Spatial-Temporal Densely Connected CNN-LSTM Nets: An Application to CO₂ Leakage Detection

Zheng Zhou¹, Youzuo Lin¹,*, Zhongping Zhang¹, Yue Wu¹, Zan Wang², Robert Dilmore², and George Guthrie¹

¹ : Earth and Environment Sciences Division, Los Alamos National Laboratory, Los Alamos, NM 87545;
² : National Energy Technology Laboratory, United States Department of Energy, Pittsburgh, PA 15236, USA.

Correspondence *: ylin@lanl.gov.

SUMMARY

In carbon capture and sequestration (also known as carbon capture and storage, or CCS), developing effective monitoring methods is needed to detect and respond to CO₂ leakage. CO₂ leakage detection methods rely on geophysical observations and monitoring sensor network. However, traditional methods usually require development of site-specific physical models and expert interpretation, and the effectiveness of these methods can be limited to the different application locations, operational scenarios, and conditions. In this paper, we developed a novel data-driven leakage detection method based on densely connected convolutional neural networks. Our method is an end-to-end detection approach, that differs from conventional leakage monitoring methods by directly learning a mapping relationship between seismic data and the CO₂ leakage mass. To account for the spatial and temporal characteristics of seismic data, our novel networks architecture combines 1D and 2D convolutional neural networks. To overcome the computational expense of solving optimization problems, we apply a densely-connecting strategy in our network architecture that reduces the number of network parameters. Based on the features generated by our convolutional neural networks, we further incorporate a long short-term memory network to utilize time-sequential information, which further improves the de-
tection accuracy. Finally, we employ our detection method to synthetic seismic datasets generated based on flow simulations of a hypothetical CO$_2$ storage scenario with injection into a partially compartmentalized sandstone storage reservoir. To evaluate method performance, we conducted multiple experiments including a random leakage test, a sequential test, and a robustness test. Numerical results show that our CO$_2$ leakage detection method successfully detects the leakage and accurately predicts the leakage mass, suggesting that it has potential for application in monitoring of real CO$_2$ storage sites.

**Key words:** Carbon Capture and Sequestration (CCS), CO$_2$ Leakage Detection, Convolutional Neural Network (CNN), Long Short Term Network (LSTM)

1 INTRODUCTION

The carbon capture and sequestration (CCS) technology collects the CO$_2$ from industrial sources such as thermal power plants and then injects compressed CO$_2$ into appropriate geologic formations underground. Storage operations must ensure that CO$_2$ is contained in the reservoir, even after injection stops (post closure) (Yang et al. 2011). Several monitoring technologies have been developed to detect CO$_2$ leakage at sequestration sites including observation of seismic data, groundwater chemistry monitoring, near-surface measurements of soil CO$_2$ fluxes, analysis of carbon isotopes in soil gas, measurement of tracer compounds injected with the sequestered CO$_2$, and nearby atmospheric monitoring of CO$_2$ and tracer gases (Korre et al. 2011; Leuning et al. 2008; Obbard 2008). Among all these techniques, monitoring leakage through seismic data has been used extensively for plume mapping, quantification of the injected volume in the reservoir and early detection of leakage (Fabriol et al. 2011). Many related detection methods have sprung up in this field, such as obtaining the elastic parameters at different injection times through Gassmann fluid substitution to estimate the CO$_2$ sequestration status (Macquet et al. 2017). Most methods for interpreting seismic data rely on expert analysis, resulting in high costs and perhaps limiting detection at low volumes.

With rapid improvements in computational power and fast data storage, machine learning techniques have been effectively applied to problems from various domains. Deep learning, a technique with its foundation in artificial neural networks, is emerging in recent years as a powerful tool (LeCun et al. 2015). Among various deep learning methods, convolutional neural networks (CNN) have achieved promising results in both detection and prediction tasks, such as speech recognition in 1D voice signal (Abdel-Hamid et al. 2014) and semantic segmentation in 2D image data (Long et al. 2015). Long short-term memory (LSTM) networks, which are a type of the recurrent neural net-
work (RNN), and have achieved great success in the tasks associated with sequential data, such as natural language text compression (Sundermeyer et al. 2012) and time-series processing (Che et al. 2018). Compared with traditional RNNs and other sequence learning methods such as hidden Markov models (Rabiner 1989), LSTM has shown its advantages in numerous applications through dealing with the exploding and vanishing gradient problem and keeping short-term “memory” for a long period of time (Sak et al. 2014).

In this paper, we developed a novel end-to-end data-driven detection method, which directly learns the mapping relation from seismic data to CO$_2$ leakage mass as shown in Figure 1. We designed our detection model based on conventional CNN and LSTM architectures. Seismic data comes with typical spatial and temporal characteristics. Seismic traces from a single receiver is a typical 1D time series, while the 2D seismogram collected from multiple receivers can be treated as imagery, and there is spatial relevance between different traces. Therefore, instead of simply adapting the existing CNN architectures, we designed a novel network architecture by combining 1D and 2D CNN to account for both spatial- and temporal characteristics of seismic data. Training a single CNN can be computationally expensive, and a combination of a couple of CNNs can be computationally prohibitive. To overcome the expensive computational cost, we built our network based on densely connected neural networks (Huang et al. 2017; Wu et al. 2018) to reduce the network parameters. In Fig. 1 we call our specially designed neural networks “spatial-temporal DenseNet” (or “ST-DenseNet”) in recognition that it is an extension of conventional DenseNet (Huang et al. 2017).

CNN-based network structures are powerful in generating high-level features from the input signals at a single time stamp. The leakage of CO$_2$ is a continuous time-dependent process. Our monitoring technique takes into consideration sequential information, which is proved to be extremely useful for an accurate monitoring and estimation of the CO$_2$ leakage mass. There have been a few traditional sequence models such as Markov models (Rabiner 1989), conditional random fields (Lafferty et al. 2001), and Kalman filters (Brown et al. 1992). However, all these models have their limitations in learning long-range dependencies, and some of them require domain knowledge or feature engineering, offering less opportunity for serendipitous discovery (Dietterich 2002). In contrast, neural networks are capable of learning high-level abstract representations automatically, and they can discover unforeseen features. In particular, LSTM (Hochreiter & Schmidhuber 1997) models have achieved state-of-the-art results for many sequential problems of varying-length sequential data. We incorporate the LSTM structures to our models to account for the historical data. These structures can capture long-range dependencies and nonlinear dynamics, therefore further improve the accuracy of our monitoring method.

We implemented a sequence of computational experiments to evaluate the performance of our
model including efficiency, accuracy and robustness in various scenarios. We firstly showed that ST-DenseNet generates more effective features in comparison with other learning techniques. We further demonstrated that with the help of the LSTM structure, ST-DenseNet can overcome the discontinuity issue in sequential CO\textsubscript{2} leakage prediction results. We also validated the robustness of our monitoring methods by utilizing an intra-site cross-location test (training and testing our model on the data from different locations) and noisy-data test. All the numerical tests of our monitoring method were based on synthetic seismic datasets generated using a model for a potential CO\textsubscript{2} storage site at Kimberlina, California ([Birkholzer et al. 2011], [Buscheck et al. 2017]). The Kimberlina site is in a partially compartmentalized sandstone basin. A model has been developed for the site ([Birkholzer et al. 2011]) and this model has been used to simulate various leakage scenarios ([Buscheck et al. 2017]).

In the following sections, we first briefly describe some basic architectures of the deep neural networks that are used in our methods (Sec. 2). We develop and discuss our novel CO\textsubscript{2} leakage monitoring method based on LSTM (Sec. 3). We then apply our method to test problems using synthetic reflection seismic data, and further discuss the results (Sec. 4). Finally, concluding remarks are presented in Sec. 5.

2 METHODOLOGY

2.1 Convolutional Neural Network

Convolutional neural network (CNN) is one of the most influential neural network structures in deep learning. LeNet, which is known as the first kind of CNN ([LeCun et al. 1995]). In 2012, AlexNet won the ImageNet competition ([Krizhevsky et al. 2012]). The authors introduced fully connected layers and max-pooling layers to help AlexNet outperform all the other methods. After that, a sequence of different structures such as VGGNet ([Simonyan & Zisserman 2014]), ResNet ([He et al. 2016]), GoogleNet ([Szegedy et al. 2017]), and DenseNet ([Huang et al. 2017]) were introduced. Regardless of the specific network architectures, all these different CNN networks consist of several common components including convolution layers, activation layers, batch normalization, and a loss function. Below, we provide some brief descriptions of these components.

2.1.1 Convolution Layer

A convolution layer consists of various of filters that in traditional algorithms are hand-engineered. Compared with fully connected layers, a convolutional layer uses relatively little pre-processing because it only needs to learn the parameters of convolutional filters. The calculation requirements of CNN are also less than multi-layer perceptron ([Geva & Sitte 1992]) because different positions of sig-
Figure 1. The schematic illustration of our detection method. In the data generation stage, the velocity models are generated from the fluid and rock simulations, and the simulated seismic data are further obtained from these velocity models. Next stage is model training. Majority of seismic data along with actual CO\textsubscript{2} leakage mass are fit into ST-DenseNet model to generate high-level features to represent the original information in seismic data. Then long short-term memory networks can merge these high-level features into time-sequential features, and the regressor will be used to detect the CO\textsubscript{2} leakage mass based on seismic data.

Signals can share the convolution filters to extract features. We devised convolutional layers to extract the feature map of signals. The convolution can be defined for signals with arbitrary dimensions. The discrete convolution operation for 2-d data is defined as

\[
X'_{i,j} = \sum_{m} \sum_{n} K_{m,n} \cdot X_{(s-1)\times i + m, (s-1)\times j + n},
\]

where \(X'_{i,j}\) denotes the value of the \((i, j)\) location of the output signals, \(X\) denotes the input signals, \(K \in \mathbb{R}^{m \times n}\) denotes the trainable kernel and \(s\) denotes the stride between each sliding location of the kernel. Figure 2 illustrates an example of how the convolution kernel is sliding on 2D data.

2.1.2 Activation Layer

In neural networks, the activation layer is applied to introduce non-linearity, which helps the neural network increase the ability of learning complex mapping relationships. Multi-layer neural network without non-linearity is the same as linear regression. The activation layer contains a non-linear function that suppresses some of the input values to zero or a value close to zero. Nonlinear activation functions allow neural networks to compute nontrivial problems using only a small number of neu-
Some seminal activation functions such as Sigmoid and Relu have been widely used in deep learning (Liu & Wang 2008). Since all three activation functions are used in ST-DenseNet, we provide some details to each of the three methods below.

(i) Rectified Linear Unit (ReLU) Function: ReLU has been considered as one of the most useful activation functions to increase the sparsity and to alleviate the problem of gradient vanishing (Nair & Hinton 2010). Its formula is defined as

\[ \text{ReLU}(x) = \max(0, x), \] (2)

where \( x \) is the input vector which needs to be activated. In this paper, we used ReLU as the activation function in each convolution block.

(ii) Sigmoid Function: Sigmoid function is a mathematical function having a characteristic S-shaped curve. The sigmoid function is differentiable and monotonic. The sigmoid function can be a perfect choice to predict the probability as an output, since its range is between 0 and 1. The sigmoid function is also widely used as activation function in different deep learning models. Sigmoid function refers to the special case of the logistic function, which is defined by the formula

\[ \text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}, \] (3)

where \( x \) is the input vector that needs to be activated. The sigmoid function is applied to restrict the value range of high-level features before they are fitted into a regressor.
2.1.3 Batch Normalization

For deep learning, the distribution of training data and testing data may not be strictly the same. In some cases, the prediction results of deep learning system will be severely inaccurate. Therefore, batch normalization (Ioffe & Szegedy 2015) is widely used to solve discrepant distribution problems, which break the independent and identically distributed (IID) assumption. Additionally, the training of deep neural network converges faster if the inputs of the network have zero-means, unit variances and decorrelated. Since each layer takes the output of the previous layer as the input, it is advantageous if the input of each layer also has these properties. Batch normalization is a technique for mini-batch gradient descent algorithms. It allows us to use much higher learning rates and to be less dependent with initialization. Batch normalization performs the following transformation for each activation $x_i$

$$B_{\gamma,\beta}(x_i) = \gamma \left( \frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}} \right) + \beta,$$

(4)

where $\gamma$ and $\beta$ are two trainable parameters; $i$ denotes the $i^{th}$ location of the feature map after a convolution layer; $\mu$ is calculated by averaging all values on the same feature map of $x_i$ given the mini-batch.

2.1.4 Convolution Block

The convolution block is usually a combination of several convolution layers (Conv), a batch normalization (BN), and an activation layer (ReLU). The convolution block can be summarized as a set of layers

$$x^{(l+1)} = \text{ReLU}(\text{BN}(\text{Conv}(x^{(l)}))),$$

(5)

where $x^{(l)}$ and $x^{(l+1)}$ denote the input and output of the $l^{th}$ convolution block.
2.1.5 Loss Function

In machine learning, a loss function (also known as cost function) is a function that measures how well the prediction results match training labels. To get the optimized prediction results, we usually seek to minimize the value of loss function. In this work, we utilized the mean squared error (MSE) as the loss function to measure the distance between groundtruth and predicted values.

\[ \text{Loss}_{MSE} = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2, \]  

(6)

where \( n \) is the sample size, \( \hat{Y} \in \mathbb{R}^1 \) is predicted leakage mass, and \( Y \in \mathbb{R}^1 \) is the groundtruth.

2.2 Residual Neural Network

As for traditional deep neural networks, an important issue is the gradient vanishing problem, which means the gradient will be vanishingly small during the back-propagation procedure (Goodfellow et al. 2016). In particular, the gradient vanishing issue can effectively prevent the parameters from updating their values, which is due to the fact that the gradient decreases exponentially with the number of neural network layers (Goodfellow et al. 2016). To solve this issue, He et al. (2016) proposed residual neural network (ResNet) framework which includes a shortcut in the convolutional neural network to avoid gradient vanishing.

Consider an input vector \( x_0 \) that is passed through a convolutional network. The network comprises \( L \) layers, each of which implements a non-linear transformation \( H_l(\cdot) \), where \( l \) is the index of the layer. \( H_l(\cdot) \) can be a composite function of operations such as batch normalization (BN), rectified linear units (ReLU), pooling, or convolution. We denote the output of the \( l^{th} \) layer as \( x_l \). Traditional convolutional feed-forward networks (such as VGG Net as shown in Fig. 4(a)) connect the output of the \( l^{th} \) layer as input to the \((l+1)^{th}\) layer, which gives rise to the following layer transition

\[ x_l = H_l(x_{l-1}). \]  

(7)

ResNet, As shown in Fig. 4(b), adds a skip-connection that bypasses the non-linear transformations with an identity function (He et al. 2016)

\[ x_l = H_l(x_{l-1}) + x_{l-1}. \]  

(8)
2.3 Densely Connected Network

The intuition behind densely connected networks (DenseNet) is similar to that of ResNet (Huang et al. 2017). Both of them aim at reusing the convolution features from previous layers and reducing the number of trainable parameters. Different from ResNet, DenseNet concatenates the features from different layers instead of directly adding them together. DenseNet has many advantages. It alleviates the vanishing-gradient problem, reinforces the feature propagation, and substantially reduces the number of parameters (Huang et al. 2017). As shown in Figure 4(c), a densely connected block is formulated as

\[ x_{l+1} = H([x_0, x_1, \ldots, x_l]), \]
\[ H(x) = W \ast (\sigma(B(x))), \]

where \( W \) is the weight matrix, the operator of “\( \ast \)” denotes convolution, \( B \) denotes batch normalization, \( \sigma(x) = \max(0, x) \) and \([x_0, x_1, \ldots, x_l]\) denotes the concatenation of all outputs of previous layers.

3 SPATIAL-TEMPORAL DENSENET

The seismic trace from a single receiver is a typical 1D time series, while the 2D seismogram collected from multiple receivers can be treated as imagery, and there is spatial relevance between different traces. In order to account for both the spatial and temporal characteristics, we developed a new network structure, called “spatial-temporal DenseNet (ST-DenseNet)”, by incorporating different CNN architectures. As shown in Fig. 4, ST-DenseNet model generates effective features by taking into consideration the spatial-temporal characteristics of seismic data. With the feature generated, we
Figure 5. A demonstration of unrolling the recurrent neural networks (RNNs), where $A$ denotes the RNN node, $X_i$ represents the time-sequential input, and $h_i$ stands for the corresponding hidden status to $X_i$ (Olah 2015).

can further improve our model by incorporating a long short-term memory (LSTM) network. We will elaborate the architecture of ST-DenseNet and how to incorporate LSTM.

3.1 ST-DenseNet: Feature Generation Using 1D/2D CNNs

Our ST-DenseNet is built upon DenseNet. We provide its specific network architecture in Table 1. The major difference between conventional DenseNet and ST-DenseNet is that our model starts with applying convolution layers to 1D time series, followed by employing convolutions on the 2D seismogram. There are a couple of benefits of using ST-DenseNet. Firstly, applying 1D convolution layers in time domain cannot only reduce the size of model parameters but also facilitates learning of important temporal features. As shown in Table 1, ST-DenseNet results in a reducationality of $10^4$ for the example simulated seismic data we used in this study. Secondly, applying 2D convolution layers fuses high-level spatial and temporal features together. All these benefits turn out to be critical in improving detection accuracy and reducing training costs.

3.2 Incorporation of Long Short-Term Memory Network

3.2.1 Classic Recurrent Neural Network

The basic RNN structure mentioned in the previous section is inspired by an intuitive idea of utilizing previous information for the learning task at a present time point. It has made a great step forward in solving sequence-relied problems but not entirely. Taking the recent sequential information into consideration might be sufficient for certain tasks, such as learning from the data generated from Markov process. However, when extending to more complicated real-world time-series problems, the information from early stage of time sequences (which also known as long-term information) can provide important supplementary information. Unfortunately, in the basic RNN structure, the information from early stage vanishes quickly as the length of input sequence increases, and the model turns out to be more difficult to capture the pattern from long-term information. To solve this problem, two popular
Table 1. The feature generation stage of ST-DenseNet. This model is designed for inputs with $6,000 \times 100 \times 6$, which are the simulated seismic data. "Conv($7 \times 1), 32, /(4 \times 1)$" denotes using $32 (7 \times 1)$ convolution kernels with a stride of $(4 \times 1)$.
Figure 6. An illustration of unrolling the long short-term memory networks, where $\sigma$ denotes the sigmoid activation function, $+$ represents the point-wise plus operation (Olah 2015).

The equivalent function $f_A$ of recurrent node $A$ can be defined as

\[
    h_t = \begin{cases} 
        f_A(x_t) = \tanh(W_{x_t} \cdot x_t + b_t), & \text{for } t = 1, \\
        f_A(h_{t-1}, x_t) = \tanh(W_{h_t} \cdot h_{t-1} + W_{x_t} \cdot x_t + b_t), & \text{for } t > 1,
    \end{cases}
\]  

where $W_{x_t}$ and $W_{h_t}$ are trainable weight matrices. $b_t$ is the corresponding bias term.

3.2.2 Long Short Term Memory Network

Long short-term memory network is a special case of RNN models. It has a chain structure similar to the basic RNNs, but the repeating modules are organized in a completely different structure. Distinguished from the other traditional RNN models, the basic unit of the LSTM hidden layer is a memory block (Zen & Sak 2015). The memory block (the outermost rectangle shown in the Figure 6) consists of memory cells that can memorize the temporary state, and a group of self-adaptive gating units to control information flow inside the memory block. Two gates, known as the input gate and the output gate, are added respectively to control the input and output in the block. The most important part of a memory cell is a recurrently connected linear unit, i.e., constant error carousel (CEC) (Gers F et al. 1999), whose activated value can represent the state of cells. Because of the existence of this structure, multiplicative gates can learn to open and close, and thus the LSTM networks can solve the problem of vanishing error and lacking long-term dependence by remaining the network error in a constant range. To prevent the internal cell values growing drastically when processing continuous sequential series without being previously segmented, a forget gate is introduced into the memory block. Similar to a human tendency to forget useless information, the forget gate allows the memory block to update by itself once the memorized information is out of date, and replaces the cell state by multiplying the activated value from forget gate. The entire computation can be summarized by a series of equations.
as follows

\[
f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f),
\]

\[
i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i),
\]

\[
\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C),
\]

\[
C_t = f_t \ast C_{t-1} + i_t \ast \tilde{C}_t,
\]

\[
o_t = \sigma(W_0 \cdot [h_{t-1}, x_t] + b_0),
\]

\[
h_t = o_t \ast \tanh(C_t),
\]

where \(\sigma\) represents the sigmoid activation function, and \(\tanh\) represents the hyperbolic tangent function. \(x_t\) and \(h_t\) represent the input and output vector at time step \(t\). \(W_f, W_i\) and \(W_0\) are trainable weight matrices. \(b_f, b_i\) and \(b_0\) are corresponding bias terms. \(C_t\) and \(\tilde{C}_t\) are memory vectors, which can be used to remember long-term information.

4 NUMERICAL EXPERIMENTS

In this section, we provide three different numerical tests using synthetic seismic data generated based on flow simulations for a model CO\(_2\) storage site (Buscheck et al. 2017; Yang et al. 2018). We designed three tests to validate the performance of our CO\(_2\) leakage detection method for different monitoring scenarios. In particular, we designed Test 1 to mimic the monitoring situation when a single seismic dataset is available. We designed Test 2 to mimic the monitoring situation when a sequence of seismic datasets is available over time. We further validated the performance of our detection method when data are noisy or training and testing data are not acquired from the same site.

4.1 Dataset

To evaluate the effectiveness of our developed approach, we test and validate our model for CO\(_2\) leakage mass detection task on simulated reflection seismic datasets. The seismic velocity models were generated based on flow simulations for hypothetical brine and CO\(_2\) leakage along damaged wellbores at a model CO\(_2\) storage site near Kimberlina in the southern San Joaquin Basin, California. The legacy wells in the flow simulations are located at 1, 3, and 6 km away from the CO\(_2\) injector. Most CO\(_2\) has reached the very top layer of the model domain and CO\(_2\) is in the gas phase in these wellbore leakage simulations. The set of flow simulations is specially developed for evaluating the effectiveness of CO\(_2\) leakage monitoring methods. At each time step in the flow simulation outputs, we constructed seismic velocity models using the Gassmanns equations (Gassmann et al. 1951). The porosity of the
Figure 7. Aquifer impact data is generated at three distances (1, 3, and 6 km) from the CO2 injector using the hydrostratigraphy (Buscheck et al. 2017).

rock is assumed to be 0.35 in the Kimberlina wellbore leakage simulations. The physical properties of the effective pore fluid (mixture of CO2 gas and brine) depend on the temperature, pore pressure, salinity of brine and CO2/brine saturation conditions. The temperature is kept constant at 40°C in the wellbore leakage simulations. The pore pressure, salinity of brine, CO2 saturation and density of the CO2 gas phase are obtained from the leakage simulation outputs. We provide in Table 2 below some representative values for the fluid and rock properties to generate simulations in our numerical tests. The workflow of constructing seismic velocity models based on flow simulations has been described in the work by Wang et al. (2018).

To generate the seismic data, we assumed a total of 3 sources and 100 receivers evenly distributed along the top boundary of the model. The source interval was 500 m, and the receiver interval was 15 m. We used a Ricker wavelet with a center frequency of 50 Hz as the source time function and a staggered-grid finite-difference scheme with a perfectly matched layered absorbing boundary condition to generate synthetic seismic reflection data (Tan & Huang 2014; Zhang & Shen 2010). The syn-
Spatial-Temporal Densely Connected CNN-LSTM Nets: An Application to CO₂ Leakage Detection

| Parameter Name                      | Parameter Value |
|-------------------------------------|-----------------|
| Salinity of Brine                   | 0.001           |
| Density of Brine                    | 1 g/cm³         |
| Density of CO₂ Gas                  | 0.075 g/cm³     |
| CO₂ Fraction in the Pores           | (0. 0.56)       |
| Porosity of the Rock                | 0.35            |
| Bulk Density of the Rock            | 1.88 g/cm³      |
| Bulk Modulus of the Rock            | 1.18 GPa        |
| Bulk Modulus of Mineral Grains      | 31.1 GPa        |
| Bulk Modulus of Dry Rock Frame      | 1.15 GPa        |

Table 2. The representative values for the fluid and rock properties used in our numerical tests to generate simulations.

Artificial trace at each receiver consisted of a collection of 2-component time-series data, each of length 6, 000. With the seismic data available, the dimension of input data to our networks was 6, 000 × 100 × 6 (as illustrated in Fig. 1), where each of the 3 dimensions corresponds to time steps, receiver number, and component/source, respectively.

The CO₂ leakage mass was varied from 0 to 10¹⁰ kilogram (kg). We scaled all the mass data by a Log function as

\[ \tilde{Y} = \log_{10}(Y + 1), \]

where \( Y \) represents the original value of CO₂ leakage mass, and \( \tilde{Y} \) stands for scaled leakage mass. A value of 1 was added to each \( Y \) to avoid taking a \( \log(0) \). All computations were carried on a computer with an Intel Xeon E5-2650 core running at 2.3 GHz, and Tesla K40c GPU with 875 MHz boost clock.

4.2 Tests on Random Leakage Monitoring

We tested the performance of ST-DenseNet provided with single seismic dataset. The purpose of this test was to mimic the monitoring situation when single seismic dataset is available. Specifically, we

|                  | ±10% Acc | ±5% Acc | ±3% Acc | Parameters |
|------------------|----------|---------|---------|------------|
| Kernel SVR       | 0.263    | fail    | fail    | 36K        |
| VGG-based CNN    | 0.719    | 0.673   | 0.651   | 29M        |
| ResNet-based CNN | 0.938    | 0.885   | 0.852   | 17M        |
| ST-DenseNet      | 0.936    | **0.912** | **0.891** | 9M         |

Table 3. The CO₂ leakage mass prediction results given by kernel support vector regression (kernel SVR), VGG-based CNN, ResNet-based CNN and ST-DenseNet. “fail” indicates the accuracy is extremely low. The results indicate that ST-DenseNet outperforms both the classical regression model and other CNN-based models.
randomly select 2,400 groups for training, 300 groups for validating, and 227 groups for testing. We compared our method to different regression models, including: 1. support vector regression with radial basis function kernel (Kernel SVR); 2. VGG-based CNN; and 3. ResNet-based CNN. We used accuracy with different error rates to evaluate the regression results. In particular, “±r% Accuracy” represents the tolerance of error rate, and this metric can be mathematically defined as

\[ 1 - r\% \leq \frac{\text{Predicted Value}}{\text{Groundtruth Value}} \leq 1 + r\%. \]  

(19)

In other words, if the predicted leakage mass is in the range from (100-r)% to (100+r)% of the actual leakage mass, the prediction result is considered as accurate. Otherwise, the prediction fails. In our test, we selected three different values of \( r = 3, 5 \) and 10 as evaluation metrics. The number of trainable parameters was also reported to reflect the computational complexity of different models.

The results from several different methods are summarized in Table 3. The kernel SVM had an extremely low accuracy, so it is listed as “fail” in the table. This result suggests that advanced feature extraction techniques are required in advance to apply SVR or other classical regression models. By comparing to the VGG-based CNN or ResNet-based CNN, ST-DenseNet yields higher accuracy. The only exception was the testing scenario of “±10% Accuracy”, where our method still produces comparable results to those obtained by using ResNet-based CNN. The number of trainable parameters can be used as an indication of the computation complexity. In Table 3, we observe that ST-DenseNet yielded the smallest number of model parameters among all the CNN-based methods. So, using single seismic dataset, ST-DenseNet cannot only produce the most accurate CO\(_2\) leakage mass detection but also requires the least amount of trainable parameters.

### 4.3 Test on Sequential Leakage Monitoring

We also tested the monitoring situation when a sequence of seismic datasets are acquired over time. We selected three leakage scenarios from three locations shown in Fig. 7 and report their detection results from Figs. 8 to 13. To account for the dependency from sequential time series, we incorporated LSTM into ST-DenseNet as illustrated in Fig. 1. We chose VGGNet and ResNet as baseline methods. In order to compare fairly, we also incorporated LSTM with both VGGNet and ResNet, which result in 5 different CNN-based baseline models including VGG-Based CNN, VGG-Based CNN with LSTM, ResNet-Based CNN, ResNet-Based CNN with LSTM, and ST-DenseNet without LSTM. To illustrate the time dependency, we plot the detection results versus time obtained using VGG-based CNN (Fig. 8(a), Fig. 10(a), and Fig. 12(a)), ResNet-based CNN (Fig. 8(b), Fig. 10(b), and Fig. 12(b)), and ST-DenseNet (Fig. 8(c), Fig. 10(c), and Fig. 12(c)). We also show the results of the detection
Figure 8. Illustration of three methods to detect CO\textsubscript{2} leakage mass at location 1 km away from injector. We provide detection results using three network architectures including VGGNet (a), ResNet (b), and ST-DenseNet (c). Within each sub-figure, we also provide detection results using network with LSTM (in “−⋄−”) and without LSTM (in “−△−”). The ground-truth is plotted in blue. We observe that (1) the incorporation of LSTM improves the detection accuracy for all three CNN architectures including ST-DenseNet; (2) ST-DenseNet yields the most accurate detection results of all three methods.

results of the leakage mass at 22 different times using all 6 methods including ST-DenseNet. In each of these figures, the ground-truth is plotted in blue. The results using VGG-Based CNNs are plotted in dark green (Fig. 8(a), Fig. 10(a), and Fig. 12(a)). The results using ResNet-Based CNNs are plotted in purple (Fig. 8(b), Fig. 10(b), and Fig. 12(b)). The results using ST-DenseNet are plotted in red (Fig. 8(c), Fig. 10(c), and Fig. 12(c)).

In Fig. 8 we observe that the incorporation of LSTM improves the detection accuracy for all three CNN architectures including ST-DenseNet. In particular, the CNN networks without LSTM not only yielded detections with significant oscillations over time, but also produced inaccurate leakage mass.
Figure 9. Illustration of four methods to detect CO$_2$ leakage mass at location 1 km away from injector. Figure 9(a) shows the detection results in “Log of GroundTruth VS. Log of Detected Mass” of the 200-year leakage process, and Fig. 9(b) is the enlarged view of the last 19 detection results in 200 years. We provide detection results using four network architectures including VGGNet (green), ResNet (purple), ST-DenseNet (red), and ST-DenseNet with LSTM (orange). The ground-truth is in blue. We observe that ST-DenseNet+LSTM model yields the most accurate detection results and the minimum variance among all four methods.

On the other hand, by incorporating LSTM, the detection results using all three CNN networks were significantly improved. The oscillation issue have been alleviated and the detection accuracy was also improved. This demonstrates that accounting the dependency from sequential time series is critical in generating an accurate leakage detection model. Furthermore, by comparing all three versions of CNN networks with LSTM, we notice that ST-DenseNet still yields the most accurate results as can be observed by comparing Fig. 8(a), 8(b), and 8(c). Similar performance can be observed from Figs. 10 and 12.

To illustrate the detection accuracy, we also provide the plots of Log of GroundTruth VS Log of Detected Mass using VGG-Based CNN (green), ResNet-Based CNN (purple), ST-DenseNet (red), and ST-DenseNet + LSTM (orange) for three different leakage locations at 1 km (Fig. 9), 3 km (Fig. 11) and 6 km (Fig. 13), respectively. We observe that the detection results of ST-DenseNet with LSTM are more accurate than those obtained using all the other detection methods with the smallest variances from GroundTruth. Therefore, we conclude from this test that ST-DenseNet with LSTM yields more accurate detection results than those obtained using VGGNet with/without LSTM and ResNet with/without LSTM when a time sequence of data is available.
Spatial-Temporal Densely Connected CNN-LSTM Nets: An Application to CO₂ Leakage Detection

Figure 10. Illustration of three methods to detect CO₂ leakage mass at location 3 km away from injector. We provide detection results using three network architectures including VGGNet (a), ResNet (b), and ST-DenseNet (c). Within each sub-figure, we also provide detection results using network with LSTM (in “−⋄−”) and without LSTM (in “−△−”). The ground-truth is plotted in blue. We observe that (1) the incorporation of LSTM improves the detection accuracy for all three CNN architectures including ST-DenseNet; (2) ST-DenseNet yields the most accurate detection results of all three methods.

4.4 Tests on Robustness

Robustness is an important issue for any deep neural network (Fawzi et al. 2017). In this section, we investigate the robustness of ST-DenseNet by implementing (1) an intra-site cross-location test to demonstrate the generalization ability of our model on unknown test data, and (2) a noisy data test to validate the performance of our method under noisy environment.
Figure 11. Illustration of four methods to detect CO₂ leakage mass at location 3 km away from injector. Figure 11(a) shows the detection results in “Log of GroundTruth VS. Log of Detected Mass” of the 200-year leakage process, and Fig. 11(b) is the enlarged view of the last 19 detection results in 200 years. We provide detection results using four network architectures including VGGNet (green), ResNet (purple), ST-DenseNet (red), and ST-DenseNet with LSTM (orange). The ground-truth is in blue. We observe that ST-DenseNet+LSTM model yields the most accurate detection results and the minimum variance among all four methods.

4.4.1 Intra-Site Cross-Location Test

We tested ST-DenseNet on intra-site cross-location datasets to demonstrate its robustness and weak generalization. The samples in our dataset were collected from three different locations within a site. In particular, the distances of the three locations to CO₂ injector were 1 km, 3 km, and 6 km, respectively (shown in Figure 7). We provide the key parameters used for generating simulations at the three well locations in Table 4. To further visualize the leakage parameters at those three locations in different time, we provide the pressure change in Table A1 (Appendix A) and CO₂ saturation in Table A2 (Appendix B).

We used samples from 2 locations for training and tested the detection methods on the third, re-

| Parameters | Well Location at 1km, 3km, 6km |
|------------|--------------------------------|
| Layer Depths | Layer 1: Atmosphere (1e-30 m thick)  
|             | Layer 2: Upper Caprock (10 m thick)  |
|             | Layer 3: Etchegoin Formation (536.23 m thick)  
|             | Layer 4: Macoma-Chanac Formation (679.04 m thick)  
|             | Layer 5: Santa Margarita-McLure Formation (185.94 m thick)  |
| Temperature | 40°C  |
| Porosity | 0.35 |

Table 4. Comparison of key parameters at the three well locations.
We provide detection results using three network architectures including VGGNet (a), ResNet (b), and ST-DenseNet (c). Within each sub-figure, we also provide detection results using network with LSTM (in “−⋄−”) and without LSTM (in “−△−”). The ground-truth is plotted in blue. We observe that (1) the incorporation of LSTM improves the detection accuracy for all three CNN architectures including ST-DenseNet; (2) ST-DenseNet yields the most accurate detection results of all three methods.

Figure 12. Illustration of three methods to detect CO$_2$ leakage mass at location 6 km away from injector. We provide detection results using three network architectures including VGGNet (a), ResNet (b), and ST-DenseNet (c). Within each sub-figure, we also provide detection results using network with LSTM (in “−⋄−”) and without LSTM (in “−△−”). The ground-truth is plotted in blue. We observe that (1) the incorporation of LSTM improves the detection accuracy for all three CNN architectures including ST-DenseNet; (2) ST-DenseNet yields the most accurate detection results of all three methods.

resulting in three different combinations of training-testing groups. We then used the average prediction accuracies obtained using ST-DenseNet based on the three groups as the overall accuracy. Similar to previous tests, all the accuracies were calculated under the criterion of ”±5% Accuracy”. This intra-site cross-location test can be much more challenging than all of the previous tests for two reasons. From the physics point of view, in all previous tests, the datasets used for training and testing came from physics simulations of the conditions. Therefore, ST-DenseNet captured the physical correspondence between data and target values (leakage mass in our problem). On the other hand, in this robustness test, training sets and testing sets were obtained from physics simulations based on different conditions. Therefore, the relationship that was learned from training may or may not have fully
Figure 13. Illustration of four methods to detect CO\textsubscript{2} leakage mass at location 6 km away from injector. Figure 13(a) shows the detection results in “Log of GroundTruth VS. Log of Detected Mass” of the 200-year leakage process, and Figure 13(b) is the enlarged view of the last 19 detection results in 200 years. We provide detection results using four network architectures including VGGNet (green), ResNet (purple), ST-DenseNet (red), and ST-DenseNet with LSTM (orange). The ground-truth is in blue. We observe that ST-DenseNet+LSTM model yields the most accurate detection results and the minimum variance among all four methods.

represented the one in the test sets. From the mathematical point of view, the goal for any machine learning algorithm including deep learning is to find a model to fit data. With both training sets and testing sets drawn from the same distribution (previous tests), it becomes relatively less challenging to generate a model comparing to the situations when training sets and testing sets are drawn from two different distributions (current test).

Table 5 presents the results of the robustness test for the various methods. The use of different locations for training and testing did degrade the accuracy compared with tests using the same locations for training and testing (Table 3). This indicates that there is some discrepancy between the training sets and testing sets. However, ST-DenseNet still achieves reasonable detection results and yields the highest accuracy among all the CNN-based networks in this intra-site cross-location test. As illustrated in the last row of Table 5, by incorporating LSTM into ST-DenseNet, we can improve the detection accuracy further.

4.4.2 Noisy Data Test

The impact of random noise to deep networks is another important fact to consider for the robustness (Fawzi et al. 2017). In real seismic measurements, random noise can be usually categorized as “natural noise” and “cultural noise” (Li et al. 2017). The level of the noise may be varied significantly
Spatial-Temporal Densely Connected CNN-LSTM Nets: An Application to CO₂ Leakage Detection

| Accuracy of Different Test Location | 1km  | 3km  | 6km  | Overall Acc |
|-------------------------------------|-----|-----|-----|-------------|
| VGG-based CNN                       | 68.6% | 73.3% | 72.1% | 71.1%        |
| ResNet-based CNN                    | 73.6% | 77.4% | 79.9% | 76.7%        |
| ST-DenseNet                         | 78.4% | 81.9% | 81.5% | 80.0%        |
| ST-DenseNet + LSTM                  | 79.3% | 82.3% | 82.6% | 81.2%        |

Table 5. The robustness test on generalization using different models including VGG-based CNN, ResNet-based CNN, ST-DenseNet and ST-DenseNet+LSTM model. All the accuracy are calculated under the criterion of "±5% Accuracy". Our ST-DenseNet outperforms all the other CNN-based models. By incorporating LSTM structure and using the sequential information, the detection results are further improved.

due to differences of recording instruments, weather conditions, environment issues etc. In this test, we generated 30 db additive Gaussian noise, which can be a good approximation of the noise level that is seen near surface according to Young et al. (1996). The Gaussian noise was added to the clean simulated seismic data as shown in Fig [14]. To validate the performance of ST-DenseNet algorithms in different scenarios, we report in Table 6 to show the results on both random leakage monitoring and sequential leakage monitoring tests. We use mean squared error(MSE) as the metric to evaluate the performance of our detection methods under noisy environment. The MSE is defined as

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2,$$

where \(n\) is the total number of testing samples, \(Y_i\) is the real leakage mass value, \(\hat{Y}_i\) is the detected leakage mass value.

All CNN-based networks are degraded when noise is present in the data (comparing Column 3 to Column 2 in Table 6). However, ST-DenseNet yields the smaller MSE error values with noise (MSE ≈ 2.653 and MSE ≈ 1.327 with LSTM) comparing to VGGNet (MSE ≈ 9.485) and ResNet (MSE ≈ 5.784). Moreover, the last column in Table 6 shows that ST-DenseNet has the least degradation among all three CNN networks.

We provide the detection results for each method when applied to data with 30db of added noise

| MSE of Different Detection Methods | Without Noise | With Noise | Degradation (With Noise - Without Noise) |
|-----------------------------------|--------------|-----------|----------------------------------------|
| VGG-based CNN                     | 6.891        | 9.485     | 2.594                                  |
| ResNet-based CNN                  | 3.972        | 5.784     | 1.812                                  |
| ST-DenseNet                       | 1.136        | 2.653     | 1.494                                  |
| ST-DenseNet + LSTM                | 0.765        | 1.327     | 0.562                                  |

Table 6. A summary of the MSE values of VGG-based CNN, ResNet-based CNN, ST-DenseNet and ST-DenseNet with LSTM architecture for testing data with or without noise. Our ST-DenseNet not only yields the smaller MSE error values, but also degrades the least among all three CNN networks.
Figure 14. Illustration of the 1-D clean seismic data (blue) and the noisy signal with 30 db Gaussian noise (red).

from Fig. 15 to Fig. 18. Specifically, in Fig. 15 we compare results obtained using ST-DenseNet (red) to those obtained using VGGNet (green) and ResNet (purple) for four different leakage scenarios. We notice that detections results of all three CNN networks including ST-DenseNet become more oscillatory comparing to detection results in Figs. 8, 10, and 12. In Figs. 16, a significant improvement can be observed both in detection accuracy and robustness after ST-DenseNet cascading with LSTM. However, ST-DenseNet still yielded the most accurate results out of all three CNN networks. Similarly, we provide the plots of the results in the Log of GroundTruth VS. Log of Detected Mass of the 200-year leakage process in Fig. 17 and the enlarged view of the last 19 detection results in Fig. 18. Again, we observe that our ST-DenseNet with LSTM yields the most accurate results compared with all the other detection methods.

5 CONCLUSIONS

In this paper, we developed a novel method based on spatial-temporal densely connected convolutional networks (ST-DenseNet) to capture abstract high-level features from the seismic data for detection of CO$_2$ leakage. To further account for sequential information from CO$_2$ leakage history data, we incorporated long short-term memory networks to ST-DenseNet. Our method not only considers both the spatial and temporal characteristics of seismic data but also significantly reduces the number of parameters in the network’s structure. We tested our method using simulated reflection seismic data generated based on various leakage scenarios at the Kimberlina site in the southern San Joaquin Basin, California. By comparing with several commonly-used machine learning methods, we demonstrated that our detection method outperforms traditional regression methods and other popular deep learning
Figure 15. Illustration of four CO$_2$ leakage mass detection scenarios on the data with 30db noise. Each of these sub-figures contains 22 different leakage mass over time (except the point of $t = 0$). The ground-truth is plotted in blue. We show results obtained using VGG-based CNN (in green), ResNet-based CNN (in purple), and ST-DenseNet (in red). Our ST-DenseNet achieves most accurate CO$_2$ leakage mass detection results among the three CNN-based methods.

models. We also demonstrate the robustness of ST-DenseNet by implementing intra-site cross-location tests and noisy-data tests. Our novel detection method shows great potential in CO$_2$ leakage detection as well as for other monitoring tasks in various subsurface applications.
Figure 16. Illustration of four CO$_2$ leakage mass detection scenarios on the data with 30db noise. Each of these sub-figures contains 22 different leakage mass over time (except the point of $t = 0$). The groundtruth is plotted in blue. We show results obtained using ST-DenseNet (in “−△−”) and ST-DenseNet with LSTM (in “−⋄−”). With the help from LSTM, the detection accuracy using ST-DenseNet can be much more improved. This again demonstrates that the dependency within the time sequence can be critical to improve the detection accuracy.

6 ACKNOWLEDGMENTS

This work was co-funded by the U.S. DOE Office of Fossil Energys Carbon Storage program and the Center for Space and Earth Science (CSES) at Los Alamos National Laboratory (LANL). The computation was performed using super-computers of LANL’s Institutional Computing Program.
Figure 17. Illustration of the detection results plotted in the Log of GroundTruth VS. Log of Detected Mass of the 200-year leakage process on the data with 30db noise. We provide detection results using four network architectures including VGGNet (green), ResNet (purple), ST-DenseNet (red), and ST-DenseNet with LSTM (orange). The ground-truth is in blue. We observe that ST-DenseNet+LSTM model yields the most accurate detection results and the minimum variance among all four methods.

APPENDIX A: COMPARISON OF KEY PARAMETERS AT THE THREE WELL LOCATIONS - PRESSURE

APPENDIX B: COMPARISON OF KEY PARAMETERS AT THE THREE WELL LOCATIONS - CO₂ SATURATION
Figure 18. Illustration of the detection results plotted in the Log of GroundTruth VS. Log of Detected Mass on the data with 30db noise. We provide the enlarged view of the last 19 detection results shown in Fig. [17]. The network architectures include VGGNet (green), ResNet (purple), ST-DenseNet (red), and ST-DenseNet with LSTM (orange). The ground-truth is in blue. We observe that ST-DenseNet+LSTM model yields the most accurate detection results and the minimum variance among all four methods.
Spatial-Temporal Densely Connected CNN-LSTM Nets: An Application to CO₂ Leakage Detection

| Pressure Change | Well Location at 1 km | Well Location at 3 km | Well Location at 6 km |
|-----------------|----------------------|----------------------|----------------------|
| \(t = 5 \text{ yr}\) | ![Image](image1.png) | ![Image](image2.png) | ![Image](image3.png) |
| \(t = 20 \text{ yr}\) | ![Image](image4.png) | ![Image](image5.png) | ![Image](image6.png) |
| \(t = 100 \text{ yr}\) | ![Image](image7.png) | ![Image](image8.png) | ![Image](image9.png) |
| \(t = 200 \text{ yr}\) | ![Image](image10.png) | ![Image](image11.png) | ![Image](image12.png) |

\textbf{Table A1.} The illustration of pressure changes at different wells in different year.
| CO₂ Saturation | Well Location at 1 km | Well Location at 3 km | Well Location at 6 km |
|----------------|-----------------------|-----------------------|-----------------------|
| t = 5 yr       | ![Image](image1.png)  | ![Image](image2.png)  | ![Image](image3.png)  |
| t = 20 yr      | ![Image](image4.png)  | ![Image](image5.png)  | ![Image](image6.png)  |
| t = 100 yr     | ![Image](image7.png)  | ![Image](image8.png)  | ![Image](image9.png)  |
| t = 200 yr     | ![Image](image10.png) | ![Image](image11.png) | ![Image](image12.png) |

*Table A2.* The illustration of CO₂ Saturation at different wells in different year.
REFERENCES

Abdel-Hamid, O., Mohamed, A.-r., Jiang, H., Deng, L., Penn, G., & Yu, D., 2014. Convolutional neural networks for speech recognition, IEEE/ACM Transactions on audio, speech, and language processing, 22(10), 1533–1545.

Birkholzer, J., Zhou, Q., Cortis, A., & Finsterle, S., 2011. A sensitivity study on regional pressure buildup from large-scale co2 storage projects, Energy Procedia, 4, 4371–4378.

Brown, R. G., Hwang, P. Y., et al., 1992. Introduction to random signals and applied Kalman filtering, vol. 3, Wiley New York.

Buscheck, T. A., Mansoor, K., Yang, X., & Carroll, S. A., 2017. Simulated data for testing monitoring techniques to detect leakage in groundwater resources: Kimberlina model with wellbore leakage, Tech. rep., Lawrence Livermore National Laboratory.

Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y., 2018. Recurrent neural networks for multivariate time series with missing values, Scientific reports, 8(1), 6085.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

Dietterich, T. G., 2002. Machine learning for sequential data: A review, in Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), pp. 15–30, Springer.

Fabriol, H., Bitri, A., Bourgeois, B., Delatre, M., Girard, J., Pajot, G., & Rohmer, J., 2011. Geophysical methods for CO2 plume imaging: comparison of performances, Energy Procedia, 4, 3604–3611.

Fawzi, A., Dezfooli, M., Mohsen, S., & Frossard, P., 2017. The robustness of deep networks-a geometric perspective, IEEE Signal Processing Magazine, 34(ARTICLE).

Gassmann, F., Maggiorini, M., Stüdler, E., & Winkler, W., 1951. Verteljahrsschrift der naturforschenden gesellschaft in zurich, Uber die elastizitat poroser medien, 96, 1–23.

Gers F. A., Schmidhuber, J., & Cummins, F., 1999. Learning to forget: Continual prediction with lstm, IET Conference Proceedings.

Geva, S. & Sitte, J., 1992. A constructive method for multivariate function approximation by multilayer perceptrons, IEEE transactions on neural networks, 3(4), 621–624.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y., 2016. Deep learning, vol. 1, MIT press Cambridge.

He, K., Zhang, X., Ren, S., & Sun, J., 2016. Deep residual learning for image recognition, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hochreiter, S. & Schmidhuber, J., 1997. Long short-term memory, Neural computation, 9(8), 1735–1780.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q., 2017. Densely connected convolutional networks., in CVPR, vol. 1, p. 3.

Ioffe, S. & Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift, arXiv preprint arXiv:1502.03167.

Korre, A., Imrie, C. E., May, F., Beaubien, S. E., Vandermeijer, V., Persoglia, S., Golmen, L., Fabriol, H., &
Dixon, T., 2011. Quantification techniques for potential co2 leakage from geological storage sites, *Energy Procedia*, 4, 3413–3420.

Krizhevsky, A., Sutskever, I., & Hinton, G. E., 2012. Imagenet classification with deep convolutional neural networks, in *Advances in neural information processing systems*, pp. 1097–1105.

Lafferty, J., McCallum, A., & Pereira, F. C., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data, *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289.

LeCun, Y., Jackel, L., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P., et al., 1995. Learning algorithms for classification: A comparison on handwritten digit recognition, *Neural networks: the statistical mechanics perspective*, 261, 276.

LeCun, Y., Bengio, Y., & Hinton, G., 2015. Deep learning, *nature*, 521(7553), 436.

Leuning, R., Etheridge, D., Luhar, A., & Dunse, B., 2008. Atmospheric monitoring and verification technologies for co2 geosequestration, *International Journal of Greenhouse Gas Control*, 2(3), 401–414.

Li, G., Li, Y., & Yang, B., 2017. Seismic exploration random noise on land: Modeling and application to noise suppression, *IEEE Transactions on Geoscience and Remote Sensing*, 55, 4668–4681.

Liu, Q. & Wang, J., 2008. A one-layer recurrent neural network with a discontinuous hard-limiting activation function for quadratic programming, *IEEE transactions on neural networks*, 19(4), 558–570.

Long, J., Shelhamer, E., & Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Macquet, M., Lawton, D. C., Donags, J., & Barraza, J., 2017. Feasibility study of time-lapse-seismic monitoring of co2 sequestration, in *EAGE/SEG Research Workshop 2017*.

Nair, V. & Hinton, G. E., 2010. Rectified linear units improve restricted boltzmann machines, in *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814.

Obbard, J., 2008. Carbon capture and sequestration: Integrating technology, monitoring and regulation, *Journal of Environmental Quality*, 37(1), 289.

Olah, C., 2015. Understanding lstm networks, [http://colah.github.io/posts/2015-08-Understanding-LSTMs/](http://colah.github.io/posts/2015-08-Understanding-LSTMs/).

Rabiner, L. R., 1989. A tutorial on hidden markov models and selected applications in speech recognition, *Proceedings of the IEEE*, 77(2), 257–286.

Sak, H., Senior, A., & Beaufays, F., 2014. Long short-term memory recurrent neural network architectures for large scale acoustic modeling, in *Fifteenth annual conference of the international speech communication association*.

Simonyan, K. & Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556*.

Sundermeyer, M., Schlüter, R., & Ney, H., 2012. Lstm neural networks for language modeling, in *Thirteenth annual conference of the international speech communication association*.

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning., in *AAAI*, vol. 4, p. 12.
Tan, S. & Huang, L., 2014. An efficient finite-difference method with high-order accuracy in both time and space domains for modelling scalar-wave propagation, *Geophysical Journal International*, **197**(2), 1250–1267.

Wang, Z., Harbert, W. P., Dilmore, R. M., & Huang, L., 2018. Modeling of time-lapse seismic monitoring using co2 leakage simulations for a model co2 storage site with realistic geology: Application in assessment of early leak-detection capabilities, *International Journal of Greenhouse Gas Control*, **76**, 39–52.

Wu, Y., Lin, Y., Zhou, Z., Bolton, D. C., Liu, J., & Paul, J., 2018. Deepdetect: A cascaded region-based densely connected network for seismic event detection, *IEEE Transactions on Geoscience and Remote Sensing*, (in Press).

Yang, X., Buscheck, T. A., Mansoor, K., Wang, Z., & Carroll, S. A., 2018. Effectiveness of geophysical monitoring methods in detecting brine and co2 leakage in underground sources of drinking water, Tech. rep., Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).

Yang, Y.-M., Small, M. J., Ogretim, E. O., Gray, D. D., Bromhal, G. S., Strazisar, B. R., & Wells, A. W., 2011. Probabilistic design of a near-surface co2 leak detection system, *Environmental science & technology*, **45**(15), 6380–6387.

Young, C. J., Chael, E. P., Withers, M. M., & Aster, R. C., 1996. A comparison of the high-frequency (≤1 hz) surface and subsurface noise environment at three sites in the united states, *Bulletin of the Seismological Society of America*, **86**(5), 1516–1528.

Zen, H. & Sak, H., 2015. Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis, in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 4470–4474, IEEE.

Zhang, W. & Shen, Y., 2010. Unsplit complex frequency-shifted pml implementation using auxiliary differential equations for seismic wave modeling, *Geophysics*, **75**(4), T141–T154.