EARTH SCIENCES
RESEARCH JOURNAL
Earth Sci. Res. J. Vol. 28, No. 4 (December, 2024): 421 - 426

UNIVERSIDAD NACIONAL DE COLOMBIA

GEOPHYSICS

Three-Dimensional Gravity Interface Inversion Based on Artificial Neural Network and Discrete Cosine Transform Algorithm

Huadong Song, Hui Fu*, Jifeng Qin
Huanghe Science and Technology University, Zhengzhou 450003, China
*Corresponding Author: 17630076382@hhstu.cn

ABSTRACT

This study applies artificial neural networks to three-dimensional gravity density interface inversion. Parker's formula based on fast Fourier transform method plays an important role in gravity interface inversion. In the training process of the artificial neural network, random generated underground interface geometries are used as inputs, while the output is the gravity anomaly data calculated by Parker's formula. A large-scale input-output training set is formed for the training process of the artificial neural network. In addition, discrete cosine transform (DFT) is introduced to compress and store matrices, which reduces computational memory, decreases computation time, and improves computational efficiency in the training and testing processes of the artificial neural network. A deep learning interface inversion algorithm based on the U-net network model is designed. On the basis of the traditional loss function, a smooth loss term and an overfitting suppression term are added to improve the smoothness and convergence efficiency of the gravity interface inversion results. Finally, the inversion prediction is verified through the test sample set to validate the generalization of the established deep learning network model. This paper analyzes the effectiveness and practicality of this method in density interface inversion through theoretical models and actual data experiments. The deep learning interface inversion method based on the improved loss function constraint effectively improves the convergence efficiency and computational stability of density interface inversion. Applying this method to synthetic data and actual measured data processing has achieved good results.

Keywords: Neural Network; Discrete Cosine Transform; Application; 3D Gravity

Inversión de la interfaz gravitacional tridimensional con base en los algoritmos de transformación de redes neuronales artificiales y coseno discreto

RESUMEN

Este estudio aplica redes neuronales artificiales a la interfaz de densidad de la inversión de gravedad tridimensional. La formula de Parker basada en la transformada rápida de Fourier juega un papel importante en la interfaz de inversión de la gravedad. En el proceso de entrenamiento de las redes neuronales artificiales, las geometrías de interfaz subterráneas generadas aleatoriamente se usaron como registros, mientras que los resultados son la información de anomalías de la gravedad calculadas con la formula de Parker. Un amplio conjunto de datos de registro y resultado se definieron para el proceso de entrenamiento de la red neuronal artificial. Adicionalmente, la transformada de coseno discreta se presenta para comprimir y almacenar matrices, y así se reduce la memoria computacional, se reduce el tiempo de computación y se mejora la eficiencia del equipo en los procesos de entrenamiento y de prueba de la red neuronal artificial. De esta forma se diseñó un algoritmo de aprendizaje profundo para la interfaz de inversión basado en el modelo de red tipo U. Con base en la tradicional función de pérdida se adicionaron un término de pérdida suave y un término de sobreajuste de supresión para mejorar la eficiencia de la uniformidad y de la convergencia de los resultados de la interfaz de inversión de la gravedad. Finalmente la predicción de la inversión se verificó a través del ejemplo de prueba para validar la generalización del modelo de red de aprendizaje profundo que se estableció. Este artículo analiza la efectividad y la practicidad de este método de inversión de la interfaz de densidad a través de modelos teóricos y experimentos con información real. Este método de inversión de interfaz de aprendizaje profundo, basada en la restricción de función de perdida mejorada, efectivamente mejora la convergencia de eficiencia y la estabilidad computacional de la inversión de la interfaz de densidad. La aplicación de este método a procesos de información sintética y de información medida ha alcanzado buenos resultados.

Palabras clave: Red neuronal; transformada discreta de coseno; aplicación; modelo de gravedad 3D

Record
Manuscript received: 19/11/2024
Accepted for publication: 27/01/2025

How to cite item:
Song, H., Fu, H., & Qin, J. (2024). Three-Dimensional Gravity Interface Inversion Based on Artificial Neural Network and Discrete Cosine Transform Algorithm. *Earth Sciences Research Journal, 28(*4), 421-426. https://doi.org/10.15446/esrj.v28n4.115333

ISSN 1794-6190 e-ISSN 2339-3459
https://doi.org/10.15446/esrj.v28n4.115333

1. Introduction

A classic problem in geophysics is the determination of the geometric shape of density interfaces related to gravity anomalies, such as the basement shape of sedimentary basins, the Moho interface, and seafloor topography. These studies provide valuable information for understanding deep-sea geological structures, regional tectonics, basin formation, seafloor topography formation processes, and many other geological and geodynamic phenomena. However, estimating the geometric shape of a specific geological structure's interface using gravity data is an ill-posed and non-unique problem. It requires careful consideration of the selection of input gravity data, pre-processing, inversion parameters (such as average depth and interface density), and other geophysical constraints to stabilize gravity inversion.

The Bott method (Bott, 1960) was initially used to solve this problem. This method is based on iterative solutions and assumes constant density differences. The initial approximate value of sediment thickness corresponding to each gravity reference position is obtained using the flat plate formula. The fitting difference between the observed data and the predicted data is then converted into a correction to the initial model (using the flat plate formula again), and the iteration process stops when the fitting difference drops below a certain threshold. Cordell and Henderson (1968) modified the iteration step length in the Bott method and introduced the ratio between the previously iterated model-generated observed data and calculated data to update the model. Silva et al. (2014) suggested accelerating convergence by dividing the residual (i.e., the difference between the observed gravity data and the currently calculated gravity data) by an arbitrarily larger positive value (no longer using the slab formula), and accepting the obtained model when the L2 norm of the residual vector is less than the previous iteration's norm. Many scholars have adopted similar methods, which will not be described in detail here.

Currently, gravity inversions are primarily based on structured grids, and research on gravity inversions based on unstructured grids has only emerged as a field of study in the last decade or so, with relatively limited literature available for reference. Among them, the research achievements of Memorial University of Newfoundland in Canada stand out in this direction. Lelièvre et al. (2012) conducted a systematic study on the generation and storage of unstructured grids, as well as forward modeling and inversion constraints based on unstructured grids, and successfully applied them to the problem of gravity inversion. Subsequently, Sun et al. (2021) also conducted research on gravity inversions based on unstructured grids and discussed the role of depth weighting functions and fuzzy c-means clustering algorithms in inversion. Ma et al. (2022) utilized the tilt angle method to conduct local adaptive grid research on gravity inversion based on unstructured grids, improving inversion efficiency by reducing the number of grids.

The aforementioned methods are all calculations performed in the spatial domain. However, when the model is complex or there are a large number of observation data, this process is computationally intensive. Since the advent of the fast Fourier transform, many scholars have tried to apply it to gravity data inversion. Parker (1972) provided a forward formula for applying the fast Fourier transform in the gravity method, which is characterized by its speed and high accuracy. With the advancement of computers, Parker's method based on the fast Fourier transform has become feasible for fast computing of magnetic or gravity fields. Oldenburg (1974) proved that Parker's expression can be used to determine the geometric shape of the interface by gravity anomaly data. Granser (1987a) rearranged the forward algorithm to obtain the depth of the sedimentary basin basement. This method is similar to the inversion program developed by Oldenburg. Granser (1987b) also applied another nonlinear frequency-domain inversion method based on Schmidt-Lichtenstein to the case of a 3D sedimentary basin with uniform density differences. Although algorithms based on the fast Fourier transform reduce computation time, they require an average depth of the interface and a low-pass filter to achieve convergence in the inversion process (Oldenburg, 1974; Granser, 1987a; Granser, 1987b). Based on the Parker-Oldenburg formula, some authors have developed computer programs to determine the geometric shape of density interfaces from gravity anomaly data. Nagendra et al. (1996) developed Fortran language program codes based on the Parker-Oldenburg method and the Granser method, respectively, for analyzing two-dimensional gravity anomaly data. Gómez-Ortiz and Agarwal (2005) proposed a Matlab language program code for the three-dimensional extension of the Parker-Oldenburg method to obtain the geometric shape of density interfaces.

The linear inversion method is simple and widely used due to its linearization process. However, it is highly dependent on the selection of initial parameters, leading to a decrease in inversion accuracy. Therefore, in recent years, non-linear inversion methods have attracted the attention of many geophysicists. Artificial neural networks (ANNs) are one of the most active non-linear inversion methods, as they can approximate a non-linear continuous function with any desired accuracy. ANNs use the input variables as independent variables, and the output variables as dependent variables, and can replace this function with any high degree of accuracy (Van der Baan and Jutten, 2000). Since the late 1980s, artificial neural networks (ANNs) have been applied to geophysical problems (Raiche, 1991) and increasingly used to interpret geophysical data. ANNs, as a computer processing method, have shown great potential for pattern recognition in geophysical exploration problems, such as log interpretation (Wiener et al., 1991), electromagnetic profiling data processing (Poulton, 1992; Winkler, 1994), ground penetrating radar (Poulton & El-Fouly, 1991), seismic waveform recognition (Dai & MacBeth, 1995; Ashida, 1996), and more. In recent years, the petroleum industry has been using ANNs to process measured data. ANNs have also been applied in the interpretation of gravity and magnetic data.

Over the past few decades, with the continuous improvement of computer performance, the application of machine learning in gravity exploration has gradually developed into a hot research direction. Especially after Adler & Oktem (2017) and Jin et al. (2017) demonstrated that machine learning can solve ill-posed problems, various geophysical inversion methods based on machine learning have emerged like mushrooms after a spring rain. Carpenter (2021) was the first to establish a mapping relationship between gravity data and salt structure through machine learning, avoiding the iterative process of conventional gravity inversion; Wang et al. (2021) used U-Net to complete the three-dimensional inversion of gravity data in the St. Nicholas mining area, and proved through comparative experiments that the machine learning inversion results have higher resolution and clearer boundary positions than conventional inversion. Li et al. (2022) used a self-supervised deep learning method to establish a new SSGI network. This method incorporates the basic theory of the gravity field into the inversion, optimizing by minimizing the difference between the observed gravity anomalies and the reconstructed gravity anomalies, reducing the uncertainty of inversion. Zhang et al. (2022) established a new convolutional neural network-DecNet for gravity data inversion, transforming the problem of mapping from two dimensions to three dimensions into four problems of mapping from two dimensions to two dimensions. This method has the advantages of high accuracy, low memory requirements, and fast convergence.

This article presents a study on the inversion of 3D gravity interfaces by combining the Parker formula with artificial neural networks, and utilizing the discrete cosine transform algorithm. The article begins with a review of the classic Parker-Oldenburg formula, followed by an explanation of the artificial neural network algorithm process and the principles of the discrete cosine transform algorithm. The effectiveness of the proposed algorithm is then demonstrated through synthetic and measured examples. Finally, the article concludes with a summary of the algorithm and suggests possible improvements.

2. Methods

2.1 Parker-Oldenburg formula

The Parker formula, derived by Parker in 1973 and 1974, provides a Fourier transform for calculating gravity anomalies caused by a subsurface density distribution. The Equation 1 is expressed as follows:

$$F[\Delta g] = 2\pi G\rho e^{(-|\vec{k}|z_0)} \sum_{n=1}^{\infty} \frac{|\vec{k}|^{n-1}}{n!} F[h(\vec{r})^n] \quad (1)$$

Here, F is the Fourier transform operator, G is the universal gravitational constant, ρ is the density contrast on the interface, $\vec{k}$ is the wave vector, $h(\vec{r})$ is the interface depth, and z_0 is the average depth of the horizontal interface.

Oldenburg rearranged the formula in 1974, and through an iterative process, the depth of the undulating interface can be obtained as follows:

$$F[h(x)] = -\frac{F[\Delta g(x)]e^{(-k-i)}}{2\pi G\rho} - \sum_{n=2}^{\infty} \frac{k^{n-1}}{n!} F[h^n(x)] \tag{2}$$

In this inversion process, we need to assume the average depth z_0 of the interface and the density contrast ρ. First, we substitute $h(x) = 0$ into the second term on the right-hand side of equation (2) and calculate the first term on the right-hand side. Then, we perform the inverse Fourier transform to obtain the first approximate value of $h(x)$. We then substitute this value into the right-hand side of equation (2) and iteratively solve until a suitable solution is obtained.

2.2 Artificial neural networks

In geophysics, given an underground model and a forward function , the corresponding geophysical data can be represented by the Equation 3:

$$\mathrm{d} = \mathrm{g(m)} \tag{3}$$

The inversion process is to find an optimal model whose forward response data can best fit the observed data d_{obs}.

Artificial neural networks consist of network units called neurons, which are arranged in layers. Generally, there are input layers, hidden layers, and output layers. When the neural network is fully connected, each neuron in the input layer is connected to any neuron in the hidden layer, and each neuron in the hidden layer is also connected to any neuron in the output layer. The connection method is expressed in a mathematical expression of weighted sum, and is subjected to the action of an activation function in the hidden layer.

Taking the input vector x^n as an example, the mathematical expression a_m^n corresponding to the m_{th} neuron in the hidden layer is:

$$a_m^n = f^n\left(\sum_{i=1}^{N} \omega_{mi}^{\mathrm{h}} x_i^n + b_m\right) \tag{4}$$

Where f represents the activation function, $m = 1,2,\cdots,$ L, the superscript h represents the hidden layer, ω_{mi}^{h} represents the weighting coefficient of the i_{th} neuron in the input layer for the m neuron in the hidden layer, x_i^n represents the i_{th} component of the input vector x^n, and b_m represents the bias term of the m_{th} neuron in the hidden layer.

The activation function f is a bounded nonlinear function, and the Sigmoid function is commonly used:

$$f(v) = \frac{1}{1+\exp\,(-\alpha v)} \tag{5}$$

Where controls the rate at which the function value tends to 0.5. The activation function endows the neural network with highly nonlinear expressive power, making it suitable for solving nonlinear problems.

Similar to equaion (4), the k_{th} neuron O_k^n in the output layer can be represented as:

$$O_k^n = f^{\circ}\left(\sum_{m=1}^{L} \omega_{km}^{\circ} a_m^n + c_k\right) \tag{6}$$

Where $k = 1,2,\cdots, M$, the superscript O represents the output layer, ω_{km}^{O} represents the weighting coefficient of the m_{th} neuron in the hidden layer for the k_{th} neuron in the output layer, and c_k represents the bias term of the k_{th} neuron in the output layer.

The vector is the expected output of the neural network, and there is an error between it and the actual output of the neural network:

$$E = \frac{1}{2}\sum_{n=1}^{Q} \sum_{k=1}^{M} (y_k^n - O_k^n)^2 \tag{7}$$

The training of the neural network is an optimization process with E as the objective function. Common optimization algorithms include Levenberg-Marquardt algorithm, conjugate gradient method, simulated annealing method, and genetic algorithm, etc.

2.3 U-net Neural Network

The U-net neural network model used in this paper is shown in Figure 1. The U-net network model extracts detailed features through a down-sampling process and restores dimensions through an up-sampling process. It concatenates shallow background information with deep local data, ensuring that positional and other shallow layer information is well retained until the data output, making the network more sensitive to positional information (Ronneberger et al., 2015). Additionally, as a fully convolutional neural network, U-net abandons complex fully connected layers and directly outputs through a 1×1 convolutional layer, making the network parameters simpler and easier to train. In the gravity data density interface inversion, there is a clear positional feature between the input and output data. Using the U-net neural network can more effectively complete the inversion of the interface undulation depth.

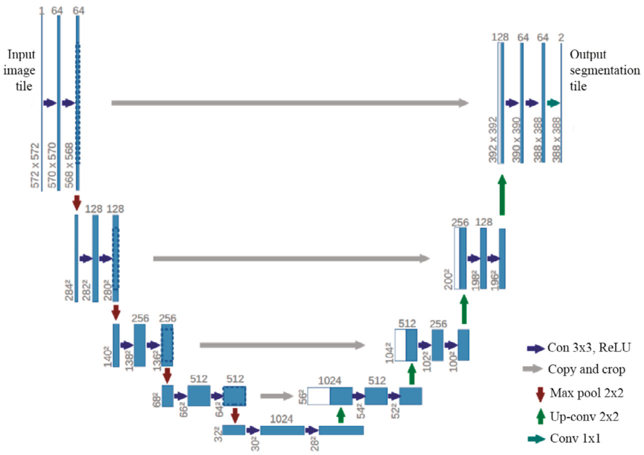

Figure 1. U-net neural network structure

In this paper, Adam, which has strong robustness, is adopted as the optimization algorithm, and the minimum square error Φ_{mse} is chosen as the basic loss function. Moreover, since it is assumed that the underground density interface is smooth, a smoothness loss term Φ_{smooth} is added to the basic loss function to penalize rough results. At the same time, to address the overfitting issue during training, an L2 norm is introduced to constrain the network parameters. The final expression of the loss function proposed in this paper is:

$$\Phi(\theta, h_{\mathrm{pre}}) = \Phi_{\mathrm{msee}}(h_{\mathrm{pre}}) + \alpha\Phi_{\mathrm{smooth}}(h_{\mathrm{pre}}) + \beta\Phi_{L_2}(\theta) \tag{8}$$

Where

$$\begin{cases} \Phi_{\mathrm{mse}} = \| h_{\mathrm{pre}} - h \|_2^2 \\ \Phi_{\mathrm{smooth}} = \| h_{\mathrm{pre}} \|_2^2 \\ \Phi_{L_2} = \| \theta \|_2^2 \end{cases} \tag{9}$$

In the context provided, θ represents the weights and biases of the neural network, while h_{pre} is the interface depth data obtained from the neural network inversion. The h is the desired output interface data, which is generated by the generator and obtained after preprocessing.

2.4 Discrete cosine transform

Discrete cosine transform (DCT) is widely used in image compression. DCT represents an image as a sum of sine waves with different amplitudes and frequencies. By performing a discrete cosine forward transform on the original object, the transformed coefficients are mainly concentrated in the upper left corner, and most of the remaining coefficients are close to zero. The threshold operation is performed on the transformed DCT coefficients, and coefficients smaller than a certain value are set to zero, which achieves the quantization process in image compression. Then, the discrete cosine inverse transform is performed to obtain the compressed image.

The 2D DFT formula for an input image data matrix A with dimensions $M * N$ is:

$$B_{pq} = \alpha_p a_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix} \quad (10)$$

The value B_{pq} is the DCT coefficient of the image data matrix A.

DCT is a reversible transform, and its inverse transform (IDFT) equation is:

$$A_{mn} = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \alpha_p \alpha_q B_{pq} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \begin{matrix} 0 \le m \le M-1 \\ 0 \le n \le N-1 \end{matrix} \quad (11)$$

Where

$$\alpha_p = \begin{cases} 1/\sqrt{M}, p = 0 \\ \sqrt{2/M}, 1 \le p \le M-1 \end{cases} \quad (12)$$

$$\alpha_q = \begin{cases} 1/\sqrt{N}, q = 0 \\ \sqrt{2/N}, 1 \le q \le N-1 \end{cases} \quad (13)$$

Assuming the randomly generated 2D subsurface medium as input data, its large dimension requires a large number of neural network units for training, which results in slow training speed. To solve this problem, we compress its data matrix based on discrete cosine transform to reduce the dimension of the data. For a 2D input data matrix, the energy of the DCT coefficients is mainly concentrated in the low-frequency part, and the high-frequency part is close to zero. In this study, we first use only the low-frequency coefficient parts of the input and output for training to obtain an optimal neural network. Then, we use this neural network to predict new inputs and obtain the low-frequency coefficients of the new output. Finally, we use the inverse discrete cosine transform to reconstruct the full-frequency data.

3. Application

3.1 Synthetic data

During the neural network training process, we used randomly generated underground interfaces to create a training set of 100,000 "interface model-gravity anomaly data" pairs, and then trained an artificial neural network. We used randomly synthesized underground interfaces to verify the correctness of the algorithm proposed in this paper. The average depth of the model is 25 km, with a range of 250 km × 250 km. In this model, the northeast is a large dome with a minimum interface depth of about 16 Km, and the south consists of several discontinuous basins with a maximum depth of about 24 Km. The density contrast of the interface is 0.4g/cm^3, and the average depth of the interface is 20 Km. The observation points on the surface are spaced "1 km ×1 km" apart, and the underground medium is divided into a grid of 256 × 256 with a grid spacing of "1 km". A 5% Gaussian random error was added during the artificial neural network training process.

During the training process, the discrete cosine transform algorithm was used to compress the data matrix. The size of the observation data matrix is known to be 250 × 250, and the size of the underground medium matrix is 256 × 256. Figure 2(a) and 2(b) show the discrete cosine transform coefficients of the true depth model and the observation data, respectively. It can be seen from the figures that the energy of the coefficients after the discrete cosine transform is mainly concentrated in the upper left corner (shown in the red box, at approximately position 10 × 10), and most of the other coefficients are close to zero. Only the coefficients within the red box were used during the artificial neural network training, compressing the 256 × 256 data matrix into a 10 × 10 data matrix, greatly reducing the computational memory, shortening the calculation time, and improving the efficiency of artificial neural network training. By comparison, we know that training without discrete cosine transform compression takes about 68,000 seconds, while after compression, the neural network training only takes about 2,400 seconds.

From Figure 3(a) and 3(c), it can be seen that the inversion results of the synthetic model match well with the true model, and both the minimum interface depth of 16 Km and the maximum interface depth of 24 Km can be well recovered. In addition, as shown in Figure 3(b) and 3(d), the gravity anomaly data calculated from the inverted depth interfaces are also consistent with the observed gravity anomaly data. Figure 3(e) and 3(f) show the differences in depth interface models and gravity anomaly data, respectively. It can be seen that the differences in the depth interface models are very small, with a range of approximately -1.0 ~ 1.0Km, while the differences in the gravity anomaly data are also very small, with a range of approximately -2.5 ~ 1.5 mGal.

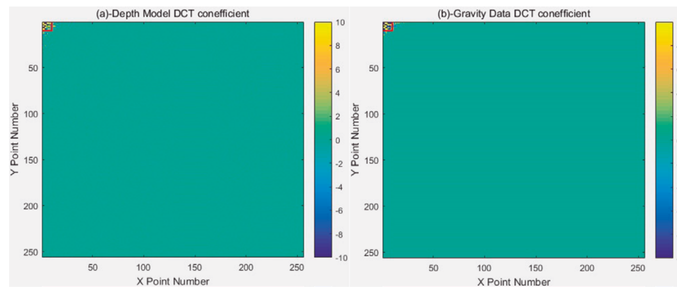

Figure 2. Discrete cosine transform coefficient of the random medium model (a)-depth model DCT coefficient (b)-gravity data DCT coefficient

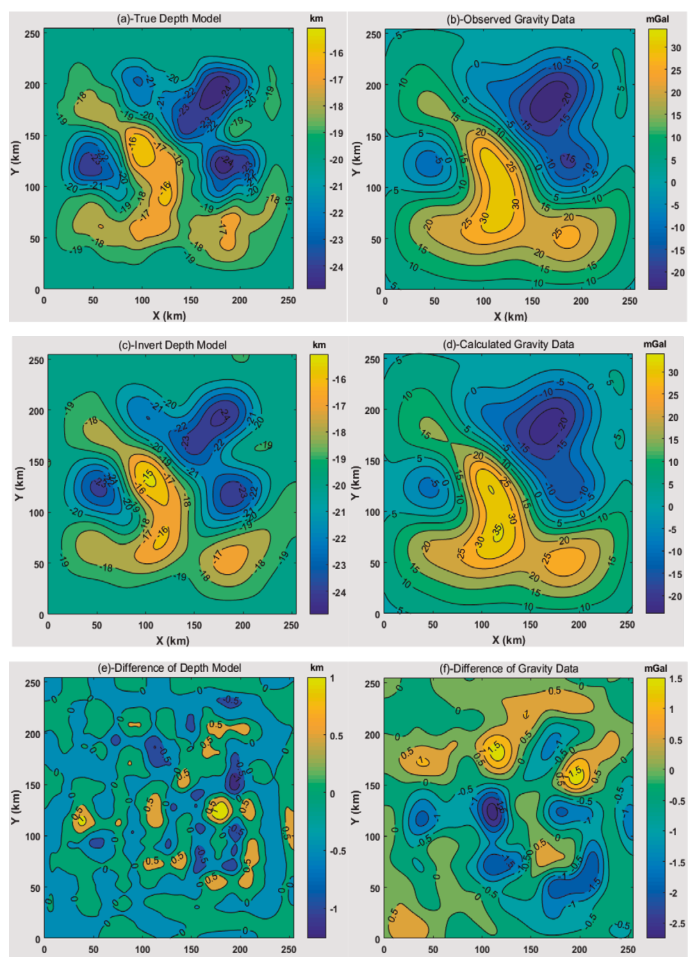

Figure 3. The random medium model
(a)-true depth model (b)-observed gravity
(c)-invert depth model (d)-calculated gravity
(e)-difference of depth model (f)-difference of gravity data

3.2 Field data

The real-world example comes from the open-source data in the article by Gomez-Ortiz and Agarwal (2005), including a subset of gravity anomaly data and the inverted depth model from the Brittany region of France. The area of this region is 300 × 300Km, with two discontinuous basins from northwest to southeast and several dome structures around them. The minimum interface depth is about 27.5 Km and the maximum interface depth is about 32 Km. The density difference between the crust and mantle is 0.4g/cm^3, and the average depth of the crust-mantle boundary is 30 km. The observation point interval is 6 km, and the underground medium grid is divided into 112 × 112 with a grid spacing of 6 km.

The training process of the artificial neural network designed for this real-world data also used the discrete cosine transform to compress the data matrix. In this real-world example, the size of the observation data matrix is 112 × 112, and the size of the underground medium grid is also 112 × 112. As shown in Figure 4(a) and 4(b), the energy of the coefficients after the cosine transform is mainly concentrated in the upper left corner (shown in the red box, at approximately position 15 × 15), and most of the other coefficients are close to zero. After applying the discrete cosine transform algorithm, we compressed the 112 × 112 data matrix into a 15 × 15 data matrix, greatly improving the efficiency of artificial neural network training. After comparison, we find that training without discrete cosine transform (DCT) compression takes approximately 24,000 seconds, whereas after compression, the neural network training only requires about 3,600 seconds.

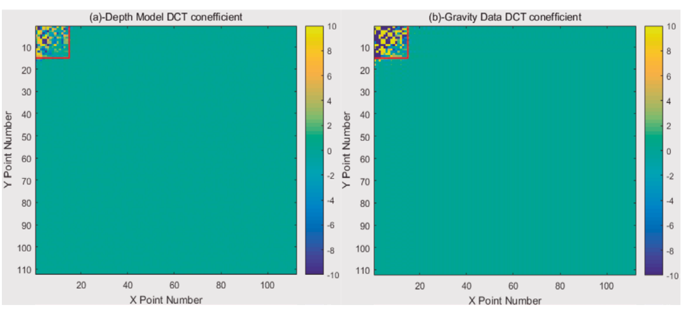

Figure 4. Discrete cosine transform coefficient of the Gomez-Ortiz and Agarwal's invert depth model (a) and Brittany's observed gravity data (b)

Figure 5 shows the comparison between the results obtained by applying our algorithm and the results in the article by Gomez-Ortiz and Agarwal (2005). As can be seen from the comparison between Figure 5(a) and 5(c), the morphology and numerical values of the inversion results obtained by our algorithm are basically consistent with those in the article by Gomez-Ortiz and Agarwal (2005). From Figure 5(b) and 5(d), it can also be seen that the gravity anomaly data calculated from the inversion results obtained by our algorithm also match well with the observed gravity anomaly data. Figure 5(e) and 5(f) further demonstrate the practicality of our algorithm. The difference range of the depth model is approximately between -1.5 ~ 2.0 Km, while the difference range of the gravity anomaly data is approximately between -3.0 ~ 2.0 mGal.

4. Conclusion

In this paper, we introduced the artificial neural network algorithm and the discrete cosine transform algorithm into the research of three-dimensional gravity interface inversion. By using the method of random modeling, the underground interface geometry shape is randomly generated as the output, and the gravity anomaly data calculated based on Parker formula using fast Fourier transform is used as the input, forming a large-scale input-output training set for the training process of the artificial neural network. In order to solve the problem of long computation time and high memory requirements faced by artificial neural network training on large data sets, a data matrix compression algorithm based on cosine transform was introduced, greatly reducing the computation time, reducing the memory requirement, and improving the computation efficiency. The trained neural network was used for the verification of synthetic and real-world data, and satisfactory results were obtained, indicating that the trained artificial neural network has good generalization ability and can be a good method for three-dimensional gravity interface inversion.

The experimental results from theoretical models and actual data demonstrate that the density interface inversion method based on the U-net network, which we have established in this paper, has superior inversion capabilities for deep and gentle density interface undulations. For density interface inversion problems characterized by steep interface undulations, further methodological design is required in aspects such as network structure design and the establishment of the sample set. This will pave the way for an intelligent and universal method for the computation of density interfaces.

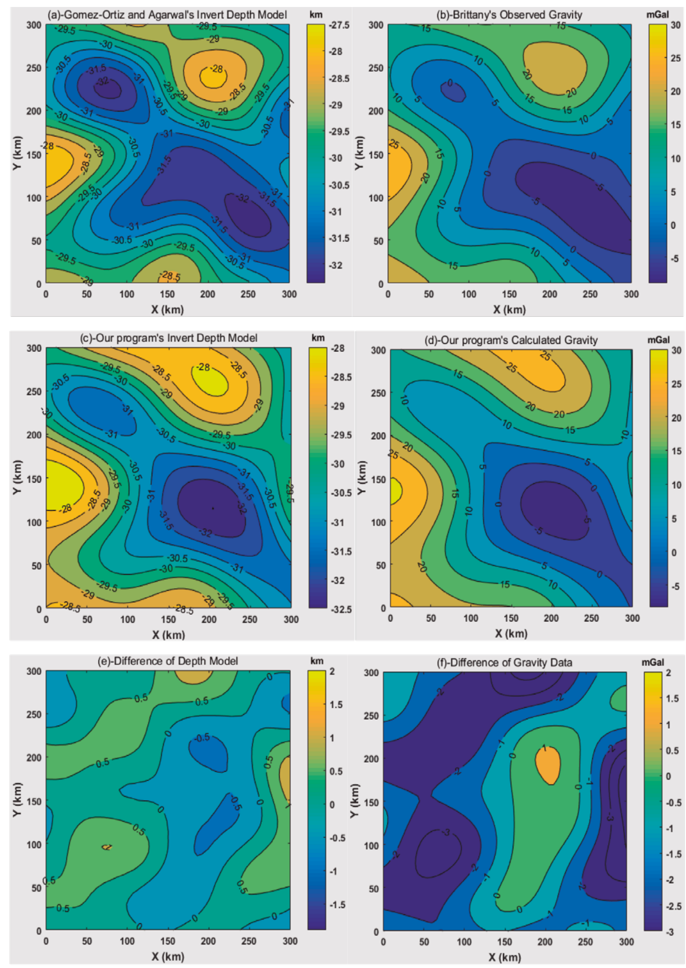

Figure 5. 3D Moho relief in Brittany (France)
(a)-Gomez-Ortiz and Agarwal's invert depth model (b)-Brittany's observed gravity data. (c)- our program's invert depth model (d)-our program's calculated gravity data (e)-difference of depth model (f)-difference of gravity data

References

Adler, J., & Öktem, O. (2017). Solving ill-posed inverse problems using iterative deep neural networks. *Inverse problems*. https://doi.org/10.1088/1361-6420/aa9581

Ashida, Y. (1996). Data processing of reflection seismic data by use of neural network. *Journal of Applied Geophysics*, *35*(2-3), 89-98. https://doi.org/10.1016/0926-9851(96)00010-9

Bott, M.H.P. (1960). The use of rapid digital computing methods for direct gravity interpretation of sedimentary basins. *Geophysical Journal of the Royal Astronomical Society*, *3*, 63-67. https://doi.org/10.1111/j.1365-246X.1960.tb00065.x

Carpenter, C. (2021). Machine-learning method determines salt structures from gravity data. *Journal of Petroleum Technology,* 73(02), 70–71. https://doi.org/10.2118/0221-0070-JPT

Cordell, L., & Henderson, R. G. (1968). Iterative three-dimensional solution of gravity anomaly data using a digital computer. *Geophysics,* 33, 596-601. https://doi.org/10.1190/1.1439955

Dai, H., & MacBeth, C. (1995). Automatic picking of seismic arrivals in local earthquake data using an artificial neural network. *Geophysical Journal International*, *120*(3), 758-774. https://doi.org/10.1111/j.1365-246X.1995.tb01851.x

Gómez-Ortiz, D., & Agarwal, B. N. (2005). 3DINVER.M: A MATLAB program to invert the gravity anomaly over a 3D horizontal density interface by Parker–Oldenburg's algorithm. *Computers & Geosciences, 31*(4), 513-520. https://doi.org/10.1016/j.cageo.2004.11.004

Granser, H. (1987a). Three-dimensional interpretation of gravity data from sedimentary basins using an exponential density-depth function. *Geophysical Prospecting,* 35, 1030-1041. https://doi.org/10.1111/j.1365-2478.1987.tb00858.x

Granser, H. (1987b). Nonlinear inversion of gravity data using the Schmidt-Lichtenstein approach. *Geophysics,* 52, 88-93. https://doi.org/10.1190/1.1442243

Jin, K. H., McCann, M. T., Froustey, E., & Unser, M. (2017). Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9), 4509–4522. DOI: 10.1109/TIP.2017.2713099

Lelièvre Peter, G., Farquharson Colin, G., & Hurich Charles, A. (2012). Joint inversion of seismic traveltimes and gravity data on unstructured grids with application to mineral exploration. *Geophysics,* 77(1), K1-K15. https://doi.org/10.1190/geo2011-0154.1

Li, Y., Jia, Z., & Lu, W. (2022). Self-supervised deep learning for 3D gravity inversion. *IEEE Transactions on Geoscience and Remote Sensing,* 60, 1-1. DOI: 10.1109/TGRS.2022.3225449

Ma, G., Niu, R., Gao, T., Li, L., Wang, T., & Meng, Q. (2022). High-efficiency gravity data inversion method based on locally adaptive unstructured meshing. *IEEE Transactions on Geoscience and Remote Sensing*, 60. DOI: 10.1109/TGRS.2022.3142042

Nagendra, R., Prasad, P., & Bhimasankaram, V. (1996). Forward and inverse computer modeling of a gravity field resulting from a density interface using Parker-Oldenberg method. Computers & Geosciences, 22(3), 227-237. https://doi.org/10.1016/0098-3004(95)00075-5

Oldenburg, D. W. (1974). The inversion and interpretation of gravity anomalies. *Geophysics,* 39(4), 526-536. https://doi.org/10.1190/1.1440444

Parker, R. L. (1972). The rapid calculation of potential anomalies. *Geophysical Journal of the Royal Astronomical Society,* 31, 447-455. https://doi.org/10.1111/j.1365-246X.1973.tb06513.x

Poulton, M., & El-Fouly, A. (1991). Preprocessing GPR signatures for cascading neural network classification. *61st SEG meeting, Houston, USA, Expanded Abstracts,* 507–509. https://doi.org/10.1190/1.1888789

Poulton, M. M., Sternberg, B. K., & Glass, C. E. (1992). Neural network pattern recognition of subsurface EM images. *Journal of Applied Geophysics*, *29*(1), 21-36. https://doi.org/10.1016/0926-9851(92)90010-I

Raiche, A. (1991). A pattern recognition approach to geophysical inversion using neural networks. *Geophysical Journal International,* 105, 629–648. https://doi.org/10.1111/j.1365-246X.1991.tb00801.x

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. 18th International Conference on Medical Image Computing and Computer-Assisted Intervention. Munich, Germany: Springer, 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

Silva, J. B. C., Santos, D. F., & Gomes, K. P. (2014). Fast gravity inversion of basement relief. *Geophysics,* 79(5), 79-91. https://doi.org/10.1190/geo2014-0024.1

Sun S., Yin C., & Gao X. (2021). 3D Gravity inversion on unstructured grids. *Applied Sciences,* 11(2), 722. https://doi.org/10.3390/app11020722

Van Der Baan, M., & Jutten, C. (2000). Neural networks in geophysical applications. *Geophysics*, 65(4), 1032-1047. https://doi.org/10.1190/1.1444797

Wang, Y. F., Zhang, Y. J., Fu, L. H., & Li, H. W. (2021). Three-dimensional gravity inversion based on 3D U-Net++. *Applied Geophysics*, 18(4), 451–460. https://doi.org/10.1007/s11770-021-0909-z

Wiener, J., Rogers, J. A., Rogers, J. R., & Moll, R. (1991). Predicting carbonate permeabilities from wireline logs using a back-propagation neural network. *61st SEG meeting, Houston, USA, Expanded Abstracts,* 285-288. https://doi.org/10.1190/1.1888943

Winkler, E. (1994). *Inversion of electromagnetic data using neural networks.* 56th EAEG meeting, Vienna, Austria, Extended Abstracts, P124.

Zhang, S., Yin, C., Cao, X., Sun, S., Liu, Y., & Ren, X. (2022). DecNet: Decomposition network for 3D gravity inversion. *Geophysics,* 87(5): 1SO-V558. https://doi.org/10.1190/geo2021-0744.1