Inversion of Time-Lapse Surface Gravity Data for Detection of 3-D CO₂ Plumes via Deep Learning

Adrian Celaya®, Bertrand Denel, Yen Sun, Mauricio Araya-Polo, and Antony Price®

Abstract—We introduce two algorithms that invert simulated gravity data to 3-D subsurface rock/flow properties. The first algorithm is data-driven, deep learning (DL)-based approach, and the second is also data-driven but considers the temporal evolution of surface gravity events. The target application of these proposed algorithms is the prediction of subsurface CO₂ plumes as a complementary tool for monitoring CO₂ sequestration deployments. Each proposed algorithm outperforms traditional inversion methods and produces high-resolution, 3-D subsurface reconstructions in near real-time. In addition, our proposed methods achieve Dice scores of up to 0.8 for predicted plume geometry and near-perfect data misfit in terms of µGals. These results indicate that combining 4-D surface gravity monitoring (low-cost acquisition) with DL techniques represents an effective and nonintrusive method for monitoring CO₂ storage sites.

Index Terms—Carbon capture and storage, deep learning (DL), gravity, inversion.

I. INTRODUCTION

GREENHOUSE Gas (GHG) reduction commitments aim to address the increase of atmospheric CO₂ concentrations implemented on a large scale. They include technologies such as fossil fuel consumption reduction (improvement of energy efficiency, application of new energy sources), expansion of absorption sources through afforestation/reforestation, and carbon dioxide capture and storage (CCS).

One of the CCS technologies currently being deployed worldwide is CO₂ geological storage. The technology involves separating and capturing CO₂ emitted from large-scale fixed sources, such as cement, petrochemical, or coal-fired power plants, and storing it in saline formations located deep underground (hereinafter referred to as “aquifers”). It holds promise as the most practical, immediate-effect, and near-term technology because a range of technologies accumulated in fields such as oil-well drilling, underground storage of natural gas, and enhanced oil recovery (EOR) are readily adaptable.

Ongoing monitoring of these geological storage sites is mandated by regulatory authorities and will need to demonstrate volume storage integrity over long periods. A passive measure of density distribution changes in the subsurface, 4-D gravity monitoring is a technique of interest as it is low-cost, rapid, and has a minimal environmental impact. Indeed, 4-D gravity monitoring techniques have been deployed to aid hydrocarbon production monitoring in the Norwegian and North Sea sectors for approximately the last 20 years.

A. Literature Review

Conventional inversion methods find a model that has the minimum possible structure and whose gravity response fits the observed data [1], [2], [3]. The minimum structure is achieved by minimizing model roughness through a linear least squares regression, resulting in a smooth model. While the least squares regression produces smooth density models, the predicted models are often larger and exhibit smaller density values than the actual model [4], [5].

Deep learning (DL) is an emerging alternative to traditional geophysical inversion [6], [7], [8], [9], [10], [11]. Over the last several years, deep convolutional neural networks (CNNs) have achieved state-of-the-art results in a variety of computer vision applications such as image classification, segmentation, and generation [12], [13], [14]. CNNs have recently been used for inversion of seismic [15], [16], [17], [18], [19], [20], electromagnetic imaging [21], [22], and electrical resistivity data [23], [24]. Concerning gravity data, Yang et al. [25] use a 3-D U-Net to invert surface gravity data to recover synthetic subsurface density anomalies. Their results demonstrate that DL is an effective approach for inverting 2-D surface gravity data. However, the density anomalies used in this study are 3-D rectangular prisms with constant density values. These anomalies are unrealistic, and their methods are not tested on realistic-looking data. Yu-Feng et al. [26] and Huang et al. [27] use 3-D U-Net variants to invert more complex density anomalies (in terms of geometry) and apply their method to the inversion of the San Nicolas mining area. While these results are promising, they are not tested on data from CO₂ storage sites [26].

Yang et al. [28] proposed a U-Net architecture for the inversion of surface gravity for monitoring CO₂ plumes. Their model successfully detected various synthetic CO₂ plumes in most test scenarios. However, the CO₂ plumes used in this study were unrealistic regarding model shape and saturation distribution. Additionally, their proposed method maps 2-D surface gravity maps to 2-D cross sections of the 3-D...
CO₂ plumes they generate. While this study is a promising proof of concept, it does not test DL-based inversion of surface gravity data in a realistic setting. Um et al. [29] present a more realistic test of DL-based inversion for CO₂ plume monitoring. Using simulated CO₂ plumes from the onshore Kimberlina site, they developed a 2-D DL architecture to perform a joint inversion with seismic, electromagnetic, and gravity data. Additionally, they use a modified version of their architecture to invert their imaging data types individually. In each case, their DL-based approach can recover CO₂ plumes. However, their approach still does not perform DL-based inversion in a fully 3-D setting. Alyousuf et al. [30] perform a 3-D inversion of realistic, physics-simulated CO₂ plumes (Johansen formation) using three-axis borehole data as the input to their proposed architecture. Their results indicate that DL-based methods can successfully invert borehole gravity data.

B. Novel Contributions

This article makes the following novel contributions.

1) We develop an effective 3-D DL-based inversion method to recover high-resolution subsurface CO₂ plumes from surface gravity data. To the best of our knowledge, this is the first fully 3-D approach for the inversion of surface gravity data which is tested on realistic, physics-simulated CO₂ plumes.

2) We develop a postprocessing method that combines the strengths of traditional and DL-based inversion.

3) We develop a time-dependent approach for DL-based inversion of surface gravity data. To the best of our knowledge, this is the first DL-based approach to consider time dependence for inverting surface gravity data.

The article is organized as follows. Section II reviews how the data is prepared. Section III describes the methods, from pure data-driven to a time-dependent one. Section IV discusses experimental setup and results. In Section V, perspective and analysis are provided. Section VI introduces the conclusions and road map for the proposed approach.

II. DATA PREPARATION

Located 60 km offshore the west coast of Norway, the Johansen formation is a candidate CO₂ storage site whose theoretical capacity exceeds 1Gt CO₂. It consists of a roughly 100 m thick aquifer that extends up to 100 km in the north-south direction and is 60 km wide. The depth ranges from 2200 to 3100 m below sea level, which offers the perfect conditions of pressure and temperature to inject CO₂ in a supercritical state.

As part of the MatMoRA project, a corresponding reservoir model is made available online by SINTEF [31]. Starting from this initial setup, we use geostatistics to generate 500 new geological realizations that all differ in terms of porosity and permeability. In this work, we relied on a Fast Gaussian procedure (as detailed in [32]), whose main parameters are given in Table I.

For each realization, we carry out a fluid flow simulation to model the evolution of the CO₂ saturation plume within the reservoir. The scenario consists of a 100-year period of CO₂ injection at a constant rate of 14400 m³/day, followed by a 400-year period of migration. We perform these simulations using the MRST toolbox [33], which implements a Vertical Equilibrium model.

Assuming that the only changes in density occur within the reservoir (with CO₂ replacing brine), one can then write the time-lapse bulk density evolution as

$$\Delta \rho = \phi \Delta S_{CO₂} (\rho_{CO₂} - \rho_{brine})$$

where $\phi$ is the porosity, $\Delta S_{CO₂}$ is the change in CO₂ saturation, $\rho_{CO₂}$ and $\rho_{brine}$ stand for the CO₂ and brine density, respectively (see [34]).

We select a single time step from each simulation to create our final dataset. For the first 100 simulations, we randomly select a time step from the first 100 years of injection. Then, we randomly select time steps for the remaining 400 simulations across all 500 years. This process results in the selection of 500 models from which to train our DL-based methods. Fig. 1 shows the distribution of the time steps in our final dataset. Note that this distribution shows that we include more models from the 100-year injection stage in our dataset. Additionally, we treat all 500 models as a single dataset for DL-based inversion, randomly selecting from all 500 models simultaneously when creating the training, validation, and testing sets for our DL-based method.

To reduce the computational cost (i.e., memory and time) and to accommodate a larger batch size during training for our DL-based methods, we resample each 3-D plume volume from a size of 440 × 530 × 145 to a size of 128 × 128 × 128.

III. METHODS

In this section, we first introduce traditional gravity modeling since it is used to generate the surface gravity maps that correspond to the CO₂ plumes described in Section II. We then describe conventional L2 inversion in Section III-B. Section III-C, outlines our proposed DL-based approach for surface gravity inversion.

A. Modeling Gravity

Time-lapse surface gravity surveys are investigated to monitor the propagation of CO₂ plumes.

| Parameters for Geostatistics Simulation of Johansen Formation |
|-----------------|-----------------|
| Porosity         | Std 0.03         |
|                  | Lower Bound 0.10 |
|                  | Upper Bound 0.40 |
| Permeability (log) | Std 2.00        |
|                  | Lower Bound -5.0 |
|                  | Upper Bound 10.0 |
| Correlation Length | Mean 26.0       |
|                  | Std 2.00         |
| Correlation       | Porosity-Permeability 0.30 |
Given a density perturbation $\Delta \rho$ observed between a monitor and a base acquisition, the gravity field recorded at a station located at $\mathbf{r}'$ can be expressed as

$$g(\mathbf{r}') = \gamma \int \int_V \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^3} \Delta \rho(\mathbf{r}) dV$$

where $\mathbf{r}$ is the spatial coordinate, $V$ is the volume of the reservoir, and $\gamma = 6.6738480 \times 10^{-11}$ m$^3$ kg$^{-1}$ s$^{-2}$ is Newton’s gravitational constant.

In our application, we assume that only the vertical component of $g$ is recorded at stations uniformly laid out along the seabed, every 500 m in the $x$- and $y$-directions. To accurately account for the reservoir geometry, (2) is discretized by means of a finite element technique. This implementation is done under the GEOSX multiphysics framework [35].

Fig. 2 shows an example of a computed time-lapse surface gravity map with its corresponding change of density observed in the subsurface.

Data normalization is an essential preprocessing step to achieve an acceptable predictive result for DL methods. We normalize each surface gravity map by subtracting its mean and dividing by its standard deviation (i.e., $z$-score normalization).

**B. L2 Inversion of Gravity Data**

The goal of many traditional inversion methods (i.e., non-DL-based methods) is to iteratively fit the gravity response from a predicted subsurface density anomaly to the original data. More specifically, the densities in a predicted model are found by minimizing an objective function subject to fit the observed data. In the case of L2 inversion, we use an L2 formulation of the objective function. To define the objective function, we first let $F : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{m_1 \times m_2}$ be a forward gravity model that maps a density anomaly of size $n_1 \times n_2 \times n_3$ to a surface gravity response of size $m_1 \times m_2$. In its most basic form, the objective function is given by

$$O(\rho) = \frac{1}{2} \| F(\rho) - G \|^2$$

where $\rho$ is a subsurface density model, and $G$ is the observed data. Our goal then is to solve the following minimization problem:

$$\min_{\rho} O(\rho).$$

Due to the ill-posed nature of gravity inversion (i.e., several anomalies can produce the same gravity response), it is not sufficient to solve (3) in an unconstrained setting. Fig. 3 illustrates the results of unconstrained L2 inversion with a null model as an initial guess. While the predicted density anomaly produces a gravity response that fits the original data, it does not match the true density anomaly. To overcome this challenge, we constrain (3) so that only cells within the known reservoir mask are updated during minimization. We use GEOSX [35] to evaluate the gradient of the objective function, and rely on LBFGS-B to perform all L2 inversions [36].

**C. Inversion of Gravity Data With DL**

DL-based inversion is a data-driven method that builds a deep neural network model to map geophysical observations
into a subsurface physical property model such as seismic velocity, electrical conductivity, density, or saturation. A large amount of data is required to train the deep neural network model. Once trained, the DL model can predict the physical property model from measured data in near real-time. In this study, a U-Net model was developed and used to predict subsurface CO\textsubscript{2} distribution from surface gravity monitoring data based on distributions from a physics simulation of the Johansen formation. This process consists of six steps as follows.

1) Create synthetic subsurface CO\textsubscript{2} plume models (Section II).
2) Generate corresponding gravity data on the ground surface using a forward gravity model (Section III-A).
3) Build a DL architecture (i.e., U-Net).
4) Formulate a loss function to minimize during training.
5) Select hyperparameters [i.e., batch size and learning rate (LR)] and train the U-Net with the surface gravity data as input and CO\textsubscript{2} plume volumes as output.
6) Select metrics and assess the prediction accuracy of the U-Net model with test data.

1) DL Architecture: We propose a modified 3-D U-Net [37], [38] to invert 2-D surface gravity maps into 3-D plume density maps. First, via a series of 2-D convolution layers and spline interpolation, the input is resized to match the reservoir geometry. We then convert the 2-D surface maps into 3-D volumes via a pointwise convolution, where the number of channels equals the depth dimension of the volumetric output. This resultant volume is the input to a standard 3-D U-Net. The output of the U-Net splits into two separate branches. One branch is responsible for segmenting the CO\textsubscript{2} plume and the other for predicting the density values within the plume. Additionally, we apply autoencoder regularization to our network. More implementation details for this network are provided at https://github.com/aecelaya/ml-gravity-inversion.

2) Loss Function:
   a) Segmentation loss: For the segmentation branch of our architecture, we use the generalized dice loss (GDL) function proposed by Sudre et al. [40]. This loss function improves upon the original Dice loss proposed by Milletari et al. [41] by adding a weighting term for each segmentation class. Adding weight terms produced significantly better Dice scores for highly imbalanced segmentation problems. The GDL loss is given by the following:
   \[
   L_{gdl} = 1 - 2 \frac{\sum_{i=1}^{C} w_k \sum_{j=1}^{N} T_{ij}^k P_{ij}^k}{\sum_{i=1}^{C} w_k \sum_{j=1}^{N} (T_{ij}^k)^2 + (P_{ij}^k)^2}
   \]
   where \( C \) denotes the number of segmentation classes, \( N \) denotes the total number of pixels (or voxels in the 3-D case), \( P_{ij}^k \) is the \( i \)th voxel in the predicted segmentation mask for class \( k \), and \( T_{ij}^k \) is the \( i \)th voxel in the ground truth mask for class \( k \). The term \( w_k \) is the weighting term for the \( k \)th class and is given by
   \[
   w_k = \left( \frac{C}{\sum_{k=1}^{C} \frac{1}{N_k}} \right) \frac{1}{N_k}
   \]
   where \( N_k \) is the total number of pixels belonging to the class \( k \) over the entire dataset. Note that the weights \( w_k \) are precomputed and remain constant throughout training.
In our case, the number of segmentation classes equals two; background and foreground. The computed class weights are approximately 0.003 and 1.997 for the background and foreground classes, respectively.

b) Regression loss: For the regression branch of our architecture, we use the mean squared error loss function, which is given by
\[ L_{\text{reg}} = \frac{1}{N} \| \rho_T - \rho_P \|^2 \]
where \( \rho_T \) and \( \rho_P \) are the ground truth and predicted density maps, respectively.

c) Autoencoder loss: For the autoencoder branch of our architecture, we, again, use the mean squared error loss function. This loss is given by
\[ L_{\text{ae}} = \frac{1}{N} \| g_T - g_P \|^2 \]
where \( g_T \) and \( g_P \) are the ground truth and reconstructed surface gravity maps, respectively.

d) Composite loss: We take a weighted combination of the segmentation, regression, and autoencoder loss functions to produce our final loss for training. This loss function is given by
\[ L = 0.7L_{\text{reg}} + 0.25L_{\text{gll}} + 0.05L_{\text{ae}}. \tag{4} \]
We select the weights in (4) via a partial grid search driven by our observations from simple synthetic 2-D models where we vary the weight of \( L_{\text{reg}} \) from 0.3 to 0.7, the weight of \( L_{\text{gll}} \) from 0.225 to 0.675, and the weight of \( L_{\text{ae}} \) from 0.025 to 0.075. The sum of the weights is such that they equal one.

3) Training and Testing Protocols: We initialize the first layer in our modified U-Net architecture with 16 feature maps and use the Adam optimizer [42]. The initial LR is set at 0.001 with a cosine decay schedule with restarts (CDR) [43]. During training, we use a batch size of eight. Random flips and additive Gaussian noise \( (0.001 \leq \sigma \leq 0.05) \) are applied to each batch as data augmentation.

Same as with the loss function, we arrive at these specific training hyperparameters via a partial grid search over the LR \((0.001, 0.0003, \text{and } 0.0001)\), LR schedule (CDR, constant LR, and reduce LR on the plateau), and batch size \((4, 8, \text{and } 16)\). We leave all other hyperparameters at their default values. Unless otherwise specified, we use the same hyperparameters for all experiments.

To evaluate the validity of our predicted inversions, we utilize the following metrics.

1) Mean Squared Error (kg/m\(^3\)) - The mean squared error in kg/m\(^3\) between the predicted and true density anomalies.
2) Mean Squared Error (\(\mu\text{Gal}\)) - The mean squared error in \(\mu\text{Gal}\) between the input gravity map and the gravity response from the predicted anomaly.
3) R-Squared - The R-squared coefficient between the predicted and true density anomalies.
4) Dice Similarity Coefficient - A measure of overlap between the nonzero masks of the predicted and true density anomalies. The Dice similarity coefficient (Dice) ranges from zero to one, where one represents a perfect prediction [44].

Unless otherwise specified, all experiments utilize the same 90/10 train-test split with 5% of the training set used as a validation set. This split results in a training set with 450 gravity/plume pairs and a test set with 50 pairs. Our models are implemented in Python using TensorFlow (v2.8.0) and trained on four NVIDIA A100 GPUs [45].

At test time, our DL-based inversion produces predictions that are of size \(128 \times 128 \times 128\). To produce our final prediction, we resample this output to the original grid resolution of \(440 \times 530 \times 145\) via linear interpolation.

D. Combined L2-DL Inversion

To overcome the potential shortfalls of L2 inversion (i.e., large model misfit) and DL-based inversion (i.e., large data misfit), we propose using L2 inversion as a postprocessing step for our DL-driven inversion. After generating a prediction with our DL model, we use that prediction as an initial guess for L2 inversion. We apply this procedure to the 50 test time predictions from our DL-based inversion.

E. DL Inversion With Varying Sensor Resolution

Up to this point, our methods use gravity maps with a uniform sensor grid spacing of 500 m. To further assess the robustness of our DL-based inversion, we generate surface gravity maps with sensors placed every 100, 250, 500 m, 1, 2, and 3 km. We train our DL-based inversion for each sensor grid resolution.

F. DL Inversion With Saturation as Target

All of our methods so far are meant to demonstrate the ability of DL-based inversion to map surface gravity to changes in subsurface density. To further assess the reliability of our DL-based method, we change our target from density to saturation. We train our DL-based inversion using 2-D surface gravity maps (500 m sensor resolution) as input and 3-D subsurface saturation as our target.

G. DL Inversion With Time-Dependency

All the methods described above use a single gravity map to produce a predicted reconstruction of the underlying subsurface density model. These methods, however, do not consider an underlying time dependence associated with our surface gravity data. Therefore, we propose adding a convolutional LSTM network at the start of our DL architecture to process sequences of gravity data. We set our sequence length to ten, this decision is arbitrary and valid for the dataset at hand, but not necessarily practical for other cases, due to time sparsity of the monitoring surveys. The sample sequence is passed into a series of 2-D convolutional LSTM layers. The output of the last convolutional LSTM layer is converted into a 3-D volume via a pointwise convolution. This resultant volume is then passed into our modified 3-D U-Net for inversion. Fig. 5 illustrates our proposed time-dependent network.

To generate data for this time-dependent network, we take sequences of length ten from a single simulation of the Johansen formation. More specifically, for the gravity map at
Fig. 5. Sketch of our proposed time-dependent architecture. In this example, a sequence of ten gravity maps is passed into a series of 2-D convolutional LSTM layers. More implementation details for this network are provided at https://github.com/acclayla/ml-gravity-inversion.

t, we concatenate the data for time steps t₀, . . . , t₈ to make the input for this example. The corresponding output for this example is the true density model at time t₉. To create our new time-dependent dataset, we repeat this process for each subsequent time step in our simulation (i.e., t₁₀, t₁₁, . . . , t₅₀₀). This procedure results in a dataset with 490 data/model pairs. As with our previously described DL approach, we take 90% of the data as a training set, the remaining 10% as a test set, and 5% of the training set as a validation set.

IV. RESULTS

A. L2 Inversion

Table II reports the accuracy of L2 inversion using the metrics described in Section III-C3. Fig. 6 illustrates two predicted CO₂ plumes using L2 inversion.

| Inversion Type | MSE (kg/m³) [µ] | MSE (µGal) [µ] | R-Squared [†] | Dice [†] |
|---------------|-----------------|----------------|---------------|---------|
| L2           | 0.67 ± 0.39     | 0.72           | 0.90 ± 0.00   | 0.00    |
| DL           | 0.26 ± 0.15     | 0.27           | 0.47 ± 0.89   | 0.13    |
| Combined L2-DL | 0.13 ± 0.09     | 0.11           | 0.00 ± 0.00   | 0.00    |

and R-Squared and plume geometry (i.e., Dice), but does not outperform L2 inversion in terms of data misfit [i.e., MSE (µGal)]. This latter underperformance can be explained by the fact that our DL model does not attempt to fit the gravity response with the original data. Instead, our proposed DL approach directly fits the predicted CO₂ plume to the true CO₂ plume. Fig. 6 illustrates two predicted CO₂ plumes using DL-based inversion. We also performed a fivefold cross-validation to assess the robustness of our proposed DL-based inversion. In this scheme, we use 20% of the dataset for testing and train on the remaining 80% of the data. We, again, set aside 5% of the training data as a validation set. This process iterates across the entire dataset until we have test time predictions for all 500 surface gravity/plume pairs. Table III shows the results of the fivefold cross-validation. We see that these results are nearly identical to the DL results from a single fold (Table II), which indicates that our DL model learned a similar mapping from surface gravity to subsurface CO₂ plumes across multiple perturbations of the training data.

2) Combined L2-DL Inversion: Table II shows the results using L2 inversion as a postprocessing step for our DL-based inversion. Our combined approach produces predictions whose gravity response matches the original observation and further reduces the model misfit. The only metric that does not improve is the Dice coefficient. This decrease in performance for Dice can be explained by the fact that our constrained approach to L2 inversion tends to “fill” a large number of grid cells within the reservoir mask with small nonzero values to fit the gravity response with the original data. This process produces nonzero masks much larger than the true nonzero mask for the CO₂ plume, which degrades the Dice metric. As a postprocessing step, we can apply a threshold to the combined

TABLE II

ACCURACY OF TRADITIONAL L2, DL-BASED, AND COMBINED L2-DL INVERSION. THE BOLD VALUES IN EACH COLUMN INDICATE WHICH METHOD PERFORMED THE BEST FOR THAT PARTICULAR METRIC. OUR COMBINED APPROACH PRODUCES PREDICTIONS WHOSE GRAVITY RESPONSE MATCHES THE ORIGINAL OBSERVATION AND PRODUCES THE LOWEST MODEL MISFIT. THE ONLY METRIC THAT DOES NOT IMPROVE FOR OUR COMBINED APPROACH IS THE DICE COEFFICIENT, WHERE OUR DL-BASED APPROACH PRODUCES THE BEST RESULT

| Inversion Type | MSE (kg/m³) [µ] | MSE (µGal) [µ] | R-Squared [†] | Dice [†] |
|----------------|-----------------|----------------|---------------|---------|
| L2             | 0.67 ± 0.39     | 0.72           | 0.90 ± 0.00   | 0.00    |
| DL             | 0.26 ± 0.15     | 0.27           | 0.47 ± 0.89   | 0.13    |
| Combined L2-DL | 0.13 ± 0.09     | 0.11           | 0.00 ± 0.00   | 0.00    |

TABLE III

ACCURACY OF DL-BASED INVERSION OF SURFACE GRAVITY DATA USING A FIVEFOLD CROSS-VALIDATION. WE SEE THAT THESE RESULTS ARE NEARLY IDENTICAL TO THE DL RESULTS FROM A SINGLE FOLD (TABLE II), WHICH INDICATES THAT OUR DL MODEL LEARNED A SIMILAR MAPPING FROM SURFACE GRAVITY TO SUBSURFACE CO₂ PLUMES ACROSS MULTIPLE PERTURBATIONS OF THE TRAINING DATA

| Mean ± Std | Median |
|------------|--------|
| MSE (kg/m³) [µ] | 0.27 ± 0.16 | 0.26 |
| MSE (µGal) [µ] | 0.43 ± 1.12 | 0.07 |
| R-Squared [†] | 0.69 ± 0.22 | 0.74 |
| Dice [†] | 0.75 ± 0.05 | 0.76 |
method appears to improve the prediction performance for larger CO₂ plumes.

3) DL Inversion With Varying Sensor Resolution: We train our DL-based inversion for each sensor grid resolution and report the results in Table IV. The similarity of the results for each resolution indicates that our DL-based inversion can learn a mapping from surface gravity to subsurface changes in density for coarse and fine sensor grids.

4) DL Inversion With Saturation as Target: The results of DL-based inversion with saturation as a target are shown in Table V, where we see that our DL-based approach yields similar results to inversion with changes in density as the target. This similar performance shows that DL-based inversion can map changes in surface gravity to multiple subsurface physical property models.

5) DL Inversion With Time-Dependence: Table VI shows the results of training a time-dependent model. Here we see that our time-dependent DL approach can successfully recover our test cases with high accuracy in terms of model and data misfit.

V. DISCUSSION

The above introduced results indicate that our DL-based method provides an effective approach for the inversion of surface gravity data. Particularly, the DL-driven approach successfully detects and recovers a variety of realistic-looking plumes that vary significantly in shape and exhibit nontrivial density distributions. Our DL-driven inversion demonstrates improvements over the widely used conventional L2 inversion in the following aspects: our custom U-Net successfully inverts surface gravity data in a completely unconstrained setting during inference, but training relies on labels that implicitly

L2-DL prediction to improve the Dice metric. We found via a grid search that the optimal threshold is $-7 \text{ kg/m}^3$. This threshold improves the mean Dice coefficient from 0.41 to 0.62. However, this improvement comes at the cost of worse data misfit in terms of $\mu\text{Gals}$.

Fig. 6 illustrates two predicted CO₂ plumes using DL-based inversion. Visually, there is little difference between the smaller plume shown at the top of the figure and the original DL prediction ($\text{SSIM} \approx 0.95$). Still, our combined L2-DL
carry prior information which from an inversion perspective represents a constraint. Because L2 inversion is constrained to only update cells within the reservoir mask, it is unsuitable for detecting out-of-reservoir leaks, a key element to be considered when monitoring CO2 storage sites. Our DL inversion also outperforms conventional inversion in terms of model misfit and recovering the plume geometry. When combined with L2 inversion, our DL method matches L2 inversion in terms of data misfit while improving model misfit.

The DL approach consistently predicts CO2 plume models across multiple sensor resolution grids. This robustness to coarser grid data can yield significant cost savings when designing sensor grids to monitor CO2 storage sites. Additionally, our results demonstrate that DL-based inversion can successfully learn mappings to multiple geophysical property models such as saturation with little to no modification to the underlying method.

The benefits of DL-based inversion are limited from a computational perspective because the conventional gravity inversion method is not computationally demanding compared with seismic inversion. L2 inversion does not require training data and does not require high computational capacity, such as what GPUs provide. Training our DL-based inversion to convergence takes roughly 3 hours when training on 4 NVIDIA A100 GPUs, notice here that our U-Net implementation is not converged when training on 4 NVIDIA A100 GPUs, notice here that our U-Net implementation is not by any means computationally optimized. However, at test time, our DL-based method is comparable in speed to L2 inversion, taking less than one second to produce multiple predictions.

The surface gravity data does not carry explicit plume depth information, limiting the information available in the input to the U-Net model. A pseudo-depth representation created from gravity data may improve our approach. Araya-Polo’s work (published later by Chen et al. [46]) obtained depth information via a downward continuation of gravity data that transforms 2-D surface maps into 3-D data, resulting in accurate segmentation results of salt structures from surface gravity data. Liu et al. [23] added a pseudo-depth channel to the 2-D resistivity data, which improved model resolution. These works suggest that our results may benefit from including depth information in our DL-driven approach.

Jointly inverting gravity data with other data types like electromagnetic or seismic imaging may also improve the performance of our DL-driven approach. Sun et al. [47] used a combination of seismic and electromagnetic data to recover salt bodies successfully. In their approach, the joint inversion of seismic and electromagnetic data outperformed inversions with just each of the individual data type. For CO2 monitoring, Um et al. [29] use joint inversion with surface gravity data, seismic, and electromagnetic imaging to improve their results compared to inversion with gravity alone. These works suggest that joint inversion with other imaging data types is a potential avenue for improving DL-based inversion.

When experimenting with simple synthetic 2-D datasets, we found that adding a segmentation loss significantly improved the performance of our DL-based inversion. We also found this result to be true with more realistic 3-D data. While it is difficult to tell that the contribution of the segmentation branch is meaningful from the loss function since both curves take on similar values close to zero early in training, the absence of this term significantly degrades our results. For example, consider the scenario where for 2-D inversion, each subsurface model is a binary circle placed randomly inside a

| Sensor Resolution | MSE (kg/m³) [µ] | MSE (µGal) [µ] | R-Squared [↑] | Dice [↑] |
|-------------------|----------------|----------------|--------------|---------|
| 100m              | 0.26 ± 0.16    | 0.27           | 0.51 ± 1.03  | 0.11    |
| 250m              | 0.26 ± 0.16    | 0.26           | 0.53 ± 1.01  | 0.11    |
| 500m              | 0.26 ± 0.15    | 0.27           | 0.47 ± 0.89  | 0.13    |
| 1000m             | 0.27 ± 0.16    | 0.27           | 0.54 ± 1.12  | 0.11    |
| 2000m             | 0.27 ± 0.17    | 0.28           | 0.65 ± 1.23  | 0.22    |
| 3000m             | 0.27 ± 0.17    | 0.26           | 0.65 ± 1.22  | 0.22    |

Table IV
ACCURACY OF DL-BASED INVERSION FOR VARYING SENSOR GRID RESOLUTIONS. THE SIMILARITY OF THE RESULTS FOR EACH RESOLUTION INDICATES THAT OUR DL-BASED INVERSION CAN LEARN A MAPPING FROM SURFACE GRAVITY TO SUBSURFACE CHANGES IN DENSITY FOR COARSE AND FINE SENSOR GRIDS.
TABLE VII
COMPARISON OF MEAN AND STANDARD DEVIATION OF DIFFERENT METRICS FOR DL-BASED INVERSION WITH REGRESSION VERSUS JOINT SEGMENTATION-REGRESSION OUTPUT FOR SIMPLE, 2-D GRAVITY INVERSION PROBLEM

| Method                  | MSE (kg/m³) [.] | R-Squared [.] | Dice [.] |
|-------------------------|-----------------|---------------|----------|
| Regression only         | 1.6e-02 (1.6e-02) | 0.64 (0.13)   | 0.79 (0.09) |
| Segmentation-regression | 1.1e-03 (9.8e-04) | 0.90 (0.08)   | 0.94 (0.07) |

64 × 64 grid. The corresponding gravity signal will be a 1-D vector. When we generate a dataset of 1000 examples and run our proposed DL-based inversion methods with a network with a single regression versus a joint segmentation-regression output, we see an improvement across all metrics. Table VII shows this in more detail. Additionally, the work presented in [7], [47], and [48] shows that incorporating segmentation methods into DL-based inversion is beneficial.

Autoencoder branches are a common regularization technique used in 3-D medical image segmentation. Autoencoder-regularized U-Nets have repeatedly demonstrated state-of-the-art performance in various medical imaging segmentation challenges [49]. Hence, we decided to incorporate this technique into our work. The autoencoder branch resulted in faster convergence, smoother loss curves, and a slight increase in the accuracy of our model. This observation is supported by the results already seen in medical imaging problems [39], [50], [51], [52]. However, this contribution is difficult to discern based on the loss curve alone. Like segmentation loss, the autoencoder loss takes on small values close to zero early in training. The choice of our hyperparameter space described in Section III-C2 and III-C3 was guided by previous experimentation with simple 2-D synthetic models for inversion. In addition, we conducted a partial grid search to find the optimal values of the hyperparameters. We found that the optimization space was relatively flat, meaning that the choices of hyperparameters did not change the results significantly. However, we note that the hyperparameter selection process highly depends on the DL practitioner’s preferences, the dataset’s specific properties, and the computational resources available. For a given dataset, one would need to perform a hyperparameter search.

Our proposed methods are all target-oriented (i.e., site-specific), with no chance to generalize outside the prior distribution. Indeed, when we tested our model with simulated gravity data from another site, it failed to reconstruct the subsurface CO₂ plume accurately. Therefore, one would need to start by generating a simulated realization of the subsurface properties to use this DL-based inversion method on a different site. While the generation of the gravity data is negligible, the generation of the density models requires as many reservoir simulations. This process can be computationally expensive. There are ways to mitigate this cost with fast solvers like vertical equilibrium (as shown here) or surrogate models. We could also limit our investigation to early-stage injection, where the plume is bulkier than in later stages of the plume’s evolution. Generative adversarial networks (GANs) may also help extend our DL-based inversion’s generalization power. Sun et al. [48] use GANs to enrich their training data and improve the performance of their segmentation model for seismic salt segmentation. Transfer learning presents an additional mitigation strategy for the lack of diversity in a training dataset. El Zini et al. [53] and Sun et al. [48] use transfer learning in their work to improve the performance of their DL models and improve their generalization power. Transfer learning achieves faster convergence compared to random initialization of networks [54]. This speed-up may also mitigate the computational bottlenecks associated with DL-based inversion when training a model with data from a new site.

Our time-dependent results indicate that DL-based inversion can successfully incorporate time-series gravity data for monitoring sequestered CO₂. However, our results are only tested on a single simulation of the Johansen formation with a large degree of overlap between samples. Furthermore, the test set for the time-dependent dataset differs from that of our purely data-driven methods. Hence, a rigorous comparison between our time-dependent model and our other methods is unfair. Further testing of our time-dependent method on multiple simulations of the same formation and varying overlap is required before we can make rigorous conclusions about the accuracy of our proposed method. From a computational point of view, adding our LSTM network to the training pipeline does not significantly affect the time or memory required for training our network.

VI. CONCLUSION

We developed an effective DL-based inversion method to recover high-resolution, subsurface CO₂ density models from surface gravity. Our DL architecture is trained on realistic, physics-simulated CO₂ plumes and gravity data. This training approach mirrors real-world site data collection and practical gravity monitoring techniques. Our DL-based inversion outperforms traditional inversion techniques for our selected metrics, and our combined L2-DL approach provides additional benefits to our pure DL approach. While there is room for improvement, the results presented here are promising and represent, to the best of our knowledge, the first fully 3-D DL-based inversion of surface gravity data derived from a physics simulation of a proposed CO₂ storage site.

However, 4-D CCS monitoring with gravity (and other nonseismic methods) is a data-poor field. Real acquired data is rare, and access is limited. Sufficiently detailed reservoir models for CCS are also challenging to come by. With this in mind, this manuscript is as much an argument for a new data acquisition method as it is a new inversion method. We hope our results, which demonstrate the efficacy of nonseismic methods for CCS monitoring, can help convince others to acquire the data required to rigorously test these methods on field data.

REFERENCES

[1] C. de Groot-Hedlin and S. Constable, “Occam’s inversion to generate smooth, two-dimensional models from magnetotelluric data,” Geophysics, vol. 55, no. 12, pp. 1613–1624, Dec. 1990.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 61, 2023

[2] Y. Li and D. W. Oldenburg, “3-D inversion of gravity data,” Geophysics, vol. 63, no. 1, pp. 109–119, Jan. 1998.

[3] M. Nabighian et al., “57th anniversary,” in The Historical Development of the Magnetic Method in Exploration: Geophysics, vol. 70, no. 6, 2005.

[4] O. Boulanger and M. Chouteau, “Constraints in 3D gravity inversion,” Geophys. Prospecting, vol. 49, no. 2, pp. 265–280, Mar. 2001.

[5] M. Rezaie, A. Moradzadeh, and A. N. Kalateh, “Fast 3D inversion of gravity data using solution space preconditioned Lanczos bidiagonalization,” J. Appl. Geophys., vol. 136, pp. 42–50, Jan. 2017.

[6] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, “Deep convolutional neural network for inverse problems in imaging,” IEEE Trans. Image Process., vol. 26, no. 9, pp. 4509–4522, Sep. 2017.

[7] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, “Deep-learning tomography,” Lead. Edge, vol. 37, no. 12, pp. 894–901, Dec. 2018.

[8] L. R. Dice, “Measures of the amount of ecological association between species,” Ecology, vol. 26, no. 3, pp. 355–366, Jul. 1945.

[9] B. Liu, S. Yang, Y. Ren, X. Xu, P. Jiang, and Y. Chen, “Deep-learning seismic full-waveform inversion for realistic structural models,” Geosci. Front., vol. 1, no. 1, pp. R31–R40, Jan. 2021.

[10] Y. Ren, L. Nie, S. Yang, P. Jiang, and Y. Chen, “Building complex seismic velocity models for deep learning inversion,” IEEE Access, vol. 9, pp. 63767–63778, 2021.

[11] Y. Wu and Y. Lin, “InversionNet: An efficient and accurate data-driven full wave inversion,” IEEE Trans. Geosci. Remote Sens., vol. 63, no. 2, pp. 1035–1056, Dec. 2020.

[12] Y. Li and D. W. Oldenburg, “3-D inversion of gravity data,” Geophysics, vol. 50, no. 5, pp. 917–926, Oct. 1985.

[13] F. Chollet. (2015). Keras. [Online]. Available: http://www.keras.org

[14] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, “3D U-Net: Learning dense volumetric segmentation from sparse annotation,” in Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent. Cham, Switzerland: Springer, 2016, pp. 424–432.

[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv:1412.6980, 2014, arXiv:1412.6980.

[16] L. R. Dice, “Measures of the amount of ecologic association between species,” Ecology, vol. 26, no. 3, pp. 355–366, Jul. 1945.

[17] C. H. Sudre, W. Li, T. K. M. Vercauteren, S. Ourselin, and M. K. I. Dodoo, “Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations,” in Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Cham, Switzerland: Springer, 2017, pp. 240–248.

[18] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-Net: Fully convolutional neural networks for volumetric medical image segmentation,” in Proc. 4th Int. Conf. 3D Vis. (3DV), Oct. 2016, pp. 565–571.

[19] P. John and M. L. Schmitt, “Large Volumes: Deep learning on an image to image data set,” in Proc. IEEE Int. Conf. Learn. Represent. 2017, pp. 1–15.

[20] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv:1412.6980, 2014, arXiv:1412.6980.

[21] D. Colombo, W. Li, E. Sandovalo-Cariel, and G. W. McNeice, “Deep-learning electromagnetic monitoring coupled to fluid flow simulators,” Geophysics, vol. 85, no. 4, pp. WA1–WA12, Jul. 2020.

[22] S. Oh, K. Noh, S. J. Seol, and J. Byun, “Cooperative deep learning of full waveforms inversion: Connecting CNN and partial differential equation in a loop,” 2021, arXiv:2110.07584.

[23] Y. Wu and Y. Lin, “InversionNet: An efficient and accurate data-driven full wave inversion,” IEEE Trans. Comput. Imag., vol. 6, pp. 419–433, 2020.

[24] M. Shahriari, D. Pardo, A. Picon, A. Galdran, and J. D. Ser, “Identifying the best machine learning algorithms for geophysical inversion versus machine learning,” IEEE Trans. Geosci. Remote Sens., vol. 61, no. 4, pp. 450–460, Dec. 2021.

[25] T. Alyousuf, Y. Li, and R. Krahenbuhl, “Machine learning inversion of seismic and electrical data for salt reconstruction,” in Proc. SPE Annu. Tech. Conf. Exhib., 2020, pp. 1–20.

[26] T. Alyousuf, Y. Li, and R. Krahenbuhl, “Machine learning inversion of seismic and electrical data for salt reconstruction,” in Proc. SPE Annu. Tech. Conf. Exhib., 2020, pp. 1–20.

[27] R. Huang, S. Lian, R. Qi, and Y. Zhang, “Deep learning 3D sparse inversion of gravity data,” J. Geophys. Res., Solid Earth, vol. 126, no. 11, Nov. 2021, Art. no. e2021JB024276.
[53] J. El Zini, Y. Rizk, and M. Awad, “A deep transfer learning framework for seismic data analysis: A case study on bright spot detection,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 5, pp. 3202–3212, May 2020.

[54] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning (Adaptive Computation and Machine Learning). Cambridge, MA, USA: MIT Press, 2016.

Adrian Celaya is currently pursuing the Ph.D. degree with the Department of Computational and Applied Mathematics (CAAM), Rice University, Houston, TX, USA.

He is co-advised by Dr. Beatrice Riviere and Dr. David Fuentes. He was a Research Intern at TotalEnergies, Paris, France, from May 2022 to August 2022. Prior to joining Rice University, he served as the Information System Security Manager onboard the U.S. Navy aircraft carrier USS Carl Vinson. After leaving the Navy in August 2020, he became a Research Assistant with the Department of Imaging Physics, The University of Texas MD Anderson Cancer Center, Houston. His research focuses on developing novel techniques for 3-D medical image segmentation.

Bertrand Denel received the Ph.D. degree in applied mathematics from the University of Pau, Pau, France, in 2004.

Subsequently, he completed a postdoctoral research fellowship at the University of Houston, Houston, TX, USA. In 2006, he joined TotalEnergies, Paris, France, and has since been working on developing numerical methods for subsurface imaging, including reverse time migration (RTM) and full waveform inversion (FWI), from HPC implementation to field applications in production. He is a Research Geophysicist with more than 15 years of experience in the field. Currently, he is leading multiphysics and joint inversion research projects aimed at improving our understanding of subsurface reservoirs, with a specific focus on CO₂ sequestration.

Yen Sun received the master’s degree in geophysics from Rice University, Houston, TX, USA, and the Ph.D. degree from the Physics and Astronomy Department, Rice University.

She is a Research Geophysicist at TotalEnergies, Paris, France, where she works on various research projects including multiphysics joint inversion, global optimization with uncertainty estimation and developing deep learning tools for subsurface model building, and image processing. Prior to her current role, she worked as a Post-Doctoral Fellow at the Harvard Medical School, Boston, MA, USA, and Rice University, where she studied soft matter physics with multidisciplinary approaches.

Mauricio Araya-Polo received the bachelor’s degree in computer engineering from the University of Chile, Santiago, Chile, and the M.Sc. and Ph.D. degrees from INRIA-UNSA, Sophia Antipolis Cedex, France.

He is a Senior Computer Scientist and a Senior Research and Development Manager at TotalEnergies EP Research and Technology USA, Houston, TX, USA. He is also an Adjunct Professor at the Earth Environmental and Planetary Science Department, Rice University, Houston, where he is also a Lecturer with the Professional Science master’s Program, Weiss School of Natural Science, where he teaches computational geophysics. In his role at TotalEnergies, he is currently leading efforts on diverse areas such as machine learning and high-performance computing (HPC). Previously, he led geophysics and machine learning efforts for Shell International E&P Inc., Houston. Before that, he worked at Repsol USA, Warrendale, PA, USA, researching near real-time visualization/computing and HPC for geophysical algorithms. Before that position, he was a Senior Computer Scientist at the Barcelona Supercomputing Center for the Kaleidoscope project.

Antony Price received the degree in physics and geology from the Gravity Gradient Research Group, University of Western Australia, Crawley WA, Australia, in 1994, working on novel processing and interpretation technology for gravity gradiometry—earning the award for best technical presentation at the ASEG student meeting.

His professional career started in 1995, working with airborne geophysics encompassing environmental and mineral exploration. This was followed by marine FTG and Robertsons Research, moving to hydrocarbon exploration. He currently holds the position of a Senior Specialist (Réalent) for nonseismic geophysics at TotalEnergies, Paris, France.

Dr. Price has more than 25-years in a career with integrated geophysics with varied global postings and as an active SEG member since 2003 participating in international conferences as a reviewer, a presenter, a workshop organizer, the session chair, and the vice chair of the SEG European Regional Advisory Committee, the SEG Technical Assessment Group, and with a nomination to the EAGE Distinguished Lecturer Program in 2016.