple amplitude. Furthermore, the tensor can be treated as a continuous variable just like amplitude, which enables extraction of multiple parameters, while reducing the storage requirements of the dictionary, and reducing off-grid modeling error.

Index Terms— Electromagnetic Induction (EMI), Low-Rank Approximation

1. INTRODUCTION

Unknown object detection that includes detailed parameter estimation is a challenging research problem. Locating landmines for disposal [1–5], radar imaging [6], and seismic processing [7] are examples of sensor array imaging problems that require parameter estimation. A very common, and very easy solution to these problems is to build a parametric model of all known variations of the objects of interest, make a set of array measurements, and then pick out the model parameters that match the measurements. This is also the basis for sparse recovery and compressive sensing [8].

Matching pursuit is a way to solve these problems [9]. The parameter space usually needs to be discretized and enumerated for all reasonable target variations. There are two common problems that arise with this approach. The first, and most important, is that the more general the dictionary, and the more unknown parameters included in the model, the larger the dictionary becomes. It is common for the dictionary to become unreasonably large. The second problem is that the nature of a dictionary is discrete sampling of a continuous parameter space. This gives rise to the "off-grid" problem which leads to modeling errors if the parameters of the measured object do not fall exactly on the sampled space of the dictionary.

This paper will discuss how the detection of a target with an electromagnetic (EM) field can be improved by using a tensor representation instead of a simple amplitude. The tensor matrix allows for a continuous approximation of some of the target parameters, such as the orientation and target symmetry. This eliminates the need for discretization, and reduces the dictionary size by approximately six orders of magnitude. This also allows for a low-rank approximation algorithm to be used since the symmetry and orientation of the magnetic polarizability are described with the eigenvalues and eigenvectors of the simple 3×3 tensor matrix. This idea of using a dictionary to solve for a continuous parameter is similar in spirit to the method in [10]. The method proposed in this paper differs from previous work [3] which assumed the orientation is known. Ho and Gader did not include orientation, but showed that changes in orientation affect the location estimates [11]. The work in [4] discretizes the orientation space and also assumes that every target can be generalized with dipole symmetry. Özdemir et al. solved the problem via a 6-D nonlinear least squares method instead of solving directly for the tensor representation with a semidefinite program [12].

This paper will examine the advantages of using the tensor amplitude representation in the context of magnetic objects measured with electromagnetic induction (EMI) sensors. Section 2 will explain how the model is built; Section 3 will show how this model leads to a tractable optimization problem; Sections 4 and 5 will give some preliminary results and concluding remarks, respectively.

2. MODEL SETUP

The model used for this problem can be found in [3, 4, 12]. The measurement set up consists of one transmitter and three receive coils (Fig. 1) that collect frequency response samples. The model used in [12] operates directly on the frequency domain model of the measurement response. For this paper, the model given in (1) uses the discrete spectrum of relaxation frequencies (DSRF) domain for target model simplification,

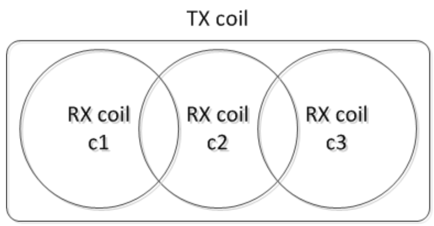

Fig. 1. Measurement device setup

and reduced dictionary size [13].

$$r_c^{l_s}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = C_0 \mathbf{g}_c^T(\boldsymbol{l}) \boldsymbol{R}(\boldsymbol{\alpha}) \,\mathrm{diag}(\boldsymbol{\lambda}) \boldsymbol{R}^T(\boldsymbol{\alpha}) \mathbf{f}(\boldsymbol{l}) \tag{1}$$

Only a single relaxation target will be discussed in this paper to simplify the presentation. Some additional details of the advantages of going to the DSRF domain for this problem can be seen in [4]. Equation (1) gives the EMI response of a target with magnetization polarizability $\boldsymbol{\lambda}$, rotated by $\boldsymbol{\alpha}$ with a $3{\times}3$ rotation matrix $\boldsymbol{R}$, at target location $\boldsymbol{l}_t$, and received with coil c at sensor location l_s. $\boldsymbol{\alpha}$ consists of the Euler rotation angles, (ϕ, θ, ψ); $\boldsymbol{l}$, the distance between l_s and $\boldsymbol{l}_t$, consists of the three relative spatial dimensions (x, y, z); $\mathrm{diag}(\boldsymbol{\lambda})$ gives the magnetic polarizability of the target on its principal axes. $\mathbf{g}_c(\boldsymbol{l})$ and $\mathbf{f}(\boldsymbol{l})$ are vectors containing the relative spatial components of the magnetic field on the receive coils and the transmit coil respectively. C_0 is a multiplicative constant that is explained fully in [12, 13].

Özdemir et al. [12] showed how this model can be cleverly rewritten into the dictionary model

$$\mathbf{r}_c(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \boldsymbol{A}_c(\boldsymbol{l}_t, \boldsymbol{\alpha})\boldsymbol{\lambda} \tag{2}$$

by stacking the sensor measurements into a column of data. Now the measurements received at each head are stacked in order to use the relative amplitudes of the head information to localize the target in cross range, y.

$$\mathbf{r}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \begin{bmatrix} \boldsymbol{A}_1(\boldsymbol{l}_t, \boldsymbol{\alpha}) \\ \boldsymbol{A}_2(\boldsymbol{l}_t, \boldsymbol{\alpha}) \\ \boldsymbol{A}_3(\boldsymbol{l}_t, \boldsymbol{\alpha}) \end{bmatrix} \boldsymbol{\lambda} = \boldsymbol{A}(\boldsymbol{l}_t, \boldsymbol{\alpha})\boldsymbol{\lambda}. \tag{3}$$

An example of $\mathbf{r}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda})$ can be seen in Fig 2.

A problem starts to arise when $\boldsymbol{A}$ is enumerated for all possible $\boldsymbol{l}_t$ and $\boldsymbol{\alpha}$. A reasonable size problem would have a $30{\times}30{\times}30$ spatial grid and a $90{\times}180{\times}180$ angle grid. $\boldsymbol{A}$ would then become size $N_c N_{l_s} {\times} N_x N_y N_z N_\phi N_\theta N_\psi \approx 90{\times}10^{11}$, which can easily become unreasonable for personal computers or hand-held field devices.

The model from (1) can be slightly changed to address this complexity issue. First, the middle terms can be combined into

$$\boldsymbol{T}(\boldsymbol{\alpha}, \boldsymbol{\lambda}) = \boldsymbol{R}(\boldsymbol{\alpha}) \,\mathrm{diag}(\boldsymbol{\lambda}) \boldsymbol{R}^T(\boldsymbol{\alpha}) = \begin{bmatrix} t_1 & t_4 & t_6 \\ t_4 & t_2 & t_5 \\ t_6 & t_5 & t_3 \end{bmatrix}, \tag{4}$$

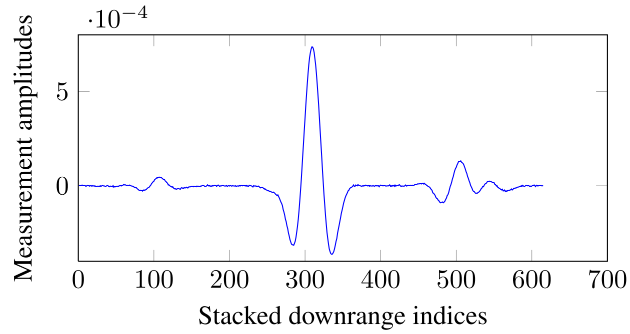

Fig. 2. Measurement vector

a symmetric positive semidefinite (PSD) matrix. Also, it is important to notice that $\boldsymbol{R}(\boldsymbol{\alpha}) \,\mathrm{diag}(\boldsymbol{\lambda}) \boldsymbol{R}^T(\boldsymbol{\alpha})$ is the eigen decomposition of $\boldsymbol{T}$. Meaning that if $\boldsymbol{T}$ is solved for directly, $\boldsymbol{R}$ and $\boldsymbol{\lambda}$ are easily recovered. This is significant, because solving for $\boldsymbol{T}$ only involves 6 unknowns, instead of the $3N_\phi N_\theta N_\psi \approx 10^6$ (where 3 corresponds to the length of $\boldsymbol{\lambda}$). The model then becomes

$$\begin{aligned} r_c^{l_s}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) &= C_0 \mathbf{g}_c^T(\boldsymbol{l}) \boldsymbol{T}(\boldsymbol{\alpha}, \boldsymbol{\lambda}) \mathbf{f}(\boldsymbol{l}) \\ &= C_0 (t_1 \mathbf{g}_c^T(\boldsymbol{l}) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{f}(\boldsymbol{l}) \\ &+ \ldots + t_6 \mathbf{g}_c^T(\boldsymbol{l}) \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \boldsymbol{f}(\boldsymbol{l})) \\ &= t_1 a_c^1(\boldsymbol{l}) + t_2 a_c^2(\boldsymbol{l}) + \ldots + t_6 a_c^6(\boldsymbol{l}) \\ &= \mathbf{a}_c(\boldsymbol{l})\mathbf{t}, \end{aligned}$$

where $\mathbf{a}_c$ is a $1{\times}6$ vector and $\mathbf{t}$ is a $6{\times}1$ vector. The same steps done in (3) can now be taken to get the measurement vector

$$\mathbf{r}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \boldsymbol{A}(\boldsymbol{l}_t)\mathbf{t}, \tag{5}$$

where $\boldsymbol{A}$ is now a $N_c N_{l_s} {\times} 6$ matrix. The model now has the orientation information embedded in the tensor, allowing for continuous orientation to be extracted without expanding the size of the dictionary.

3. OPTIMIZATION

Once the model has been created with the desirable properties, a technique to solve for the unknowns is needed. Assuming that there is some kind of modeling error, i.e., off grid target location, and measurement noise, a simple noise term $\boldsymbol{\eta}$ should be added to the measurement model

$$\mathbf{r}(\boldsymbol{l}_t, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \boldsymbol{A}(\boldsymbol{l}_t, \boldsymbol{\alpha})\mathbf{t} + \boldsymbol{\eta}. \tag{6}$$

3.1. Multiple Small Optimizations

First, assume that the location of the target is known. Then the only unknown parameters in the problem are the orientation and magnetic polarizability of the target on its principal axes. The symmetric tensor matrix is written as the sum of 6 symmetric basis matrices

$$\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda}) = t_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + ... + t_6 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad (7)$$

where the vector $\mathbf{t}$ has the amplitudes necessary to rebuild the larger tensor matrix. It is important to remember that $\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda})$ has the important properties, not the vector $\mathbf{t}$. In order to get a close fit, the optimization should include a constraint on the residual. But to keep the eigenvalues, which are the magnetic polarizability of the target on its principle axes, from becoming unreasonably large, a constraint on them is needed as well. The eigenvalues can be constrained by minimizing the trace of $\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda})$ because $\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda})$ is a square matrix. This type of problem can be solved with a semidefinite program (SDP) [14]. The optimization problem setup becomes

$$\begin{aligned} \min \quad & \mathbf{tr}(\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda})) \\ \text{s.t.} \quad & \boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda}) \succeq 0 \\ & \| \mathbf{r}(\boldsymbol{l}_t,\boldsymbol{\alpha},\boldsymbol{\lambda}) - \boldsymbol{A}(\boldsymbol{l}_t)\mathbf{t} \|_2 < \epsilon, \end{aligned} \quad (8)$$

and with a little reorganization can be passed into a solver such as CVX [15]. $\boldsymbol{T}(\boldsymbol{\alpha},\boldsymbol{\lambda}) \succeq 0$ represents the PSD constraint. The residual constraint ϵ is set by the noise level in the system and not due to the discretization of the dictionary. This is an advantage since dictionary elements that are located close together tend to look very similar to one another, generally with small amplitude changes being the only difference. This will be addressed in the next subsection.

The optimization problem in (8) will provide the tensor information for the target, giving the orientation and magnetic polarizability, which will aid in finding the location. Now that there is a set up to solve for the tensor information assuming the location is known, the problem needs to be expanded to find an unknown location. The minimization in (8) could be carried out for each possible spatial grid location ($\boldsymbol{l}_t$), and the residual evaluated

$$\text{Res}(\boldsymbol{l}_t) = \| \mathbf{r}(\boldsymbol{l}_t,\boldsymbol{\alpha},\boldsymbol{\lambda}) - \boldsymbol{A}(\boldsymbol{l}_t)\mathbf{t} \|_2 . \quad (9)$$

If there is a single target present, the minimum residual would provide the location estimate. If there were more than one single target, a matching pursuit type technique could be run to iterate and extract the target locations.

3.2. Single Large Optimization

The problem in the previous subsection can be solved with a single optimization. This will allow for the convex optimization problem to use the inherent sparsity in the number of targets to its advantage in finding the location. First, the model dictionary needs to be expanded to account for the unknown locations. The $\boldsymbol{A}(\boldsymbol{l}_t)$ matrices for the N_{l_t} possible target locations are concatenated into a block matrix

$$\boldsymbol{A} = [\boldsymbol{A}(1) \mid \boldsymbol{A}(2) \mid ... \mid \boldsymbol{A}(N_{l_t})]. \quad (10)$$

The vector $\mathbf{t}$ now becomes a length $6N_{l_t}$ sparse vector, and

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{T}_1(\boldsymbol{\alpha},\boldsymbol{\lambda}) & 0 & 0 & 0 \\ 0 & \boldsymbol{T}_2(\boldsymbol{\alpha},\boldsymbol{\lambda}) & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{T}_{N_{l_t}}(\boldsymbol{\alpha},\boldsymbol{\lambda}) \end{bmatrix}. \quad (11)$$

All of the previous properties still apply, $\boldsymbol{T}$ is symmetric, $\boldsymbol{T}\succeq 0$, and can be constructed directly from the solution vector $\mathbf{t}$. The fact that $\boldsymbol{T}\succeq 0$ follows straight from the fact that each $\boldsymbol{T}_{l_t}(\boldsymbol{\alpha},\boldsymbol{\lambda})\succeq 0$. It is obvious in the block diagonal structure of $\boldsymbol{T}$, that the eigenvalues of $\boldsymbol{T}$ are the enumeration of all the eigenvalues from all $\boldsymbol{T}_{l_t}$. It follows directly that if there exists a $\boldsymbol{T}_{l_t}$ that has a negative eigenvalue, then $\boldsymbol{T}$ must also have the same negative eigenvalue. Also, if $\boldsymbol{T}$ contains at least one negative eigenvalue, then there must exist at least one $\boldsymbol{T}_{l_t}$ that has the corresponding negative eigenvalue. This proves that $\boldsymbol{T}\succeq 0$ if and only if all $\boldsymbol{T}_{l_t}\succeq 0$. Now (8) becomes the expanded convex optimization problem

$$\begin{aligned} \min \quad & \mathbf{tr}(\boldsymbol{T}) \\ \text{s.t.} \quad & \boldsymbol{T} \succeq 0 \\ & \| \mathbf{r}(\boldsymbol{l}_t,\boldsymbol{\alpha},\boldsymbol{\lambda}) - \boldsymbol{A}\mathbf{t} \|_2 < \epsilon. \end{aligned} \quad (12)$$

Using the trace of $\boldsymbol{T}$ is a relaxation of the rank minimization problem and will take advantage of the fact that $\boldsymbol{T}$ is going to be very low rank, generally between one and six [16]. There maybe some concern that this problem could become extremely large, since $\boldsymbol{T}$ is of size $3N_xN_yN_z \times 3N_xN_yN_z$. This is not a problem, since $\boldsymbol{T}$ is only a five diagonal, symmetric matrix, and the trace is only concerned with the main diagonal. This allows many storage efficient techniques to be used to keep complexity down. This low-rank matrix approximation is in effect enforcing a sparsity constraint on the number of targets. If there is only one target, and there is no noise, this technique and the one from the previous subsection would provide identical results. Since those assumptions are not true in practice, noise must be dealt with.

The noise must be bounded by the coherence of the dictionary grid. The tensor representation technique avoids the need to grid the orientation parameters, but it does not address the need for a discretized grid in the spatial location dimensions. There could quite possibly be a way to eliminate the need for a location grid as well, but it has not been addressed here. This leads to an issue of dictionary coherence in the location grid. This is the euclidean distance between dictionary vectors corresponding to similarly oriented targets which is defined for this problem by

$$d_{ij} = \| \mathbf{r}(i,\boldsymbol{\alpha},\boldsymbol{\lambda}) - \mathbf{r}(j,\boldsymbol{\alpha},\boldsymbol{\lambda}) \|_2 . \quad (13)$$

For this particular dictionary, d_{ij} is very small with respect to the amplitude of each signal if i and j are close. For a particular orientation, $d_{ij}/2 \geq \| \boldsymbol{\eta} \|_2$. This means the norm of the noise vector must be less than half of the smallest distance between two dictionary locations. This should be taken into consideration when setting up the grid for the possible locations, as well as in the design of the measurement system.

4. RESULTS

Two simple experiments were run to test the method. The simulation treats a very small problem, but can be expanded easily. Only two spatial dimensions are considered, y and z. N_y=7 at 2cm spacing, N_z=8 with 1cm spacing. Also, only ϕ and θ are used for orientation. A consideration for setting up these problems is how to choose the ϵ from (12). The SNR for each experiment is set at 35dB. For these experiments, a starting point for ϵ is selected to be a simple estimate of the noise. It is reasonable to assume that much of the downrange measurements will contain no target, so the norm is taken in this estimated target-less area to get the estimated noise amplitude. Then an L-curve, like the one in Fig. 3, is made by varying ϵ below and above the starting ϵ. A point is then selected at the knee of the curve to find the best solution.

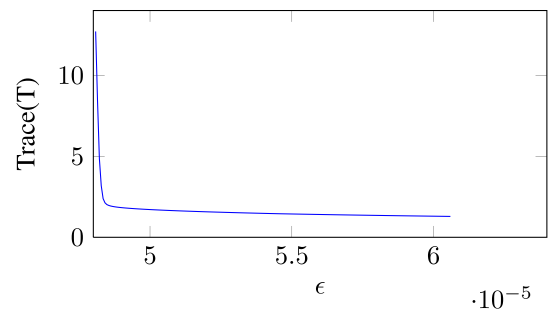

Fig. 3. L-curve for ϵ selection

The first experiment consists of a single target located at $(y, z) = (0, 6.5)$ cm, with a 2-dimensional EM symmetry $\boldsymbol{\lambda} = (.5, 0, 1)$, oriented at $(\phi, \theta) = (0°, 22.5°)$, which is represented by the tensor

$$\boldsymbol{T} = \begin{bmatrix} 0.92 & 0.38 \\ 0 & 0 \\ -0.38 & 0.92 \end{bmatrix} \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.92 & 0.38 \\ 0 & 0 \\ -0.38 & 0.92 \end{bmatrix}^T$$

written as an eigenvalue expansion. The estimated tensor corresponded to the correct location, and the estimated tensor is

$$\hat{\boldsymbol{T}} = \begin{bmatrix} 0.90 & 0.42 \\ 0.03 & 0.00 \\ -0.42 & 0.90 \end{bmatrix} \begin{bmatrix} 0.48 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.90 & 0.42 \\ 0.03 & 0.00 \\ -0.42 & 0.90 \end{bmatrix}^T .$$

The $\boldsymbol{\lambda}$ are normalized such that the largest eigenvalue is one. The estimation of the tensor is quite accurate. $\lambda(1)$ has less than a 5% error, and $\lambda(2)$ is exact. The orientation estimate is also very accurate. By using simple trigonometry, the angles of the eigenvectors (rotation matrices) are within $4°$ of the actual orientation.

The second experiment consists of two targets located at $(y_1, z_1) = (0, 6.5)$ cm and $(y_2, z_2) = (-6, 7.5)$ cm, with EM symmetries $\boldsymbol{\lambda}_1 = (0, 0, 1)$ and $\boldsymbol{\lambda}_2 = (1, 0, 0)$, both are oriented at $(\phi, \theta) = (45°, 22.5°)$, and represented with tensors

$$\boldsymbol{T}_1 = \begin{bmatrix} 0.27 \\ -0.27 \\ 0.92 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} 0.27 \\ -0.27 \\ 0.92 \end{bmatrix}^T ,$$

and

$$\boldsymbol{T}_2 = \begin{bmatrix} -0.65 \\ 0.65 \\ 0.38 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} -0.65 \\ 0.65 \\ 0.38 \end{bmatrix}^T ,$$

The estimated tensor corresponded to the correct locations again, and the estimated tensors are

$$\hat{\boldsymbol{T}}_1 = \begin{bmatrix} 0.28 \\ -0.26 \\ 0.92 \end{bmatrix} \begin{bmatrix} 0.98 \end{bmatrix} \begin{bmatrix} 0.28 \\ -0.26 \\ 0.92 \end{bmatrix}^T ,$$

and

$$\hat{\boldsymbol{T}}_2 = \begin{bmatrix} -0.64 \\ 0.65 \\ 0.39 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} -0.64 \\ 0.65 \\ 0.39 \end{bmatrix}^T .$$

Again, the estimates of the tensors are quite accurate: $\boldsymbol{\lambda_1}$ has less than a 2% error, and $\boldsymbol{\lambda_1}$ is exact. The orientation estimate was even more accurate than the first simulation with both targets being oriented within $1°$ of the actual orientation.

5. CONCLUSIONS

An algorithm has been presented to accurately, and efficiently localize and characterize a subterranean magnetic object by directly extracting its 3-dimensional tensor amplitude. The advantages include: large reduction in computer storage, the ability to overcome sensor insensitivity, and the ability to minimize modeling error by not discretizing the orientation parameter space.

Acknowledgment

This work is supported in part by the US Army REDCOM CERDEC Night Vision and Electronic Sensors Directorate, Science and Technology Division, Countermine Branch and in part by the U. S. Army Research Office under grant number W911NF-11-1-0153.

6. REFERENCES

[1] T. Counts, A. C. Gurbuz, W. R. Scott, Jr., J. H. McClellan, and K. Kim, "Multistatic Ground-Penetrating Radar Experiments," *IEEE Trans. Geoscience and Remote Sensing*, vol. 45, no. 8, pp. 2544–2553, 2007.

[2] A. C. Gurbuz, J. H. McClellan, and W. R. Scott, Jr., "A Compressive Sensing Data Acquisition and Imaging Method for Stepped Frequency GPRs," *IEEE Trans. Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 2640–2650, 2009.

[3] A. C. Gurbuz, W. R. Scott, Jr., and J. H. McClellan, "Location estimation using a broadband electromagnetic induction array," *Proc. SPIE*, vol. 7303, pp. 73030U–73030U–9, 2009.

[4] K. R. Krueger, W. R. Scott, Jr., and J. H. McClellan, "Location and orientation estimation of buried targets using electromagnetic induction sensors," *Proc. SPIE*, Apr. 2012.

[5] K. Krueger, J. H. McClellan, and W. R. Scott, Jr., "3-d imaging for ground penetrating radar using compressive sensing with block-toeplitz structures," in *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2012 IEEE 7th*, june 2012, pp. 229 –232.

[6] R. Baraniuk and P. Steeghs, "Compressive radar imaging," *IEEE Radar Conference*, Apr. 2007.

[7] F. J. Herrmann, G. Hennenfent, P. Moghaddam, and D. Wand, "Seismic data processing with curvelets: a multiscale and nonlinear approach," *SEG Annual Meeting*, Sept. 2007.

[8] D. L. Donoho, "Compressed sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[9] S.G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3397 –3415, dec 1993.

[10] C. Ekanadham, D. Tranchina, and E. P. Simoncelli, "Recovery of Sparse Translation-Invariant Signals With Continuous Basis Pursuit," *IEEE Transactions on Signal Processing*, pp. 4735–4744, Oct. 2011.

[11] K. C. Ho and P. D. Gader, "On the estimation of target depth using the single transmit multiple receive metal detector array," *Proc. SPIE*, Apr. 2012.

[12] M. Özdemir, E. L. Miller, and S. Norton, "Localization and characterization of buried objects from multifrequency array inductive data," *Proc. SPIE*, Apr. 1999.

[13] W. R. Scott, Jr. and G. D. Larson, "Measured dipole expansion of discrete relaxations to represent the electromagnetic induction response of buried metal targets," *Proc. SPIE 7664*, pp. 76640E–76640E–11, 2010.

[14] S. P. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[15] M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming.," 2008, `http://cvxr.com/cvx/`.

[16] C. Beck and R. D'Andrea, "Computational study and comparisons of LFT reducibility methods," in *American Control Conference, 1998. Proceedings of the 1998*, jun 1998, vol. 2, pp. 1013 –1017 vol.2.