

Machine learning for the classification of unexploded ordnance (UXO) from electromagnetic data

Lindsey J. Heagy¹, Douglas W. Oldenburg², Fernando Pérez¹ & Laurens Beran³

¹Department of Statistics, University of California Berkeley

²Geophysical Inversion Facility, University of British Columbia

³Black Tusk Geophysics

SUMMARY

Electromagnetic methods are widely used for the detection and classification of unexploded ordnance objects in former war zones or military training grounds. Typically, targets are classified using intrinsic parameters that are estimated via inversion of the observed data. In this work, we present an approach for using convolutional neural networks to classify unexploded ordnance directly from time-domain electromagnetic data. The outputs of the network are probabilities that a given window of signal is associated with an ordnance object, as well as a classification, which is simply the class with the largest probability. We demonstrate our approach with a synthetic example based on a test site in Australia and show that the trained neural network can distinguish between small, medium, and large ordnance objects and clutter. These results illustrate the potential utility of machine learning for the interpretation of electromagnetic data collected over sites contaminated with ordnance.

INTRODUCTION

Unexploded ordnance (UXO) are munitions that were armed, fired and remain unexploded. They are a hazard in countries that have had active wars or military training grounds. In the US alone, it is estimated that there are on the order of 10 million acres of land at over 1400 sites that are contaminated with UXO; the associated clean-up costs are in the 10s of billions (Etter and Delaney, 2003). Improving classification to better identify items which must be dug up, and distinguish them from clutter, which may safely be left in the ground, can significantly reduce these costs.

UXO items have a significant magnetic permeability and are electrically conductive, so they may be detected with magnetic or electromagnetic techniques. For the classification step, we require data that have sufficient information-content to allow us to extract features or estimate parameters that can be used to distinguish between ordnance items and clutter. Multi-component time-domain electromagnetic (TDEM) systems have proven to be well-suited for this task (cf. Bell et al. (2001); Pasion and Oldenburg (2001); Zhang et al. (2003); Billings et al. (2010)).

To interpret the TDEM data and associate an observed signal with an ordnance item, a common approach is to conduct a series of inversions to estimate the location, orientation, and electromagnetic properties of the target and then match these properties to a library of known ordnance objects Andrews and Nelson (2011); Beran et al. (2013). These parametric inversions are non-linear and prone to getting stuck in local minima. Given the success of neural networks for classification tasks, not only for images, but also for scientific data, the question

we pose is: can we construct a neural network to classify ordnance objects directly from TDEM data without performing an inversion? Specifically of interest is to obtain the probability that an observed signal is associated with a given ordnance object.

MODELLING THE ELECTROMAGNETIC RESPONSE OF A UXO

Figure 1 demonstrates the setup of a time-domain electromagnetic experiment to detect and classify an ordnance object. A transmitter generates a time-varying magnetic field \mathbf{h}_T which induces eddy currents in the conductive, permeable ordnance object. These eddy currents produce a secondary magnetic field \mathbf{h}_R . At the receiver location, we measure the time-rate-of-change of the secondary magnetic flux ($\partial \mathbf{b}_R / \partial t = \mu_0 \partial \mathbf{h}_R / \partial t$).

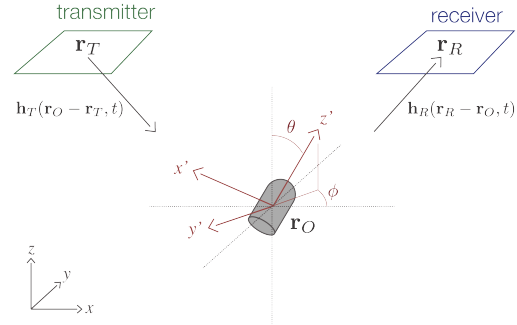


Figure 1: Electromagnetic experiment to detect a UXO. The transmitter produces a primary magnetic field which excites currents in a conductive UXO producing a secondary magnetic field which is measured at the receiver.

To simulate the electromagnetic response of an unexploded ordnance object, we use an approximate forward model, namely that due to the response of a transient magnetic dipole

$$\frac{\partial \mathbf{b}_R}{\partial t}(\mathbf{r}, t) = \frac{1}{r^3} (3\hat{\mathbf{r}}(\hat{\mathbf{r}} \cdot \mathbf{m}(t) - \mathbf{m}(t))) \quad (1)$$

where $\mathbf{r} = r\hat{\mathbf{r}} = \mathbf{r}_R - \mathbf{r}_O$ is the separation between the receiver and the ordnance object (Bell et al., 2001; Pasion and Oldenburg, 2001; Zhang et al., 2003). The moment, $\mathbf{m}(t)$, is given by

$$\mathbf{m}(t) = \mathbf{P}(t) \cdot \mathbf{h}_T(\mathbf{r}_O - \mathbf{r}_T, t) \quad (2)$$

Bell et al. (2001). It depends upon the strength of the primary magnetic field at the location of the ordnance object as well as the properties of object, which are captured in the polarizability tensor

$$\mathbf{P}(t) = \mathbf{E}(\phi, \theta, \psi) \cdot \mathbf{L}(t) \cdot \mathbf{E}^T(\phi, \theta, \psi) \quad (3)$$

The matrix \mathbf{E} is an orthogonal matrix which rotates the coordinate system from the geographic coordinates (x, y, z) in Figure

Machine learning for the classification of UXO

1) to the local, body-centered coordinate system of the ordnance object (x, y, z in Figure 1). The polarization matrix $\mathbf{L}(t)$ is a diagonal matrix whose entries are the principal polarizabilities of the ordnance object (in the x', y', z' coordinates):

$$\mathbf{L}(t) = \begin{pmatrix} L_1(t) & & \\ & L_2(t) & \\ & & L_3(t) \end{pmatrix} \quad (4)$$

These polarizabilities depend upon the physical properties of the ordnance object as well as its geometry. For forward modelling, we use the Python implementation of BTInvert provided by Black Tusk Geophysics.

SURVEY AND DATA

We consider an UltraTEM towed array, with the geometry shown in figure 2 (Billings et al., 2018). The system consists of 5 transmitters and 11 receivers which each measure 3 components of $\partial \mathbf{b} / \partial t$ at 27 time channels between 0.15 ms and 2.5 ms. A sounding collected at a single location of the towed array then consists of 165 time-series EM data ($5 \times 11 \times 3$).

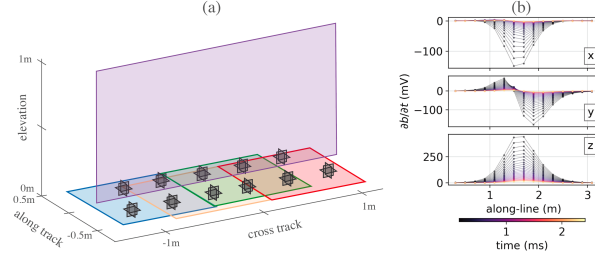


Figure 2: (a) Geometry of the UltraTEM system. Colors indicate transmitters and the grey cubes are the three-component $\partial \mathbf{b} / \partial t$ receivers. (b) A sample of simulated data (one transmitter and one 3-component receiver) over a Medium ISO. The color of the lines indicates the time-channel.

A typical approach for working with these data would be to identify regions with anomalous signal, select a subset of data containing the anomaly, and then perform parametric inversions to estimate the location, orientation, and principal polarizabilities of an object. These polarizabilities are used in the classification step where they are matched with a library of known ordnance polarizabilities.

MACHINE LEARNING APPROACH

Rather than going through an inversion process and then classifying objects, our aim is to build a neural network that, given UltraTEM data, classifies the signal as being associated with an ordnance type, clutter, or background.

Convolutional Neural Networks (CNNs) are widely used in applications for images, audio, and video data (e.g. Krizhevsky et al. (2012); LeCun and Bengio (1995); LeCun et al. (2010)). Convolutional filters exploit spatial and temporal relationships to learn from the data. CNNs are increasingly being adopted for working with geophysical data. For example they have been applied for seismic horizon tracking (Peters and Haber, 2020) and mineral prospectivity mapping (Granek, 2016), among others.

In an image classification problem with color images, the input data would typically include 3 channels of input (red, green and blue), with each channel having an associated $n_x \times n_y$ image. Analogously, for the UltraTEM data, we have 165 input channels (one for each transmitter-receiver pair), and the size of the “image” is the number of along-line data points by number of time-channels (27). Figure 2(b), shows a sample of 3 of the 165 TDEM profiles over a Medium ISO; these are the data that are input to the network. In the following work, we work with 15 along-line data points, which corresponds to a ~ 3 m window. The spacing between measurement points is typical of UltraTEM field data.

The neural network we use consists of 2 convolution layers, followed by a dense layer for classification (also referred to as a fully connected layer). Each of the convolutional layers uses 3 convolutional kernels, batch normalization, a ReLU activation and a max pooling operation. There are 165 input channels, the first convolutional layer outputs 33 channels and the second layer outputs 11. In total, the network has $\sim 75,000$ trainable parameters. We denote a forward pass of the network as

$$\mathbf{s} = \mathcal{F}_{\mathbf{w}}(\mathbf{x}) \quad (5)$$

where \mathbf{x} is a tensor containing the input features, \mathbf{w} are the trainable weights, and \mathbf{s} are the outputs of the network which has k entries where k is the number of classes. The probability that the feature-set \mathbf{x} is associated with class j is computed via the softmax function (cf. Hastie et al. (2009))

$$p(j|\mathbf{s}) = \frac{e^{s_j}}{\sum_k e^{s_k}} \quad (6)$$

The predicted class is then simply the entry with the largest probability

$$y_{\text{pred}} = \arg \max_j p(j|\mathbf{s}) \quad (7)$$

The training problem is an optimization problem in the network weights. The loss function we adopt is the cross entropy loss,

$$\min_{\mathbf{w}} \phi = - \sum_k q_k \log(p_k) \quad (8)$$

which is standard for classification problems. The label is encoded in $q_k \in 0, 1$ which takes the value 0 for all entries except the one corresponding to the true class label (referred to as one-hot encoding). Our implementation uses Pytorch (Paszke et al., 2019), and can be accessed at <https://github.com/simpeg-research/heagy-et-al-2020-uxo-seg>.

To construct the input features \mathbf{x} , we perform two normalization steps on the data, which are illustrated in Figure 3. In the first, we scale the data as a function of time-channel and multiply the data at each time by its time-channel (Figure 3b). The purpose of this step is to amplify the influence of late-time channels in the network while still preserving characteristics of the decay, which are indicative of the principal polarizabilities. In the second step, we scale all 165 inputs by the maximum amplitude across all data so that all inputs are in the range $[-1, 1]$ (Figure 3c). The aim of this step is to force the network to learn spatial and temporal relationships within the data rather than simply the mean amplitude of signal associated with a given ordnance object, which is influenced both by the ordnance type and by the depth of burial.

Machine learning for the classification of UXO

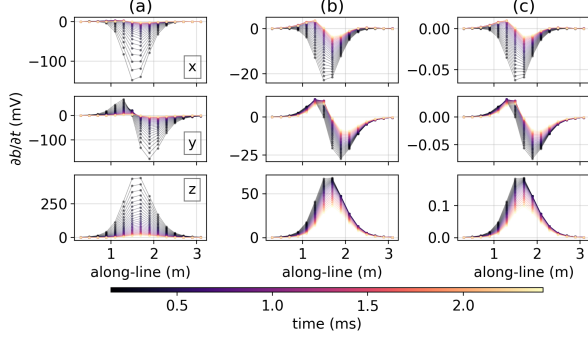


Figure 3: Subset of (a) simulated data, (b) data that have been multiplied by t , the time-channel (step 1 in the normalizations), (c) data from (b) which have been normalized by the maximum amplitude across all 165 inputs.

class	[0, 4]
x-location (m)	[-1.75, 1.75]
y-location (m)	[-0.5, 3.5]
depth (m)	[0, 0.5]
yaw	[0, 2π]
pitch	[0, 2π]
mean noise amplitude (log sampled)	[10^{-5} , 10^{-2}]

Table 1: Range of variable assigned in the training, test, and validation sets.

Training

We consider 5 classes: background, Small ISO, Medium ISO, Large ISO, and clutter. For each realization we randomly assign the class, noise level, and location and orientation of the target object. The range of parameters is shown in table 1. The training set we generate consists of 8192 simulations and the test and validation sets each consist of 1024 realizations. For our noise model, we generate gaussian random noise of the form

$$\varepsilon = a \frac{1}{t} \mathcal{N}(0, 1) \quad (9)$$

where a is the amplitude / variance (in the range specified in Table 1), t is time (ms) and $\mathcal{N}(0, 1)$ is the normal distribution. The functional form of the noise model mimics the $\sim 1/t$ decay expected at intermediate times over a conductive, permeable target (Pasion, 1999) and is consistent with the noise observed at the test plot. Polarizabilities for each ordnance type are specified by the Department of Defence database (Murray et al., 2016). To simulate clutter, we include both: (a) ordnance objects that are not expected at a given site, here we include the polarizabilities for a 20mm munition, and (b) spherical objects, which have $L_1 = L_2 = L_3$; for the following work, we use the polarizabilities associated with a Small ISO. How to design a “clutter” class that includes sufficiently diverse signals that are not associated with targets of interest is an avenue for future development.

We train the network for 10 epochs using stochastic gradient descent and mini-batches of size 32. The training, test, and validation accuracy are 98.7%, 97.6%, and 98.2%, respectively.

To demonstrate the use of the trained classifier for interpreting signal, we consider a single profile of data collected over 3 ob-

jects, a Small ISO ($5m, -0.4m$), a clutter item ($17m, -0.2m$), and a Medium ISO ($26m, -0.7m$). The along-line location of each item is indicated by the squares in Figure 4a (the vertical location of the squares has no meaning). All items are centered in the cross-line direction ($x = 0$). To interpret the signal, we consider a moving 3m window and perform a classification for each window of data. This can be thought of as a coarse-scale segmentation problem in machine learning. The classifications are denoted by the circles in Figure 4a; each circle is the classification of data in a 3m window centered at the location of the dot. To simplify the image, we do not plot classifications of “background” in Figure 4a. Figure 4b shows the associated class probabilities, computed using equation 6.

Near the along-line center of each target, our classifications are correct and the associated probabilities near 1. Of note is that the training set only included simulations of ordnance objects up to 0.5m depth; the Medium ISO in this example is deeper (0.7m) and the classifications near that object are correct. There are misclassifications and lower probabilities near the edge of the anomalous signals. Our training set only contained some simulations where the item of interest was outside the along-line window of signal we are considering, but only to a maximum of $\pm 0.5m$. The misclassifications are in regions where there is still coherent signal, but the network has not seen training data with objects that are further than 0.5m from the window of interest. How best to treat these “edge effects” requires further investigation. We could expand the variability of the training set to include items further away; this would then increase the footprint of the classification. Another alternative is to include simulations of distant objects in the background-class, or perhaps we will require some combination of these approaches.

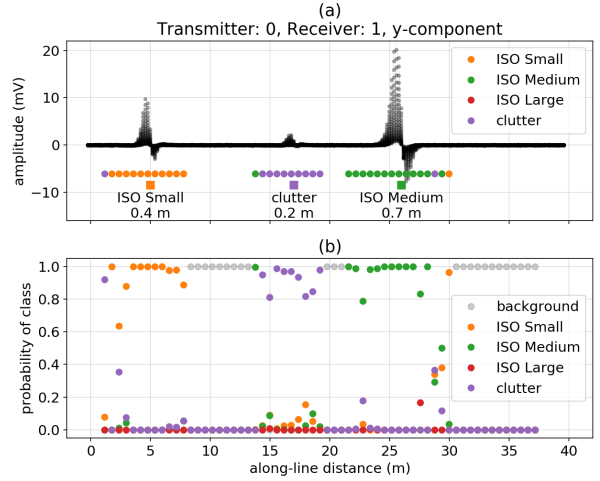


Figure 4: (a) Simulated UltraTEM data collected along a profile line (black) with 3 items whose along-line location is indicated by the squares. The number indicated beneath each item indicates its depth. The colored circles denote the classification obtained from the trained neural network for signal within a 3m window that is centered at the along-line location of the circle. (b) Probabilities associated with each class.

Machine learning for the classification of UXO

SYNTHETIC EXAMPLE

We now consider how this approach could be used to produce maps indicating the probability of a given ordnance item at a field site. We construct a synthetic example based on the data collected at a test plot in Australia. Figure 5 shows the survey lines (grey) and the location and depth of ordnance items and clutter. These locations and depths were drawn for a subset of the items seeded at the test plot. As was done in the previ-

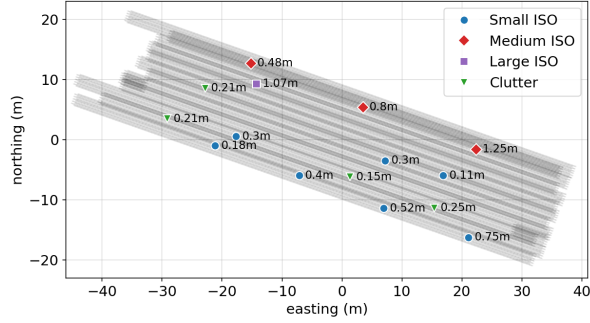


Figure 5: Survey geometry based on the data collected at a test plot in Australia. Ordnance locations are selected from a subset of those seeded at the test plot. The values to the right of the markers indicate the depth of each object.

ous section, we use a sliding 3m window and use the trained network to provide probabilities associated with each of the classes. These probabilities are gridded using the block reduce and spline operations included in `verde` (Uieda, 2018). A map of the probabilities for each item are shown in Figure 6. Overall, there is good agreement between where significant probabilities are assigned and the true spatial location of each target object. In Figure 6b, there is a false-positive near (-15m, 5m) easting. We attribute this to “edge effects”, as we observed in the previous section, due to the Large ISO. Similarly, we see halo-effects in Figure 6d where moderate probabilities are assigned to the clutter class in the vicinity of ordnance objects.

DISCUSSION AND CONCLUSIONS

The problem of classifying unexploded ordnance from electromagnetic data has several characteristics that make it a candidate for the use of CNNs: we are able to construct synthetic labeled training sets from a library of known ordnance objects, there are labeled field data sets, and the data acquired from a given EM system (e.g. the UltraTEM) has a well-defined structure. The synthetic results we have shown demonstrate the potential for neural networks to be a useful interpretation tool to augment inversion-based approaches. The method we present could be used to prioritize areas where a cued sensor may be deployed to collect longer off-time data, or to provide an independent interpretation to complement an inversion result.

This work is in early stages, and there are open questions with respect to constructing robust noise and clutter models, handling multi-object scenarios where two or more ordnance items are in close proximity, as well as addressing challenging geologic scenarios such as settings with magnetic soils which

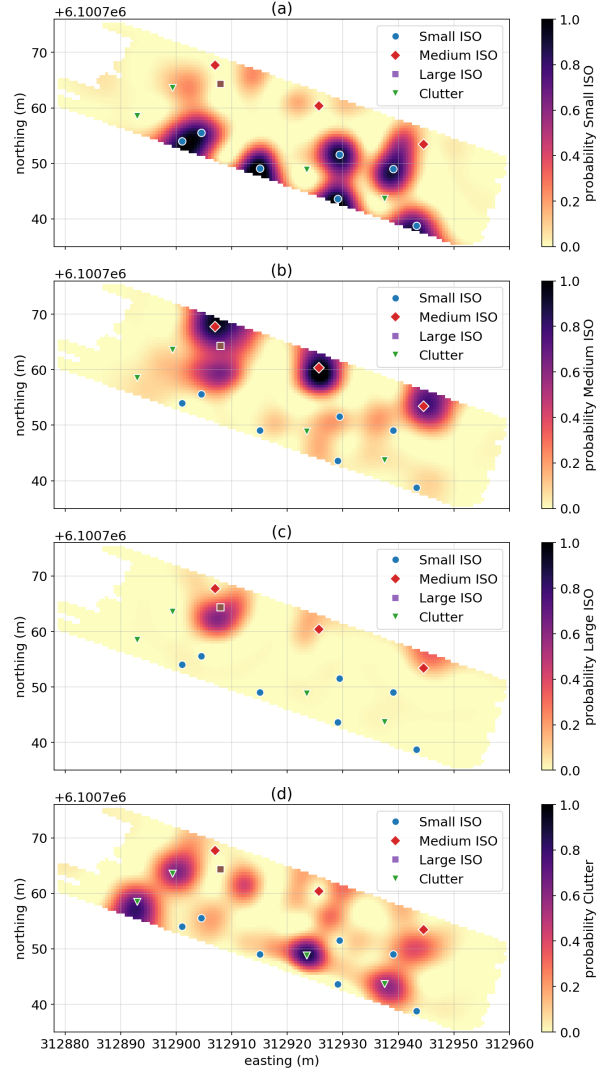


Figure 6: Maps of the probability of each target class: (a) Small ISO, (b) Medium ISO, (c) Large ISO, (d) Clutter.

complicate the EM response. Additionally, we plan to further investigate strategies for designing the network architecture, input features, and the loss function (e.g. incorporating the asymmetry between the consequences of false positives and false negatives). The CNN architecture we presented is specific to the geometry of the UltraTEM system, however, we anticipate that this approach could be readily adapted to other multi-component EM systems.

ACKNOWLEDGEMENTS

The authors are grateful to David Sinex, Len Pasion, Lin-Ping Song, and Nicolas Lhomme for constructive conversations on the setup of this problem, for providing background information in the UltraTEM system, as well as support on the use of BTInvert. We are also grateful to Josh Bloom for conversations on neural network architectures. Thanks to Black Tusk Geophysics for providing data from the test plot which motivated the synthetic example used in this paper, as well as for providing access to BTInvert.

Machine learning for the classification of UXO

REFERENCES

- Andrews, A., and H. Nelson, 2011, Implementing Advanced Classification on Munitions Response.
- Bell, T., B. Barrow, J. Miller, and D. Keiswetter, 2001, Time and Frequency Domain Electromagnetic Induction Signatures of Unexploded Ordnance: Subsurface Sensing Technologies and Applications, **2**, 153–175.
- Beran, L., B. Zelt, and S. Billings, 2013, Detecting and Classifying UXO: Journal of ERW and Mine Action, **1**, 57–63.
- Billings, S., D. Lutes, L. Beran, W. Rowlands, Mitchell, and Steven, 2018, UltraTEM: Flexible transmitter and receiver configurations for buried metal detection in challenging and varied conditions: FastTIMES, **23**, 98–106.
- Billings, S. D., L. R. Pasion, L. Beran, N. Lhomme, L. P. Song, D. W. Oldenburg, K. Kingdon, D. Sinex, and J. Jacobson, 2010, Unexploded ordnance discrimination using magnetic and electromagnetic sensors: Case study from a former military site: Geophysics, **75**.
- Etter, D., and B. Delaney, 2003, Report of the Defense Science Board Task Force on Military Software: Technical report.
- Granek, J., 2016, Application of machine learning algorithms to mineral prospectivity mapping.
- Hastie, T., R. Tibshirani, and J. Friedman, 2009, The elements of statistical learning: data mining, inference, and prediction: Springer Science & Business Media.
- Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012, ImageNet Classification with Deep Convolutional Neural Networks, *in* Advances in Neural Information Processing Systems 25: Curran Associates, Inc., 1097–1105.
- LeCun, Y., and Y. Bengio, 1995, Convolutional Networks for Images, Speech, and Time-Series.
- LeCun, Y., K. Kavukcuoglu, and C. Farabet, 2010, Convolutional networks and applications in vision: Proceedings of 2010 IEEE International Symposium on Circuits and Systems, IEEE, 253–256.
- Murray, C., N. Khadr, T. H. Bell, and D. Steinhurst, 2016, Munitions Classification Library.
- Pasion, L., 1999, Detecting Unexploded Ordnance With Time Domain Electromagnetic Induction: 146.
- Pasion, L. R., and D. W. Oldenburg, 2001, A Discrimination Algorithm for UXO Using Time Domain Electromagnetics: Journal of Environmental and Engineering Geophysics, **6**, 91–102.
- Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, 2019, PyTorch: An Imperative Style, High-Performance Deep Learning Library, *in* Advances in Neural Information Processing Systems 32: Curran Associates, Inc., 8024–8035.
- Peters, B., and E. Haber, 2020, Fully reversible neural networks for large-scale 3D seismic horizon tracking: **1**, 1–7.
- Uieda, L., 2018, Verde: Processing and gridding spatial data using Green’s functions: Journal of Open Source Software, **3**, 957.
- Zhang, Y., L. Collins, H. Yu, C. E. Baum, and L. Carin, 2003, Sensing of unexploded ordnance with magnetometer and induction data: Theory and signal processing: IEEE Transactions on Geoscience and Remote Sensing, **41**, 1005–1015.