

Integrating neural networks in least-squares inversion of airborne time-domain electromagnetic data

Muhammad Rizwan Asif¹, Nikolaj Foged², Pradip Kumar Maurya², Denys James Grombacher², Anders Vest Christiansen², Esben Auken², and Jakob Juul Larsen³

ABSTRACT

Airborne time-domain electromagnetic surveys produce extremely large data sets with thousands of line kilometers of data and millions of possible models to explain the data. Inversion of such data sets to obtain the resistivity structures of the subsurface is computationally intensive and involves calculation of a significant number of forward and derivative responses for solving the least-squares inverse problem. The flight altitude of the airborne system needs to be included in the modeling, which adds further complexity. We propose to integrate neural networks in a damped iterative least-squares inversion framework to expedite the inversion process. We train two separate neural networks to predict the forward responses and partial derivatives independently for a broad range of resistivity structures and flight altitudes. Data inversion is not only used for producing the final subsurface models but also used during data processing, or to produce intermediate

results during a survey. With these purposes in mind, we provide three inversion schemes with a tunable balance between computational time and modeling accuracy: (1) numerical forward responses used initially in combination with neural network derivatives, and the derivatives switched to a numerical solution in final iterations, (2) numerical forward responses in combination with neural network derivatives used throughout the inversion, and (3) only neural network forward responses and derivatives used in inversion. Experiments on field data find that we improve inversion speed without any loss in modeling accuracy with our first approach, whereas the second scheme gives a significant speedup at the cost of minor and often acceptable deviations in the inversion results from the conventional nonlinear inversion. The last approach is the fastest and captures the overall resistivity structures quite well. Therefore, depending on the modeling accuracy, inversion speedup factors of up to 50 are realized by using the proposed schemes.

INTRODUCTION

Airborne electromagnetic (AEM) methods are extensively deployed for the characterization of the resistivity structures of the subsurface and are used for many applications including mineral exploration (Guo et al., 2020), groundwater investigation (Chandra et al., 2019), saltwater intrusion (Pedersen et al., 2017; Gottschalk et al., 2020), and geothermal studies (Foley et al., 2020). For a more thorough introduction to AEM and its applications, refer to Auken et al. (2017) and Yin et al. (2015). In airborne time-domain electromagnetics (ATEMs), a

primary electromagnetic field is generated at a flight altitude of up to 100 m using a transmitter coil, which is attached to an airplane or carried as a sling load by a helicopter. The primary field generates eddy currents in the ground and the decay of the associated magnetic field is recorded as a voltage change in the receiver coil. The electrical resistivity structure is then determined through a nonlinear optimization process referred to as inversion. A simple inverse problem for the transient electromagnetic (TEM) method can be expressed as

$$\mathbf{d}_{\text{obs}} = \mathbf{F}(\mathbf{m}), \quad (1)$$

Manuscript received by the Editor 20 May 2021; revised manuscript received 17 February 2022; published ahead of production 15 March 2022; published online 4 May 2022.

¹Aarhus University, Department of Geoscience, HydroGeophysics Group (HGG), Aarhus, Denmark and Aarhus University, Department of Electrical and Computer Engineering, Aarhus, Denmark. E-mail: rizwanasif@geo.au.dk (corresponding author).

²Aarhus University, Department of Geoscience, HydroGeophysics Group (HGG), Aarhus, Denmark. E-mail: nikolaj.foged@geo.au.dk; pradip.maurya@geo.au.dk; denys.grombacher@geo.au.dk; anders.vest@geo.au.dk; esben.auken@geo.au.dk.

³Aarhus University, Department of Electrical and Computer Engineering, Aarhus, Denmark. E-mail: jjl@ece.au.dk.

© 2022 Society of Exploration Geophysicists. All rights reserved.

where \mathbf{d}_{obs} is the observed data and \mathbf{F} is the forward operator and gives the TEM forward response for the model parameters \mathbf{m} based on the physical laws of electromagnetic field propagation. For an AEM survey, $\mathbf{d}_{\text{obs}} = (d_1, d_2, d_3, \dots, d_M)$ and $\mathbf{m} = (\log(\rho_1), \log(\rho_2), \log(\rho_3), \dots, \log(\rho_N), h)$, where h is the flight altitude. The flight altitude h is generally recorded, e.g., with a laser altimeter, and it is therefore known with an uncertainty for a given time instant. However, we include it as an inversion parameter with the recorded value as a strong prior because an erroneous measured altitude would introduce errors in the shallow subsurface resistivity structures. Hence, if \mathbf{m} and \mathbf{F} are known, calculating an AEM response is a forward problem. Conversely, if \mathbf{d}_{obs} and \mathbf{F} are known, then obtaining \mathbf{m} is a typical inverse problem.

Large-scale AEM surveys are often inverted in a 1D formulation, where each sounding curve is deterministically inverted to a 1D resistivity model. Modern AEM surveys can hold hundreds of thousands of soundings; hence, there are millions of model parameters. Therefore, inversion of such large-scale surveys becomes computationally expensive, mainly due to the repeated calculation of data using the forward model and the partial derivatives that are part of the optimization process. To avoid this computationally intensive process, artificial neural networks (ANNs) can be used to model the relationship between the observed sounding data \mathbf{d}_{obs} and model parameters \mathbf{m} directly. ANNs, inspired by biological neural networks, are computing systems that aim to loosely mimic the processing characteristics of a human brain. Such networks learn to perform tasks by considering examples, without needing to know the physical model of the underlying problem (Chen et al., 2019). ANNs are composed of multiple layers of perceptrons with the perceptrons organized in interconnected layers of neurons commonly known as hidden layers. The input layer gathers input parameters, and the output layer predicts the output signals to which input patterns may map. This is achieved by tuning the weights of the neurons, e.g., by backpropagation (Rumelhart et al., 1985), until the loss function is minimized. ANNs are effective in approximating any arbitrary function to good accuracy (Heaton, 2008; Wu et al., 2020) and have been thoroughly discussed in the literature (Liu et al., 2017; Abiodun et al., 2018, 2019).

AEM data have been directly inverted using shallow feed-forward neural networks (Zhu et al., 2012; Li et al., 2019, 2020b), deep neural networks (Bai et al., 2020; Feng et al., 2020; Noh et al., 2020), convolutional neural networks (Puzyrev and Swidinsky, 2021), and long short-term memory networks (Li et al., 2020a). Different from the above-mentioned approaches that are extremely fast in producing inversion results, we focus on using neural networks to calculate the forward model data and partial derivatives used in the least-squares inversion framework for an AEM setup while taking the flight altitude of the airborne system into account. This allows us to build on well-established inversion algorithms (Auken et al., 2015) and make use of hybrid framework (Christiansen et al., 2016), which both benefit from the existing regularization schemes, e.g., laterally constrained regularization (Auken and Christiansen, 2004; Auken et al., 2005) or spatially constrained regularization (Viezzoli et al., 2008). The approach described here could of course be implemented in any other existing inversion scheme that relies on the computation of forward responses and derivatives.

Although neural networks can be sufficiently accurate in predicting forward responses (Asif et al., 2021a; Bording et al., 2021), the uncertainty is random. Therefore, the neural network forward data

predictions cannot be used directly to compute reliable derivatives for layer resistivities. Hence, we train two independent neural networks to compute forward responses and partial derivatives, respectively, and investigate the performance benefits on predicting ATEM data from the SkyTEM system (Sørensen and Auken, 2004).

This work extends Asif et al. (2021b), who use neural networks to calculate the partial derivatives for the inversion of ground-based TEM data in a standard deterministic way. Here, we specifically deal with the added complexities associated with the effects of flight altitude in airborne systems where the flight altitude needs to be considered during data modeling. The flight altitude also needs to be taken into account during the inversion to deal with erroneous measured altitudes. In addition, the vastly increased dynamic data range due to the varying amplitude of AEM data to flight height causes additional complications.

In this study, we consider the effects of flight altitude while predicting the forward responses and partial derivatives for an airborne setup using neural networks. We also consider the flight altitude as an inversion parameter with the recorded value as a strong prior and model its partial derivative using a neural network during an iterative least-squares inversion to deal with erroneous measured altitudes. To deal with the varying dynamic range in AEM forward modeling due to the effect of flight altitude, an appropriate normalization is chosen to achieve satisfactory accuracy, which ensures high performance in an AEM setup. To improve the performance of neural networks, we propose a novel approach to generate a comprehensive database of resistivity models with realistic subsurface patterns, which is essential to obtain improved inversion results.

The rest of this paper is structured in the following order. First, we discuss the mathematical formulation of the least-squares inverse problem and the details of our neural networks for approximate forward modeling and calculation of partial derivatives, which we term as forwardNet (fNet) and derivativeNet (dNet). Then, we evaluate the performance of our networks and present inversion results on field data by integrating fNet and dNet in the least-squares method. Then, we discuss the limitations and prospects of the proposed schemes. Finally, we give concluding remarks.

PROPOSED METHODOLOGY

The inversion is formulated to obtain a set of model parameters \mathbf{m} that minimizes the cost function $\phi(\mathbf{m})$ as

$$\phi(\mathbf{m}) = \underbrace{\|\mathbf{Q}_d(\mathbf{d}_{\text{obs}} - F(\mathbf{m}))\|_{L_2}^2}_{\text{data misfit}} + \underbrace{\|\mathbf{Q}_p \mathbf{R}_p \mathbf{m}\|_{L_2}^2}_{\text{smoothness constraints}}, \quad (2)$$

where \mathbf{Q}_d is a matrix where the diagonals are the inverse of the data variance and \mathbf{Q}_p is a matrix where the diagonals are the inverse of the variance associated with the constraints given by the roughness matrix \mathbf{R}_p that constrain the model parameters \mathbf{m} in neighboring depth layers. Here, it is assumed that there is no correlated noise, meaning that there are no off-diagonal elements in \mathbf{Q}_d . The correlation between neighboring models is enforced through \mathbf{R}_p . Other approaches to the inversion setup as in equation 2 are possible, for instance, as recently proposed by Bai et al. (2021), where estimates of the modeling errors are included in the inversion scheme without implying smoothness constraints.

To minimize equation 2, we update the model parameters \mathbf{m} iteratively with a Levenberg-Marquardt method:

$$\mathbf{m}_{n+1} = \mathbf{m}_n + [\mathbf{G}_n^T \mathbf{C}_{\text{obs}}^{-1} \mathbf{G}_n + \mathbf{R}_p^T \mathbf{C}_c^{-1} \mathbf{R}_p + \lambda \mathbf{I}]^{-1} \cdot [\mathbf{G}_n^T \mathbf{C}_{\text{obs}}^{-1} (\mathbf{d}_{\text{obs}} - F(\mathbf{m}_n)) + \mathbf{R}_p^T \mathbf{C}_c^{-1} (-\mathbf{R}_p \mathbf{m}_n)], \quad (3)$$

where \mathbf{G}_n is the Jacobian matrix for the n th model, $\mathbf{C}_{\text{obs}}^{-1} = \mathbf{Q}_d^T \mathbf{Q}_d$ is the covariance matrix holding the data uncertainties, $\mathbf{C}_c^{-1} = \mathbf{Q}_p^T \mathbf{Q}_p$ defines the strength of the smoothness constraints, λ is the damping parameter, and \mathbf{I} is the identity matrix. For more details on equations 2 and 3, the readers are referred to [Auken and Christiansen \(2004\)](#) and [Xue et al. \(2020\)](#).

The i th column in the Jacobian matrix \mathbf{G} , equation 4, is constructed by computing partial derivatives in logarithmic space for the model vector \mathbf{m} having t parameters:

$$\mathbf{G}_i = \frac{\partial \log(\mathbf{d})}{\partial \log(m_i)} = \frac{m_i}{\mathbf{d}} \frac{\partial \mathbf{d}}{\partial m_i}, \quad i = 1, 2, \dots, t, \quad (4)$$

where $\partial \mathbf{d}$ is the partial derivative vector computed as symmetric differences between two forward responses separated by a small perturbation for the i th element in model vector \mathbf{m} .

If the matrix inverse in equation 3 is formulated numerically efficiently ([Kirkegaard et al., 2015](#)), it is evident that most of the computation time during a least-squares inversion lies within the calculation of $F(\mathbf{m})$ and in particular \mathbf{G} . If \mathbf{m} has t parameters, $t + 1$ forward computations are required to construct \mathbf{G} using single-sided derivatives. For symmetric (double-sided) derivatives, $2t + 1$ forward computations are required. To avoid the above described computationally intensive process, we deploy neural networks to predict TEM forward responses $F(\mathbf{m})$ and the partial derivatives for \mathbf{G} to be used within the least-squares inversion framework. In the following sections, we discuss the details of our neural networks that have been trained on an Intel Xeon Gold 6132 central processing unit (CPU) at 2.60 GHz with two NVIDIA GeForce RTX 2080 Ti graphics processing units (GPUs) using MATLAB 2019b with the deep learning toolbox.

ForwardNet

The fNet input layer consists of 31 parameters. The first 30 input parameters correspond to the layer resistivities ρ in the model vector \mathbf{m} . Inversions with predefined layer thicknesses, often called multi-layer or smooth inversions, are common in TEM. Therefore, the layer thicknesses are fixed and are not considered as an input parameter. The flight altitude h has a large influence on the TEM forward response, and it is crucial to include this as an input parameter.

The output of fNet is the impulse response, i.e., $\partial \mathbf{B} / \partial t$, for the corresponding inputs. The responses are generated at 108 discrete time gates from 1 ns to 37 ms with exponentially increasing gate widths sampled at 14 gates/decade. Although the actual ATEM data span a narrower time interval, a wider range is used to obtain accurate responses after system-response convolution.

Figure 1 shows the influence of flight altitude on representative TEM responses. We observe that the impact of flight altitude on the TEM response decreases with time and has a larger impact in the conductive case than the resistive case.

Prior to network training, it is advantageous to normalize the inputs and target outputs ([Sola and Sevilla, 1997](#); [Theodoridis and Koutroumbas, 2009](#); [Abiodun et al., 2018](#)). Because the forward response does not vary linearly with resistivity, we consider the logarithmic variations to normalize the layer resistivities ρ which are then scaled between $[a, b]$ as

$$\rho_N = a + \frac{(b - a)(\log_{10}(\rho) - \log_{10}(\rho_{\min}))}{\log_{10}(\rho_{\max}) - \log_{10}(\rho_{\min})}, \quad (5)$$

where ρ_N is the normalized resistivity model of ρ and ρ_{\min} and ρ_{\max} refer to the minimum and maximum resistivity values in the training data set, respectively.

The flight altitude above the ground h is normalized by

$$h_N = a + \frac{(b - a)(h - h_{\min})}{h_{\max} - h_{\min}}, \quad (6)$$

where h_N and h are the normalized and actual flight altitude, respectively; a and b are the same as equation 5; and h_{\min} and h_{\max} are the minimum and maximum flight altitude in the training set, respectively.

The network input for fNet for a common scale of $[-1, 1]$ is defined as a 31×1 vector:

$$\mathbf{X}_{\text{fNet}} = \begin{bmatrix} \rho_N \\ h_N \end{bmatrix}. \quad (7)$$

The target outputs, a vector of 108 $\partial \mathbf{B} / \partial t$ data points, are normalized as

$$\mathbf{Y}_{\text{fNet}} = \frac{\frac{\partial \mathbf{B}}{\partial t} - \boldsymbol{\mu}}{\boldsymbol{\sigma}}, \quad (8)$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the gate-wise mean and standard deviation of the amplitudes of $\partial \mathbf{B} / \partial t$ over the training data set, respectively. This normalization results in positive and negative values, and ensures that each time gate is treated equally. The target outputs are not

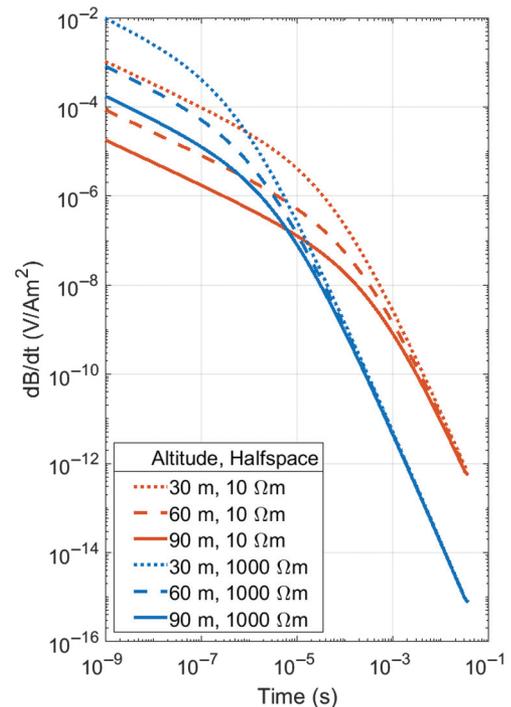


Figure 1. The TEM impulse responses for three different flight altitudes (30, 60, and 90 m). The blue curves represent the 1000 Ωm half-space model and the red curves represent the 10 Ωm half-space model.

scaled between fixed limits $[a, b]$ to allow for the effect of varying dynamic range of the forward responses because of the sensitivity to flight altitude.

Our fNet consists of four layers. The first layer consists of 31 neurons and corresponds to the input variables. The next two layers, i.e., the hidden layers, have 384 neurons each. The depth of the network, i.e., the number of hidden layers, and the number of neurons in each of the hidden layers are selected based on the grid-search method (Liashchynskyi and Liashchynskyi, 2019). The final layer corresponds to the impulse response data points and consists of 108 neurons. The neurons in the hidden layers are interconnected by weights \mathbf{w} and biases \mathbf{b} , which are learnable parameters that control the strength of the connection between neurons.

As our normalized input and output vectors can take positive and negative values, we use the hyperbolic tangent function as the activation function. The hyperbolic tangent function has a wide activation range, and a continuous and steep derivative, which helps in faster learning (Feng and Lu, 2019).

As the goal of fNet is to minimize the error between the numerical forward data \mathbf{Y} and the predicted forward data $\hat{\mathbf{Y}}$, we define the loss function C as the sum of the squares error (SSE). However, fNet uses the fully connected feed-forward neural network, which is prone to overfitting. To avoid overfitting, we introduce a regularization term that causes the network weights \mathbf{w} and biases \mathbf{b} to have smaller values, which forces the network to result in smoother outputs and is less likely to overfit on training data. Hence, the loss function for the training of the neural network is

$$C = (1 - \gamma) \underbrace{\sum_{i=1}^N (\mathbf{Y}_i - \hat{\mathbf{Y}}_i)^2}_{\text{network errors}} + \gamma \underbrace{\left(\frac{1}{n} \sum_{j=1}^n \mathbf{w}_j^2 + \frac{1}{n} \sum_{j=1}^n \mathbf{b}_j^2 \right)}_{\text{regularization term}}. \quad (9)$$

The first term in equation 9 is the sum of the squares error (SSEs), whereas the second term is a regularization term controlled by the performance ratio γ . If γ is too big, the network will not fit the training data adequately. If it is too small, the network may be prone to overfitting. Empirical analysis shows that $\gamma = 10^{-4}$ results in good performance for the problem at hand. The scaled conjugate gradient algorithm (Møller, 1993) with full-batch update (Zheng et al., 2016) is used for backpropagation to minimize the loss function with an early stopping criterion to avoid overfitting.

DerivativeNet

Our dNet input layer consists of 32 parameters. The first 31 input parameters are the same as fNet. We include another parameter, i.e., the layer index i for the layer resistivities $\boldsymbol{\rho}$, at which the perturbation is to be applied as shown in equation 4 which is equivalent to the i th column in the Jacobian matrix. The layer index i is normalized in the same way as shown in equation 6, where i_{\min} and i_{\max} are the minimum and maximum layer in the layer resistivities $\boldsymbol{\rho}$, respectively.

The 32×1 input vector for dNet is defined as

$$\mathbf{x}_{\text{dNet}} = \begin{bmatrix} \boldsymbol{\rho}_N \\ h_N \\ i_N \end{bmatrix}. \quad (10)$$

The output of dNet is the partial derivative on a logarithmic scale with respect to the i th resistivity layer in $\boldsymbol{\rho}$. The partial derivatives are generated as symmetric differences between two forward responses by perturbing the i th resistivity layer for flight altitude h .

The target outputs for dNet, a vector of 108 partial derivative data points, are normalized as

$$\mathbf{Y}_{\text{dNet}} = \frac{\mathbf{G}_t - \mu_G}{\sigma_G}, \quad (11)$$

where μ_G and σ_G are the global mean and standard deviation of the amplitudes of partial derivatives in the training set, respectively. Due to the varying sensitivity of partial derivatives when different layers in a resistivity model are perturbed, the gate-wise mean and standard deviation of the amplitudes of partial derivatives of the training set have not been used as it may affect the smoothness of the normalized partial derivatives.

Because the complexity of dNet is higher than fNet, hyperparameter optimization by the grid-search method reveals that a five-layer deep network having three hidden layers is required to map the partial derivatives with good accuracy. In these five layers, the first layer consists of 32 neurons, which correspond to the input parameters. The next three hidden layers consist of 384 neurons each. The final regression layer, similar to fNet, consists of 108 neurons corresponding to the data points for partial derivatives. The depth of the dNet architecture and the number of neurons in each of the hidden layers of dNet also are selected by the grid-search approach. All other settings for the training of dNet are kept the same as fNet. It is very likely that some other parameterization of the network architecture would yield similar results.

TRAINING AND TESTING DATA SETS

For a neural network that must generalize to a broad range of geologic settings, the training data set having resistivity models should cover the model space with realistic subsurface patterns (Asif et al., 2021a). Initially, we use broadbanded von Kármán covariance functions (Møller et al., 2001; Christiansen and Auken, 2003) to create various 1D resistivity models with geologically realistic structures by varying resistivities, spatial distances, correlation lengths, and amplitudes. Furthermore, shape layering is imposed by stitching together two to five randomly selected depth intervals of the generated von Kármán models. The suite of von Kármán models will contain some resistivity variations and patterns that are unlikely to be resolved due to the resolution limits of the TEM method. In practice, these highly complex resistivity structures often produce TEM data that can be well fitted by significantly simpler resistivity models. Therefore, inclusion of highly complex models, which are unlikely to be resolved and produce data consistent with simpler resistivity structures, would decrease the quality of the training data set.

To improve the quality of the training data set and include the resistivity models having a high degree of ATEM resolvable structures, we first calculate ATEM forward responses of the von Kármán models and then invert the forward responses back to a resistivity model. The obtained resistivity models will primarily hold the structures of the von Kármán models that are evident in the forward data. The von Kármán models are discretized to 90 layers to ensure finer subsurface patterns relative to the resolution of TEM data. However, the inversion is carried out with

a smooth 30-layered model, which is a feasible inversion model setup for ATEM data. The AarhusInv inversion code (Auken et al., 2015) is used for forward and inversion calculations and a 3% uniform data uncertainty is assumed prior to the inversion. Four model examples are shown in Figure 2, displaying the von Kármán model (blue line) and the corresponding training model (red line).

In total, 700,000 1D training models are generated, spanning the resistivity domain from 1 to 1000 Ωm. Out of the 700,000 training models (Figure 3a), 650,000 are used for the training of the network and 50,000 are used as the validation set. Because the validation set is representative of the training set, a small percentage of 1D training models is used. The corresponding forward responses and partial derivatives also are computed by AarhusInv (Auken et al., 2015), where the flight altitude for each training model is uniformly chosen between 10 and 120 m. As shown in Figure 3a, the resistivity distribution for the training models is not perfectly uniform, which is caused by the inversion process in the training model generation. It takes approximately 2 days to generate the training models, the corresponding forward responses, and the partial derivatives. In addition, it takes approximately 5 h each to train fNet and dNet. However, this process needs to be done only once for a given TEM system, which in our case is the SkyTEM system. Once the network is trained, the proposed schemes can be put into practice indefinitely to invert one or more 3D volume data sets for the SkyTEM system.

To evaluate the performance of our neural networks independently, we use the standard inverted resistivity models from two Danish ATEM surveys conducted in Skovby (Figure 3b) and Gribskov (Figure 3c) consisting of approximately 10,000 resistivity models having recorded flight altitudes between 15 and 96 m. However, the final assessment is obtained from the inversion results when the networks are integrated into the least-squares inversion framework where the model parameters are iteratively updated for the Skovby and Gribskov survey data.

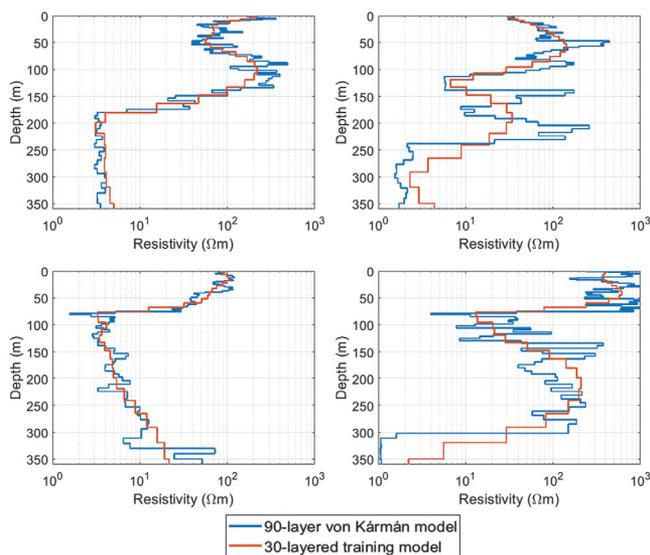


Figure 2. Some examples of 90-layer von Kármán models (blue line) and the corresponding 30-layered training models (red line).

RESULTS

ForwardNet

To evaluate the performance of fNet, we use a 3% relative error metric based on a pragmatic imposed minimum data uncertainty for TEM systems. For 801 inverted models from the Skovby survey data set, all (100%) of the data points of the forward responses from fNet are evaluated to be within a 3% error of the numerical forward responses. For 9219 resistivity models in the Gribskov survey, the percentage of gates within a 3% relative error is evaluated to be 99.5%. A comparison example of the numerical and the fNet predicted forward responses for the given flight altitude is shown in Figure 4.

The performance of our neural network is affected by the degree of variation between the forward data curves, and more variability in these curves results in better performance (Asif et al., 2021a). The variation in the forward data curves decreases by the increase in resistivity in the model parameters as the sensitivity of the data decreases for high-resistivity layers. Therefore, it can be seen in Figure 4 that the fNet errors rise with the increase in resistivity of the subsurface models. In terms of time comparison, the fNet computes at least 1180 forward responses per second and the conventional method only computes approximately 10 forward responses per second. Therefore, a speedup factor of more than 100 is realized in calculating the forward responses when fNet is used.

DerivativeNet

Although fNet achieves high precision in predicting forward responses, it cannot be used directly to compute reliable derivatives for layer resistivities ρ . The reason for this is that the change in response to a small perturbation of a layer resistivity is small. As the error (despite its being small) on the individual data points is uncorrelated, a high numerical inaccuracy on the difference is inevitable.

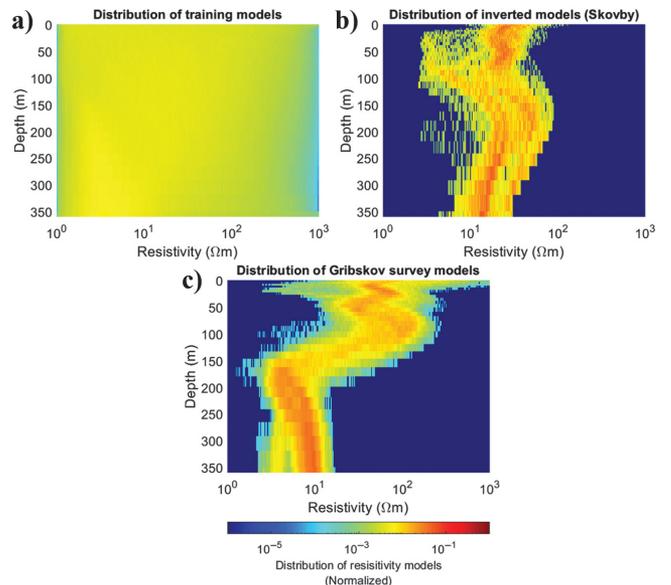


Figure 3. The resistivity models for the training and testing of the network. (a) Distribution of training models; (b) distribution of the Skovby survey, standard inverted models; and (c) distribution of the Gribskov survey, standard inversion models.

Therefore, dNet is used to predict partial derivatives with respect to layer resistivities only. To test the performance of dNet, we generate the partial derivatives by using the numerical forward response to compare it with the output of dNet. To construct the Jacobian matrix \mathbf{G} for one 30-layer model, as shown in equation 4, fNet is evaluated twice to calculate the partial derivative with respect to the flight altitude h , and dNet is evaluated 30 times for layer resistivities ρ . Note that the differences between two fNet calculations when calculating the derivatives for the flight altitude is possible because the sensitivity of the forward response to this parameter is high. Therefore, the errors on individual data points are insignificant when forming the difference.

The sign of partial derivatives in the Jacobian matrix determines the direction of the model update. Therefore, we use the sign of the partial derivative data points as the primary metric for performance evaluation. The sign accuracy of partial derivatives on the Skovby and Gribskov survey resistivity models is evaluated to be 97.4% and 96.7%, respectively. We also present a visual comparison (Figure 5) of partial derivatives constructed by numerical forward responses and the proposed approach for four inverted models from the Gribskov survey. As shown in Figure 5, the derivatives from neural network show a similar pattern and magnitude as the numerically computed derivatives. We also show the absolute relative error of the partial derivatives computed by the two approaches. As for computational efficiency, the dNet achieves a speedup factor of at least 70 by producing more than 1000 derivatives per second in comparison with

the 14 derivatives per second by numerical forward responses for different flight altitudes.

The decisive criterion for the goodness of performance lies in the inversion results. Therefore, in the next section, we evaluate if the accuracy from fNet and dNet is sufficient to achieve a stable iterative solution and similar inversion results as for the standard fully nonlinear least-squares inversion solution. For that purpose, similar to Christiansen et al. (2016) and Asif et al. (2021b), we use four iterative least-squares inversion schemes differing in the calculation of forward responses and partial derivatives:

- 1) Full nonlinear inversion: full nonlinear numerical forward response (fFull) + full nonlinear numerical derivatives (dFull) for all iterations (benchmark inversion).
- 2) Hybrid inversion: fFull + dNet for several iterations until a certain misfit is met, then switching to fFull + dFull for the last iterations.
- 3) dNet inversion: fFull + dNet for all iterations.
- 4) Full neural network inversion: fNet + dNet for all iterations.

Field data inversion

We invert the data from a small Skovby survey (Figure 6a) consisting of approximately 25 km of data and a larger Gribskov survey (Figure 6b) with approximately 300 km of data using a smooth spatially constrained setup (Viezzoli et al., 2008). Both surveys have been conducted in Denmark, where the subsurface is generally composed of glacial sediments (sands, gravels, and clay tills) that sit atop prequaternary deposits. In Skovby, the prequaternary unit underlying the glacial sediments is Paleogene clay. In the Gribskov case, the depth to the prequaternary limestone is approximately 80 m. As shown in the resistivity distributions (Figure 3b and 3c), the top approximately 150 m is less resistive in the Skovby case compared with Gribskov. Deeper parts of the Gribskov survey show very low resistivities due to the intrusion of saltwater. We have chosen these two surveys with their distinct different resistivity regimes to demonstrate that the trained network covers a wide range of resistivity models.

Figure 7 shows the inversion results of the four different schemes masked below the depth of investigation (Christiansen and Auken, 2012) for the cross section marked in Figure 6a. All inversions shown in Figure 7 begin with 30 Ωm half-space models and the data residual (data misfit) for each model is shown. The data residual is calculated as a least-squares difference between the observed data and modeled data in logarithmic space normalized with the data uncertainties. Hence, a data residual of one corresponds to a fit to within one standard deviation of the data uncertainty. The data residual for all of the schemes is computed using fFull to reflect the actual misfit. Visually, all four inversion results in Figure 7 are very similar and would lead to similar geologic interpretation. We also show the ratio (log difference) of the

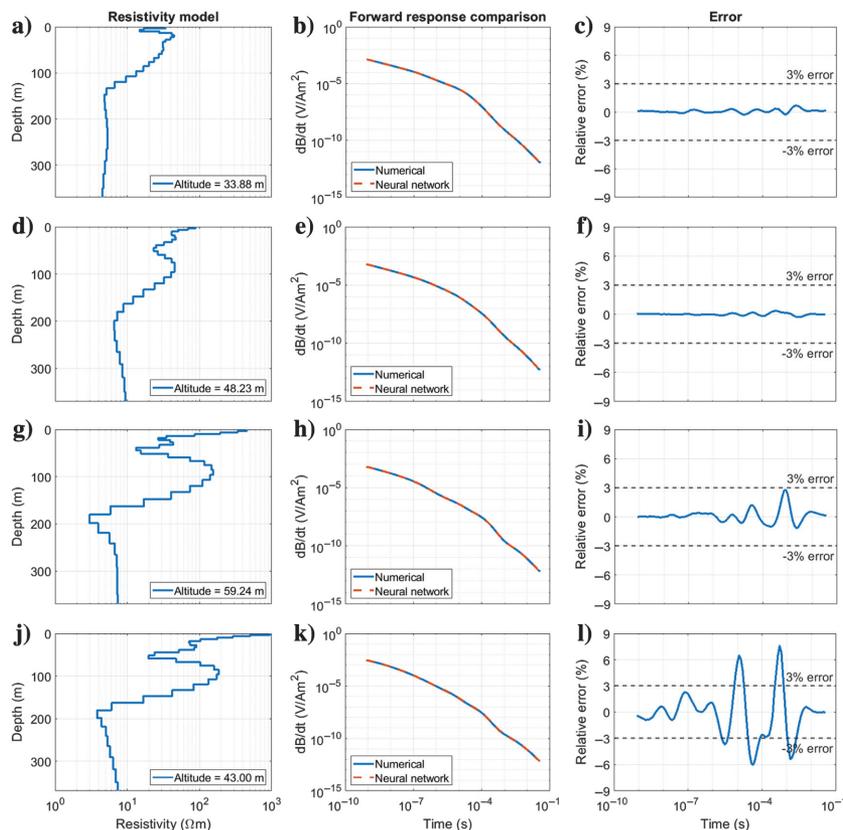


Figure 4. For four resistivity models (a, d, g, and j) from the Gribskov survey, a comparison of the numerical and fNet forward responses (b, e, h, and k) with the relative error of each (c, f, i, and l), respectively. The flight altitude is also shown for each model.

three inversion schemes with respect to the full nonlinear inversion in Figure 8. Quantitatively, the hybrid inversion shown in Figure 7b has a geometric mean of resistivity difference of 1.6% in comparison with the full nonlinear inversion, which is shown in Figure 7a. The dNet inversion (Figure 7c) deviates 0.8%, whereas the neural network inversion (Figure 7d) diverges 2.1% from the full nonlinear inversion. The resistivity differences for the three inversion schemes against the full nonlinear inversion are unbiased.

In Figure 9, we present the noise-normalized data misfits (residuals) of the four inversion schemes for representative early, intermediate, and late time data. It is shown that the full nonlinear, hybrid, and dNet inversion result in similar residuals, whereas the residual for full neural network inversion is generally on the higher side for most cases, which is in agreement with the sounding-by-sounding data misfit shown in Figure 7.

The four inversion results for the cross section marked in Figure 6b from the Gribskov survey are shown in Figure 10. In the case of the Gribskov survey, an 80 Ωm half-space is used as the starting model for all inversions. As expected, the full nonlinear

inversion (Figure 10a) and the hybrid inversion (Figure 10b) deliver almost identical inversion results and data residuals, because they both use the fFull + dFull combination for the final iterations. The geometric mean of resistivity difference of the hybrid inversion to the full nonlinear inversion is 3.9%. The dNet inversion (Figure 10c) has similar resistivity structures and comparable data residuals to the full nonlinear inversion with mean resistivity differences of 4.3%. The data residual for the full neural network inversion (Figure 10d) is slightly higher than the full nonlinear inversion due to additional errors introduced by fNet for high-resistive models. However, the full neural network deviates 5.8%, but captures the overall resistivity structures and patterns quite well. The resistivity differences and the individual data residuals (not shown) display similar trends as shown in Figures 8 and 9.

The data residuals of the full surveys for the different inversion schemes are summarized in Table 1. The data residual for the full survey also shows the same trend as for the selected resistivity cross sections. The full nonlinear, hybrid, and dNet inversions produce comparable data residuals that are fully acceptable as final inversion

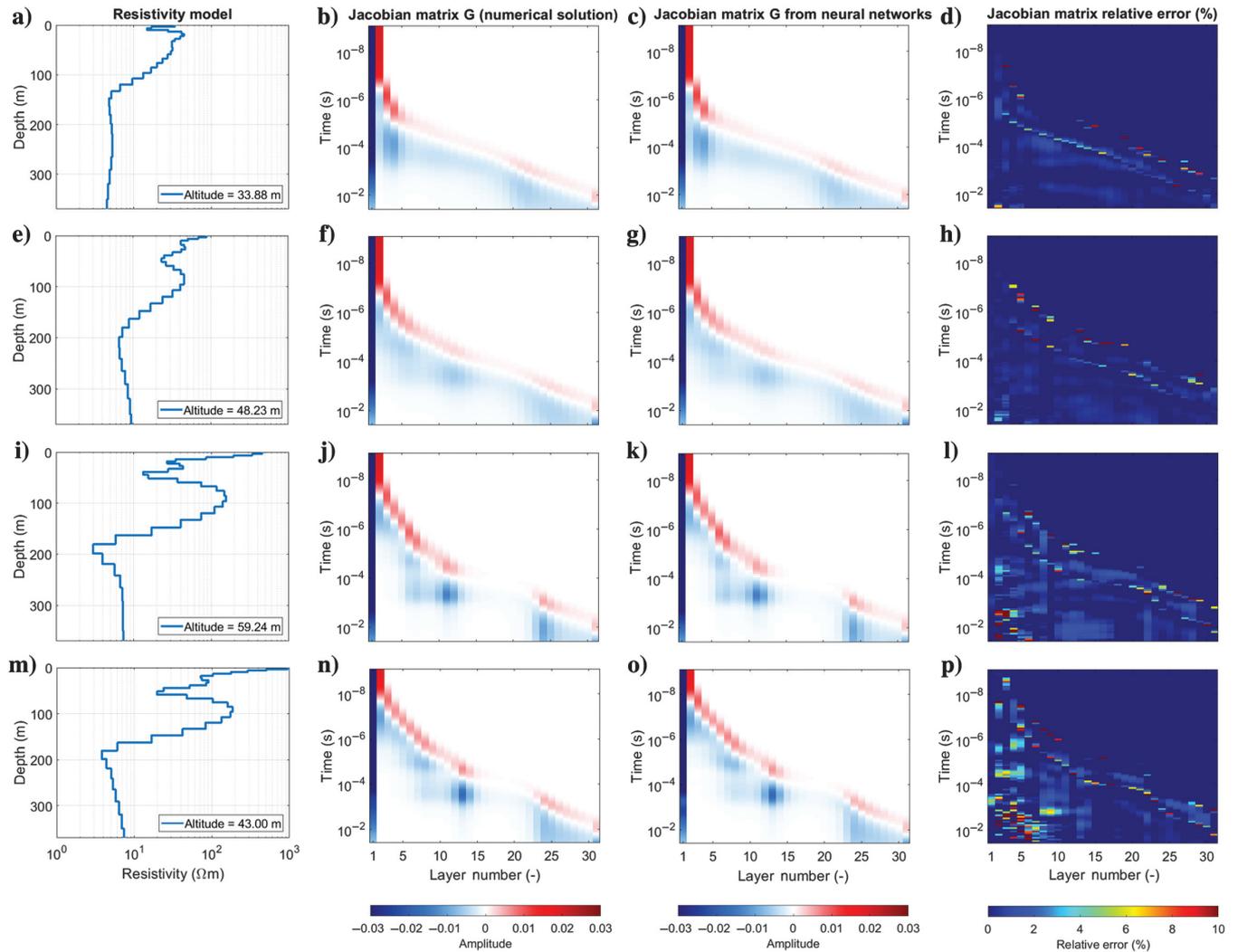


Figure 5. For four resistivity models (a, e, i, and m) from the Gribskov survey, visualization of the Jacobian matrix (derivatives) for the full nonlinear solution (b, f, j, and n) and by fNet and dNet (c, g, k, and o), with the relative error of each (d, h, l, and p), respectively. The flight altitude is shown for each model.

results. For the full neural network inversion of the Skovby survey, the data residual is very close to the full nonlinear inversion, whereas the full neural network inversion for the Gribskov survey results in a larger data residual of 1.02.

Computational time comparison

A detailed time comparison of four inversion schemes on the Skovby and Gribskov field data is provided in Table 1, in which the full neural network inversion scheme is the fastest and achieves a speedup factor of approximately 50 compared with full nonlinear inversion. Each iteration is expedited at least 65 times when fNet and dNet are used together. The dNet inversion achieves a speedup factor of approximately 15 during an inversion, and each inversion iteration is accelerated approximately 17 times when a combination of fFull and dNet is used. The hybrid inversion scheme generally uses one additional iteration to compensate for the errors introduced by dNet. However, it is still at least five times faster than the full nonlinear inversion. Because the inversion process is entirely model dependent, the number of additional iterations varies depending on the data.

It can be seen in Table 1 that the full nonlinear inversion, dNet inversion, and the full neural network inversion take the same number of iterations for the Skovby survey. However, the hybrid inversion uses one additional iteration of full solution on top of the iterations taken by the dNet inversion, resulting in a total of nine iterations. In the case of the Gribskov survey where the data are collected in a

high-resistivity environment, the dNet inversion takes the same number of iterations as the full nonlinear inversion, whereas the hybrid inversion uses one extra iteration of full solution on top of the 11 iterations taken by the dNet inversion. Because the errors in fNet

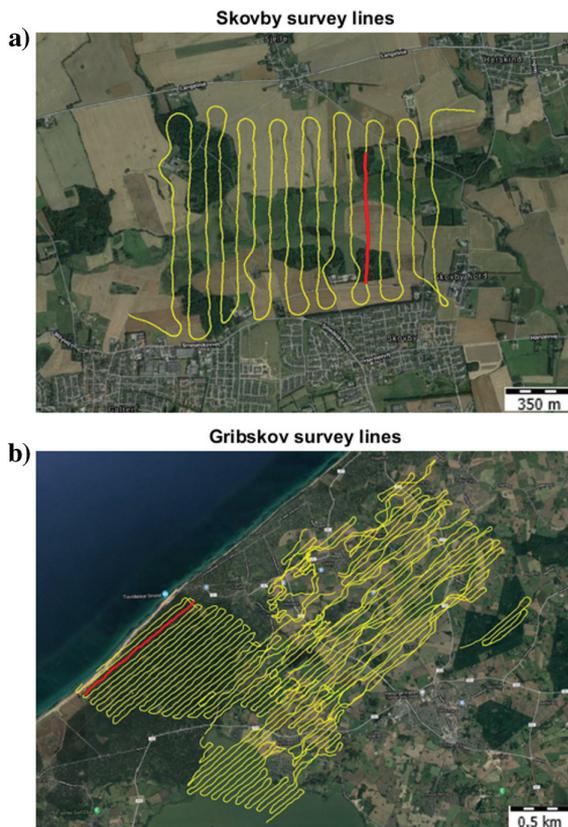


Figure 6. Field survey areas. (a) Skovby survey and (b) Gribskov survey conducted close to a coastline. The survey lines are indicated by the yellow dots and the red lines mark the locations of the cross sections shown in Figures 7 and 8.

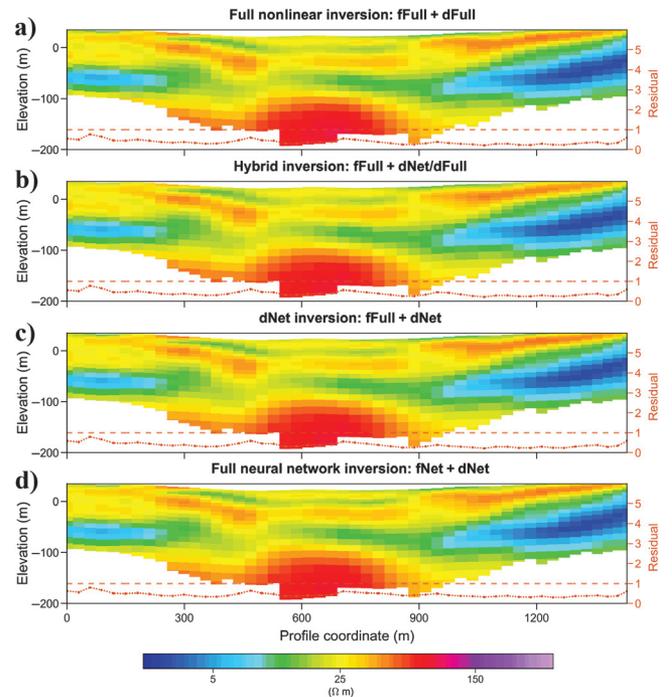


Figure 7. Inversion results of a cross section from the Skovby survey (masked below the depth of investigation) and the corresponding data misfit (red line and right axis). The red hatched line indicates a data misfit of 1.0 being the average fit within the data uncertainty. (a) Full nonlinear inversion, (b) hybrid inversion, (c) dNet inversion, and (d) full neural network inversion.

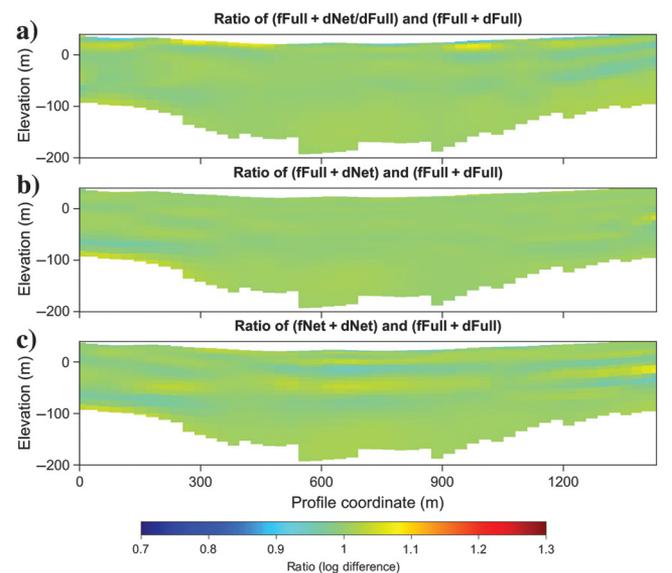


Figure 8. Ratio (log difference) of three inversion schemes with respect to the full nonlinear inversion. (a) Ratio with respect to hybrid inversion, (b) ratio with respect to dNet inversion, and (c) ratio with respect to full neural network inversion.

are higher for high-resistivity environments, the full neural network inversion takes two additional iterations as compared with the full nonlinear inversion and converges in 13 iterations.

DISCUSSION

Investigation of the inversion results of AEM data from two field surveys possessing distinct geologies shows that the hybrid and the dNet inversion schemes result in essentially the same inversion results as the full nonlinear inversion. However, the dNet inversion achieves a considerable speedup as compared with the hybrid inversion framework. The full neural network inversion is the fastest and results in some deviations for high-resistivity regions but captures all the features of the subsurface geology quite well. It can be deployed safely if prior knowledge about the geology of the survey area shows it to be relatively conductive. The full neural network solution also can be effective for applications that have some flexibility in modeling accuracy and geologic interpretation.

Because higher deviation is seen in the inversion results for neural network-based inversion schemes for high-resistivity environments, the accuracy of fNet and dNet may be improved by using, e.g., convolutional neural networks for complex feature mapping. However, it may require deep network architecture and a larger training set that would increase the computational complexity.

The computational time for the comparison of the inversion results is calculated on a single CPU core. However, GPUs are renowned for powerful parallel computing. Therefore, the computational performance could drastically improve by using GPUs for fNet and dNet. It is important to note here that the system response convolution takes more than half of the computation time during the computation of fNet and dNet. Therefore, the optimization of the system response convolution would decrease the computational complexity for further improvement in the inversion process.

Our fNet and dNet are trained on time gates ranging from 1 ns to 37 ms, which is significantly larger than the normal range of ATEM data time gates. During an inversion, the networks' output is interpolated for the gate times at which evaluation is required for a specific survey. Therefore, fNet and dNet require no modification for changing the number of gates or gate times as long as the acquisition interval is within the gate-time range on which the networks are trained. However, our networks are trained for a resistivity range of 1–1000 Ωm. If the resistivity range goes beyond this limit, the network may require retraining on an improved training data set to reach the required level of accuracy.

It is shown in Asif et al. (2021b) that training a network for forward modeling and partial derivative calculations for a ground-based TEM system on smooth 1D resistivity models results in satisfactory performance if a sharp inversion scheme (Vignoli et al., 2015) is used. Therefore, the proposed approach also is expected to work well for such inversion schemes despite being trained on smooth resistivity models. However, the performance accuracy may be improved further for sharp regularization schemes if the network is trained on sharp resistivity models, which is beyond the scope of this work. Nevertheless, it might require a different network parameterization that can be optimized by the grid-search method. This also is true if the proposed approach is to be extended for other TEM systems.

The proposed combination of two independent neural networks within the least-squares inversion framework helps in mitigating several limitations of existing neural network-based 1D AEM inversion

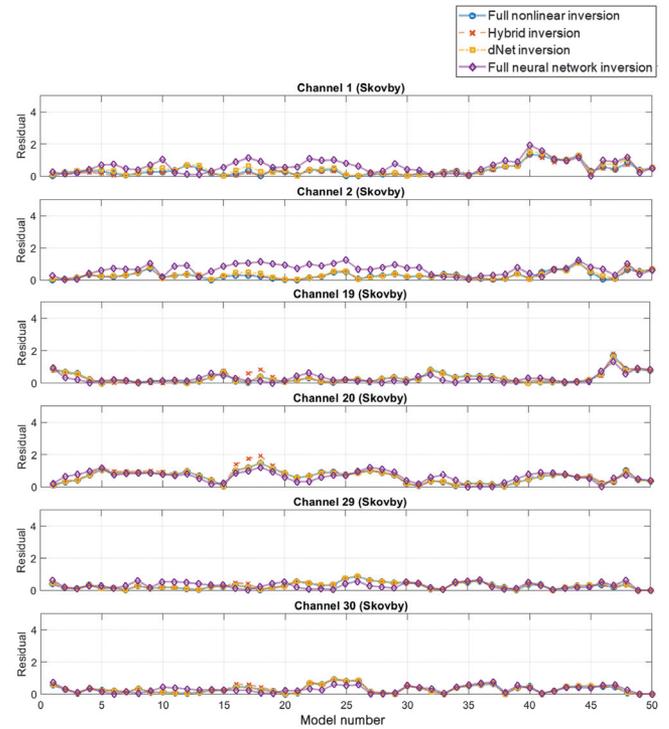


Figure 9. Data residuals for individual channels/times for representative early, intermediate, and late times of the four inversion schemes.

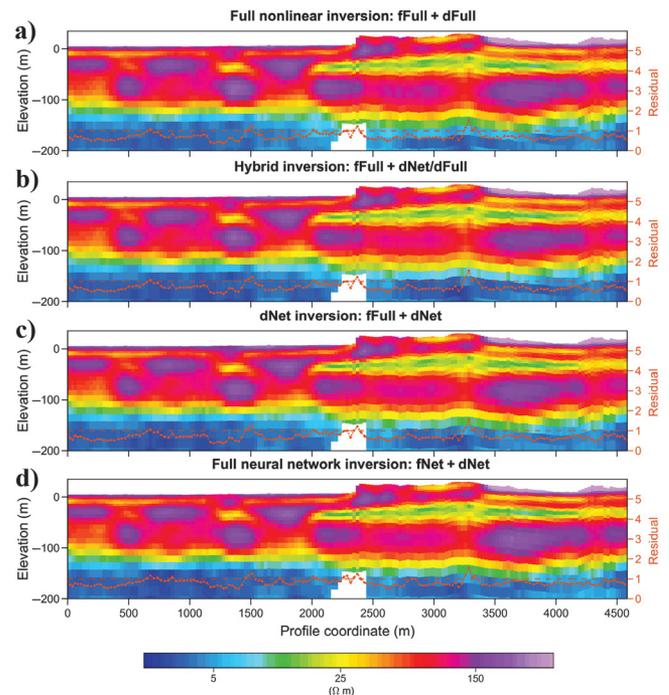


Figure 10. Inversion results of a cross section from the Gribkov survey (masked below the depth of investigation) and the corresponding data residual (red line and right axis). The red hatched line indicates a data misfit of 1.0 being the average fit within the data uncertainty. (a) Full nonlinear inversion, (b) hybrid inversion, (c) dNet inversion, and (d) full neural network inversion.

Table 1. Comparison of presented inversion schemes (on one CPU core).

Survey area	Inversion scheme	Iteration count	Inversion time		Data misfit
			Minutes	% ⁵	
Skovby survey (801 models)	Full nonlinear inversion	8	249	100	0.44
	Hybrid inversion	9	45	18	0.44
	dNet inversion	8	14	6	0.44
	Full neural network inversion	8	5	2	0.47
Gribskov survey (9219 models)	Full nonlinear inversion	11	3759	100	0.85
	Hybrid inversion	12	682	18	0.89
	dNet inversion	11	258	7	0.89
	Full neural network inversion	13	113	3	1.02

⁵Percentages of the full nonlinear inversion time.

methods. Traditional methods, which use one network to invert AEM data directly, require network retraining when any of the system settings are changed, e.g., the transmitter waveform and receiver coil characteristics. In addition, each data sounding and the 1D resistivity models are considered independently, which limits their adaptation to gradient-based inversion routines that use various regularization schemes. However, the proposed approach allows flexibility to various regularization schemes and ATEM system characteristics without the need of network retraining while significantly improving the computational efficiency of existing inversion algorithms. Moreover, the data uncertainty also is taken into account by keeping the least-squares inversion framework.

One of the limitations of the proposed study is the inversion with fixed layer thicknesses. A generalized approach would consider variable model thickness in the training of the neural network, which is to be considered in future work. Another limitation is the applicability to 1D inversion. Therefore, the proposed approach inherits the general limitations of 1D inversion schemes. In addition, the data used in this study are obtained from an ATEM system where the flight altitude is taken into account during forward and derivative calculations and considered as an inversion parameter. However, the proposed schemes would still be effective if the flight altitude is excluded as an inversion parameter.

CONCLUSION

We have demonstrated the use of neural networks within a least-squares inversion framework for an ATEM system. Neural network-based forward modeling and partial derivative calculation opens the possibility of faster inversions with little to no loss in inversion precision. By integrating neural networks in the well-established least-squares inversion framework, the coherence between the inverted models is preserved and the noise in the data is handled explicitly. In addition, by using the impulse responses for the training of our neural networks, retraining is avoided if system settings, e.g., the waveform and/or receiver coil characteristics, are changed. Field AEM data collected at different flight altitudes show that the full nonlinear inversion, the hybrid approach, and the dNet inversion give identical results, whereas the full neural network inversion results in some deviations in high-resistivity environments but is significantly faster. Because these inversion schemes have different

levels of modeling inaccuracies during an inversion, a tunable balance between computational complexity and modeling accuracy may be achieved as geophysical applications have varying degrees of tolerance in modeling accuracy and geophysical interpretation.

DATA AND MATERIALS AVAILABILITY

Data associated with this research are available and can be obtained by contacting the corresponding author.

REFERENCES

- Abiodun, O. I., A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, 2018, State-of-the-art in artificial neural network applications: A survey: *Heliyon*, **4**, e00938, doi: [10.1016/j.heliyon.2018.e00938](https://doi.org/10.1016/j.heliyon.2018.e00938).
- Abiodun, O. I., A. Jantan, A. E. Omolara, K. V. Dada, A. M. Umar, O. U. Linus, H. Arshad, A. A. Kazaure, U. Gana, and M. U. Kiru, 2019, Comprehensive review of artificial neural network applications to pattern recognition: *IEEE Access*, **7**, 158820–158846, doi: [10.1109/ACCESS.2019.2945545](https://doi.org/10.1109/ACCESS.2019.2945545).
- Asif, M. R., T. S. Bording, A. S. Barfod, D. J. Grombacher, P. K. Maurya, A. V. Christiansen, E. Auken, and J. J. Larsen, 2021a, Effect of data preprocessing on the performance of neural networks for 1-D transient electromagnetic forward modelling: *IEEE Access*, **9**, 34635–34646, doi: [10.1109/ACCESS.2021.3061761](https://doi.org/10.1109/ACCESS.2021.3061761).
- Asif, M. R., T. S. Bording, P. K. Maurya, B. Zhang, G. Fiandaca, D. J. Grombacher, A. V. Christiansen, E. Auken, and J. J. Larsen, 2021b, A neural network-based hybrid framework for least-squares inversion of transient electromagnetic data: *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 4503610, doi: [10.1109/TGRS.2021.3076121](https://doi.org/10.1109/TGRS.2021.3076121).
- Auken, E., T. Boesen, and A. V. Christiansen, 2017, A review of airborne electromagnetic methods with focus on geotechnical and hydrological applications from 2007 to 2017: *Advances in Geophysics*, **58**, 47–93, doi: [10.1016/bs.agph.2017.10.002](https://doi.org/10.1016/bs.agph.2017.10.002).
- Auken, E., and A. V. Christiansen, 2004, Layered and laterally constrained 2D inversion of resistivity data: *Geophysics*, **69**, 752–761, doi: [10.1190/1.1759461](https://doi.org/10.1190/1.1759461).
- Auken, E., A. V. Christiansen, B. H. Jacobsen, N. Foged, and K. I. Sørensen, 2005, Piecewise 1D laterally constrained inversion of resistivity data: *Geophysical Prospecting*, **53**, 497–506, doi: [10.1111/j.1365-2478.2005.00486.x](https://doi.org/10.1111/j.1365-2478.2005.00486.x).
- Auken, E., A. V. Christiansen, C. Kirkegaard, G. Fiandaca, C. Schamper, A. A. Behroozmand, A. Binley, E. Nielsen, F. Effersø, and N. B. Christensen, 2015, An overview of a highly versatile forward and stable inverse algorithm for airborne, ground-based and borehole electromagnetic and electric data: *Exploration Geophysics*, **46**, 223–235, doi: [10.1071/EG13097](https://doi.org/10.1071/EG13097).
- Bai, P., G. Vignoli, and T. M. Hansen, 2021, 1D stochastic inversion of airborne time-domain electromagnetic data with realistic prior and accounting for the forward modeling error: *Remote Sensing*, **13**, 3881, doi: [10.3390/rs13193881](https://doi.org/10.3390/rs13193881).

- Bai, P., G. Vignoli, A. Viezzoli, J. Nevalainen, and G. Vacca, 2020, (Quasi)-real-time inversion of airborne time-domain electromagnetic data via artificial neural network: *Remote Sensing*, **12**, 3440, doi: [10.3390/rs12203440](https://doi.org/10.3390/rs12203440).
- Bording, T. S., M. R. Asif, A. S. Barfod, J. J. Larsen, B. Zhang, D. J. Grombacher, A. V. Christiansen, K. W. Engebretsen, J. B. Pedersen, P. K. Maurya, and E. Auken, 2021, Machine learning based fast forward modelling of ground-based time-domain electromagnetic data: *Journal of Applied Geophysics*, **187**, 104290, doi: [10.1016/j.jappgeo.2021.104290](https://doi.org/10.1016/j.jappgeo.2021.104290).
- Chandra, S., E. Auken, P. K. Maurya, S. Ahmed, and S. K. Verma, 2019, Large scale mapping of fractures and groundwater pathways in crystalline hardrock by AEM: *Scientific Reports*, **9**, 1–11, doi: [10.1038/s41598-018-36153-1](https://doi.org/10.1038/s41598-018-36153-1).
- Chen, Y. Y., Y. H. Lin, C. C. Kung, M. H. Chung, and I. Yen, 2019, Design and implementation of cloud analytics-assisted smart power meters considering advanced artificial intelligence as edge analytics in demand-side management for smart homes: *Sensors*, **19**, 2047, doi: [10.3390/s19092047](https://doi.org/10.3390/s19092047).
- Christiansen, A. V., and E. Auken, 2003, Layered 2-D inversion of profile data, evaluated using stochastic models: ASEG, Extended Abstracts, 1–8, doi: [10.1071/ASEG2003_3DEMab005](https://doi.org/10.1071/ASEG2003_3DEMab005).
- Christiansen, A. V., and E. Auken, 2012, A global measure for depth of investigation: *Geophysics*, **77**, no. 4, WB171–WB177, doi: [10.1190/geo2011-0393.1](https://doi.org/10.1190/geo2011-0393.1).
- Christiansen, A. V., E. Auken, C. Kirkegaard, C. Schamper, and G. Vignoli, 2016, An efficient hybrid scheme for fast and accurate inversion of airborne transient electromagnetic data: *Exploration Geophysics*, **47**, 323–330, doi: [10.1071/EG14121](https://doi.org/10.1071/EG14121).
- Feng, B., J. F. Zhang, D. Li, and Y. Bai, 2020, Resistivity-depth imaging with the airborne transient electromagnetic method based on an artificial neural network: *Journal of Environmental and Engineering Geophysics*, **25**, 355–368, doi: [10.32389/JEEG19-087](https://doi.org/10.32389/JEEG19-087).
- Feng, J., and S. Lu, 2019, Performance analysis of various activation functions in artificial neural networks: *Journal of Physics: Conference Series*, **1237**, 022030, doi: [10.1088/1742-6596/1237/2/022030](https://doi.org/10.1088/1742-6596/1237/2/022030).
- Foley, N., S. Tulaczyk, E. Auken, D. Grombacher, J. Mikucki, N. Foged, K. Myers, H. Dugan, P. T. Doran, and R. A. Virginia, 2020, Mapping geothermal heat flux using permafrost thickness constrained by airborne electromagnetic surveys on the western coast of Ross Island, Antarctica: *Exploration Geophysics*, **51**, 84–93, doi: [10.1080/08123985.2019.1651618](https://doi.org/10.1080/08123985.2019.1651618).
- Gottschalk, I., R. Knight, T. Asch, J. Abraham, and J. Cannia, 2020, Using an airborne electromagnetic method to map saltwater intrusion in the northern Salinas Valley, California: *Geophysics*, **85**, no. 4, B119–B131, doi: [10.1190/geo2019-0272.1](https://doi.org/10.1190/geo2019-0272.1).
- Guo, Z., G. Xue, J. Liu, and X. Wu, 2020, Electromagnetic methods for mineral exploration in China: A review: *Ore Geology Reviews*, **118**, 103357, doi: [10.1016/j.oregeorev.2020.103357](https://doi.org/10.1016/j.oregeorev.2020.103357).
- Heaton, J., 2008, Introduction to neural networks with Java: Heaton Research, Inc.
- Kirkegaard, C., K. Andersen, T. Boesen, A. V. Christiansen, E. Auken, and G. Fiandaca, 2015, Utilizing massively parallel co-processors in the AarhusInv 1D forward and inverse AEM modelling code: ASEG, Extended Abstracts, 1–3, doi: [10.1071/ASEG2015ab125](https://doi.org/10.1071/ASEG2015ab125).
- Li, J., Y. Liu, C. Yin, X. Ren, and Y. Su, 2020a, Fast imaging of time-domain airborne EM data using deep learning technology: *Geophysics*, **85**, no. 5, E163–E170, doi: [10.1190/geo2019-0015.1](https://doi.org/10.1190/geo2019-0015.1).
- Li, R., H. Zhang, N. Yu, R. Li, and Q. Zhuang, 2019, A fast approximation for 1-D inversion of transient electromagnetic data by using a back propagation neural network and improved particle swarm optimization: *Nonlinear Processes in Geophysics*, **26**, 445–456, doi: [10.5194/npg-26-445-2019](https://doi.org/10.5194/npg-26-445-2019).
- Li, R., H. Zhang, Q. Zhuang, R. Li, and Y. Chen, 2020b, BP neural network and improved differential evolution for transient electromagnetic inversion: *Computers & Geosciences*, **137**, 104434, doi: [10.1016/j.cageo.2020.104434](https://doi.org/10.1016/j.cageo.2020.104434).
- Liashchynskiy, P., and P. Liashchynskiy, 2019, Grid search, random search, genetic algorithm: A big comparison for NAS: arXiv preprint arXiv:1912.06059.
- Liu, W., Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, 2017, A survey of deep neural network architectures and their applications: *Neurocomputing*, **234**, 11–26, doi: [10.1016/j.neucom.2016.12.038](https://doi.org/10.1016/j.neucom.2016.12.038).
- Møller, I., B. H. Jacobsen, and N. B. Christensen, 2001, Rapid inversion of 2-D geoelectrical data by multichannel deconvolution: *Geophysics*, **66**, 800–808, doi: [10.1190/1.1444969](https://doi.org/10.1190/1.1444969).
- Møller, M. F., 1993, A scaled conjugate gradient algorithm for fast supervised learning: *Neural Networks*, **6**, 525–533, doi: [10.1016/S0893-6080\(05\)80056-5](https://doi.org/10.1016/S0893-6080(05)80056-5).
- Noh, K., D. Yoon, and J. Byun, 2020, Imaging subsurface resistivity structure from airborne electromagnetic induction data using deep neural network: *Exploration Geophysics*, **51**, 214–220, doi: [10.1080/08123985.2019.1668240](https://doi.org/10.1080/08123985.2019.1668240).
- Pedersen, J. B., F. W. Schaars, A. V. Christiansen, N. Foged, C. Schamper, H. Rolf, and E. Auken, 2017, Mapping the fresh-saltwater interface in the coastal zone using high-resolution airborne electromagnetics: *First Break*, **35**, 57–61, doi: [10.3997/1365-2397.35.8.89806](https://doi.org/10.3997/1365-2397.35.8.89806).
- Puzrev, V., and A. Swidinsky, 2021, Inversion of 1D frequency- and time-domain electromagnetic data with convolutional neural networks: *Computers & Geosciences*, **149**, 104681, doi: [10.1016/j.cageo.2020.104681](https://doi.org/10.1016/j.cageo.2020.104681).
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams, 1985, Learning internal representations by error propagation: California Univ San Diego La Jolla Inst for Cognitive Science.
- Sola, J., and J. Sevilla, 1997, Importance of input data normalization for the application of neural networks to complex industrial problems: *IEEE Transactions on Nuclear Science*, **44**, 1464–1468, doi: [10.1109/23.589532](https://doi.org/10.1109/23.589532).
- Sørensen, K. I., and E. Auken, 2004, SkyTEM — A new high-resolution helicopter transient electromagnetic system: *Exploration Geophysics*, **35**, 191–199, doi: [10.1071/EG04194](https://doi.org/10.1071/EG04194).
- Theodoridis, S., and K. Koutroumbas, 2009, Pattern recognition: Academic Press.
- Viezzoli, A., A. V. Christiansen, E. Auken, and K. Sørensen, 2008, Quasi-3D modeling of airborne TEM data by spatially constrained inversion: *Geophysics*, **73**, no. 3, F105–F113, doi: [10.1190/1.2895521](https://doi.org/10.1190/1.2895521).
- Vignoli, G., G. Fiandaca, A. V. Christiansen, C. Kirkegaard, and E. Auken, 2015, Sharp spatially constrained inversion with applications to transient electromagnetic data: *Geophysical Prospecting*, **63**, 243–255, doi: [10.1111/1365-2478.12185](https://doi.org/10.1111/1365-2478.12185).
- Wu, X., G. Xue, Y. He, and J. Xue, 2020, Removal of multisource noise in airborne electromagnetic data based on deep learning: *Geophysics*, **85**, no. 6, B207–B222, doi: [10.1190/geo2019-0555.1](https://doi.org/10.1190/geo2019-0555.1).
- Xue, G., H. Li, Y. He, J. Xue, and X. Wu, 2020, Development of the inversion method for transient electromagnetic data: *IEEE Access*, **8**, 146172–146181, doi: [10.1109/ACCESS.2020.3013626](https://doi.org/10.1109/ACCESS.2020.3013626).
- Yin, C. C., X. Y. Ren, Y. H. Liu, Y. F. Qi, C. K. Qiu, and C. Jing, 2015, Review on airborne electromagnetic inverse theory and applications: *Geophysics*, **80**, no. 4, W17–W31, doi: [10.1190/geo2014-0544.1](https://doi.org/10.1190/geo2014-0544.1).
- Zheng, J., Q. Ma, and W. Zhou, 2016, Performance comparison of full-batch BP and mini-batch BP algorithm on spark framework: 8th International Conference on Wireless Communications & Signal Processing (WCSP), 1–5.
- Zhu, K. G., M. Y. Ma, H. W. Che, E. W. Yang, Y. J. Ji, S. B. Yu, and J. Lin, 2012, PC-based artificial neural network inversion for airborne time-domain electromagnetic data: *Journal of Applied Geophysics*, **9**, 1–8, doi: [10.1007/s11770-012-0307-7](https://doi.org/10.1007/s11770-012-0307-7).

Biographies and photographs of the authors are not available.