

Article

Development of a Deep Neural Network Model for the Relocation of Mining-Induced Seismic Event

Chenlu Wang and Luyi Shen * 

Faculty of Engineering, China University of Geosciences, Wuhan 430074, China; 923699959@cug.edu.cn

* Correspondence: shenluyi@cug.edu.cn

Abstract: The precise relocation of seismic events is critical for many engineering projects. Swarms of minor or micro earthquakes typically reveal stress concentration and spots of greater seismic hazards. Particularly in the context of deep underground mining, advanced techniques that can accurately relocate microseismicities are urgently in demand. Here, we developed a neural network-based modeling training method that can precisely relocate seismicities and invert for velocities at the same time, with preconfigured receiver network locations. Our model can be iteratively improved with field recorded data. We showed that, with roughly eight iterations, we can reasonably resolve for the earthquake locations for both clusters of events, namely spatially distributed with linear pattern or randomly scattered. Our initially trained model, which only focused on events that had a linear distribution pattern, was used as the base for the training of the subsequent models which could better resolve for randomly scattered event locations. Although we stopped at the eighth iteration, the process reported here can be continued, as the model will have a better performance with more iterations.

Keywords: microseismicity; source relocation; deep learning; joint inversion



Citation: Wang, C.; Shen, L. Development of a Deep Neural Network Model for the Relocation of Mining-Induced Seismic Event. *Appl. Sci.* **2024**, *14*, 6911. <https://doi.org/10.3390/app14166911>

Academic Editor: Andrea Luca Rizzo

Received: 2 July 2024

Revised: 1 August 2024

Accepted: 1 August 2024

Published: 7 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Inversion for the precise location of micro or minor earthquakes holds significant importance, particularly in the contexts of hydraulic fracturing [1,2], mining operations [3–5], and other activities associated with microseismic events, which include dam and water reservoirs [6], geothermal resource exploitation [7], and landslide monitoring [8,9]. Such endeavors require a detailed understanding of the subterranean movements that can lead to these small-scale seismic events. As industries that engage in underground resource extraction (e.g., enhanced geothermal resource development, deep underground mining, etc.) and excavation engineering activities (e.g., urban underground space development, large hydro plants, etc.) continue to expand, the ability to accurately map and understand these seismic occurrences becomes paramount. This is not only crucial for the operational efficiency and safety of such industries but also plays a vital role in mitigating potential environmental impacts [10]. Furthermore, the importance of being able to precisely determine the locations of earthquakes is not limited only to the related industries and underground works, but it is also significant in the common structures used for normal life activities; natural earthquake and geohazard mitigation at the local scale can also benefit from accurately determined active or passive seismic sources [11,12].

Earthquake relocation techniques that allow us to pinpoint the origins of seismic activity with greater accuracy is therefore a key tool in the geophysical toolkit [13]. By investigating seismic source relocation and inversion techniques, particularly with the machine learning techniques that have seen great success on many aspects of geophysical inversion [13–15], we enhance our ability to monitor, predict, and respond to the geophysical changes (e.g., subsurface deformation, fault activation, ground water intrusion, etc.) induced by human activities and natural processes [16] alike.

1.1. Engineering Context

The relocation of deep mining-induced earthquakes, particularly those associated with coal mining operations, represent a critical area of study within the field of geophysics, and investigations into such a topic are currently in urgent demand due to the frequency of mining-related seismic disasters [17–20]. Also known as rock bursts [21], mining-induced seismicity is a phenomenon frequently observed in deep coal-mining activities, where the sudden movement of rock masses that resembles earthquake events occur when the accumulated stress within the rock layers is suddenly released, often triggered by the excavation processes inherent to mining operations [22]. Unlike the relatively predictable microseismicity induced by hydraulic fracturing, rock bursts are characterized by their sudden onset and potentially catastrophic impacts on mine safety and operations [23].

Hydraulic fracturing deliberately induces microseismicity to fracture rock formations and facilitate the extraction of oil or gas. In this context, the induced seismicity is a direct consequence of fluid injection under high pressure, leading to predictable and manageable microseismic events. These microseismic events are generally of low magnitude, and their monitoring helps in optimizing the fracturing process and assessing its effectiveness [24].

In contrast to hydraulic fracturing, where induced microseismic events are often a controlled and monitored aspect of the extraction process, the seismicity associated with coal mining, especially in the form of rock bursts, poses significant challenges for the safety of mining operations [22], particularly when the mining depth goes into the deep subsurface. The seismicity induced by deep coal-mining operations, particularly through the mechanism of rock bursts, is inherently linked to the removal of material from the Earth, leading to an unbalancing of the stress distribution in the surrounding rock [23]. The abrupt release of this stress can result in violent and unpredictable seismic events, posing significant risks to mine infrastructure and personnel. The complex interplays among geological conditions, mining depth, and the method of coal extraction adds layers of complexity to predicting and mitigating rock bursts [21]. This unpredictability and potential for substantial impact differentiate rock burst phenomena from the more controlled seismic events associated with hydraulic fracturing.

The relocation and detailed analysis of seismic events caused by deep mining activities, especially rock bursts, are thus crucial for developing strategies to mitigate these risks [25,26]. By developing seismic inversion techniques that is applicable for deep subsurface mining, researchers can gain a better understanding of the stress conditions and geological factors contributing to these events [27,28]. This knowledge not only aids in improving the safety and efficiency of mining operations but also contributes to our broader understanding of induced seismicity and its various manifestations across different industrial activities.

1.2. Seismic Source Inversion Techniques

Relocation of microseismicities rely on the inversion of seismic signals. This process essentially uses recordings from geophysical acoustic wave transducers that measure the motion of ground caused by the proxy seismic events as input. The arrival times of the waves, the shape and amplitudes of the waveforms, and the energy and frequency spectra all contain information on the sources of the event and properties of the geological materials between the sources and the recorders. Thus, we can invert these measurements for the source locations and the surrounding geological materials' physical properties [29]. Inversion techniques can be broadly classified into linear inversion and nonlinear inversion [30].

Linear inversion relies on pre-calculated ray paths, assuming a simplistic model where the changes in the seismic waves' travel times linearly correlate with changes in the subsurface properties [31]. This method is efficient and straightforward, suitable for scenarios with relatively homogenous geological structures. However, its simplicity also limits its applicability to more complex subsurface conditions. Nonlinear inversion goes beyond the assumptions of linear models by allowing for a more intricate relationship between the seismic data and the subsurface properties [32]. This approach does not assume

a direct, linear correlation between the changes in wave travel times and the subsurface structure, accommodating more complex geological scenarios. However, this increased accuracy comes at the cost of higher computational demands and the need for initial guess models that are close to the true subsurface conditions to avoid local minima in the solution space [33,34].

Several issues pose significant challenges to the effectiveness of seismic inversion techniques. Particularly, varying and unknown velocity structures are of concern; the Earth's subsurface is rarely uniform, featuring layers and anomalies with differing velocities. These variations can significantly affect seismic wave propagation, making it difficult to accurately model and predict seismic events. Seismic waves do not always follow straight paths. Their trajectories can be highly nonlinear, especially in heterogeneous media, complicating the modeling of their travel times and the inversion process.

The joint inversion of seismic velocities and source parameters offers a comprehensive approach to understanding subsurface properties and seismic event characteristics [35,36]. However, this method is notoriously slow and challenging to converge. The complexity arises from the need to simultaneously solve for both the medium's velocity structure and the seismic sources' parameters, a process that significantly increases the computational load and the solution space's dimensionality. The interdependence of velocities and source parameters means that inaccuracies in one can lead to errors in the other, making the process iterative and time-consuming. Achieving convergence requires not only significant computational resources but also clever algorithms and strategies to navigate the vast and complex solution space effectively. Despite these challenges, when successful, joint inversion can provide unparalleled insights into the Earth's subsurface and the dynamics of seismic events.

The integration of machine learning (ML) techniques into seismic source relocation and inversion represents a cutting-edge advancement in geophysical sciences, offering new avenues to overcome traditional challenges associated with these processes. Machine learning, with its ability to handle large datasets and uncover complex patterns within them, is well-suited to enhancing the accuracy and efficiency of seismic analyses.

Machine learning models, especially those based on deep learning, have shown remarkable success in improving the precision of seismic source relocation [37]. By training on vast arrays of seismic data, these models can learn the intricate relationships between seismic waveforms and their sources' locations. This capability allows for more accurate pinpointing of seismic events, even in areas with complex subsurface structures where traditional methods struggle. Moreover, ML models can significantly accelerate the relocation process, handling data volumes and complexities that would be impractical for conventional approaches. In seismic inversion, machine learning can address several of the longstanding difficulties, such as dealing with non-linearities in the data, unknown velocity structures, and the effects of seismic anisotropy. Neural networks, for example, can approximate the non-linear relationship between seismic data and subsurface properties without explicitly solving the complex physics-based equations. This approach can lead to faster convergence times and more accurate subsurface models, even in the presence of noisy or incomplete data. One of the most promising aspects of using ML in seismic inversion and relocation is its potential to streamline joint inversion processes. By efficiently integrating information from different seismic sources and accounting for the interplay between velocity structures and source mechanics, ML algorithms can facilitate more coherent and converged solutions, albeit with the caveat of requiring substantial computational resources and well-curated training datasets.

1.3. Objective and Scope of This Study

In the context of relocating sources of mining-induced earthquakes, this study aims to evaluate the efficacy of using deep neural networks (DNN) for the joint inversion of earthquake sources. Mining-induced seismicity, also known as rock bursts, constitutes the primary hazard impacting mining safety. Consequently, there is a pressing need for

the precise relocation of mining-induced seismic events. The process of relocating mining-induced seismicity is fundamentally analogous to the relocation of seismicity induced by hydraulic fracturing. Inspired by recent advancements in large artificial intelligence models, we aim to investigate whether a sufficiently large DNN model can effectively address the earthquake relocation problem by simultaneously inverting for both velocity and source location.

2. Data and Method

Here, we created a scenario to replicate real mining sites in Dongtan, China, by deploying a network of 27 seismometer stations (Figure 1a). Each station can detect seismic waves originating from the earthquake source (Figure 1b). These seismometers are designed to monitor earthquakes occurring at depths of approximately 1000 to 2500 m within the network (Figure 1c).

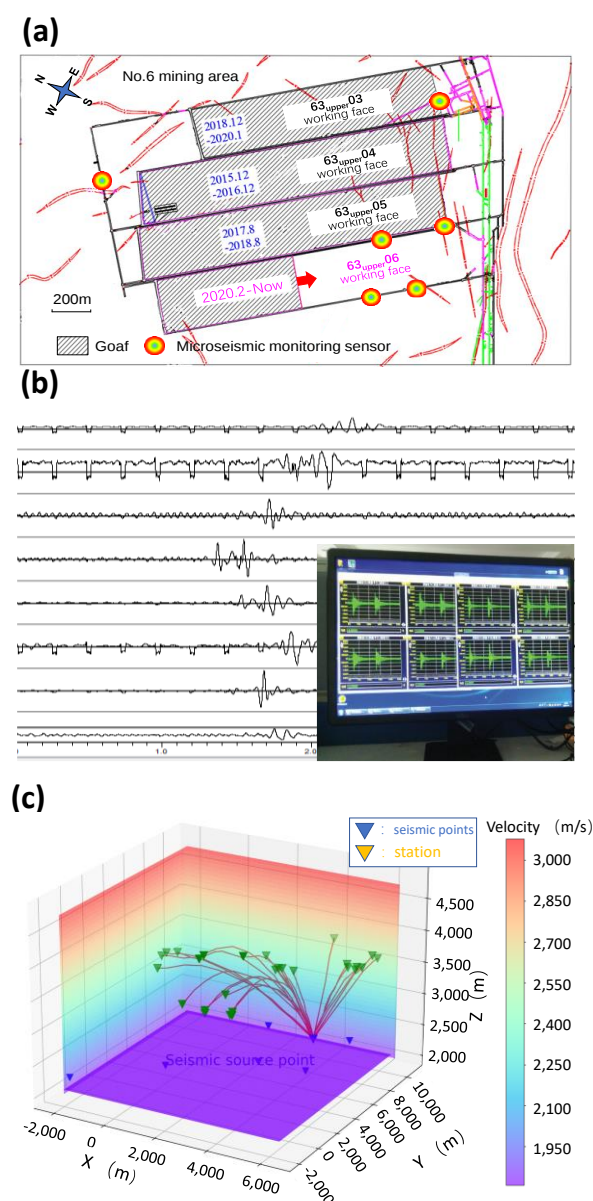


Figure 1. (a) Working face and seismometers in the No.6 mining area of DongTan coal mine, adapted from Wu et al. [38]; (b) raw waveform data recorded by seismometers; (c) station–ray–source coordinate diagram. Green represents stations, red represents rays, and blue triangles represent the seismic sources, with background colors indicating varying velocities.

The objective of this study was to train a model using synthetic data that could predict wave velocity and earthquake locations based on the P-wave arrival times at each station. Our methodology consisted of three phases, namely data generation, model training, and model testing. Uniquely, our approach did not follow a strict sequential order for these phases. We began by generating initial data with an assumed velocity model, used it for training, and subsequently generated more data with different velocity models for further training.

The core of our research lies in training a model that can be iteratively improved with the influx of newly acquired field recording data, regardless of its current state of training. Such recorded data can be continuously acquired through seismometers deployed near the mining site.

Each iteration may employ different methodologies; however, if the methods are scientifically sound, the subsequent model would invariably surpass its predecessor in performance. Thus, we proposed a novel approach to solving the problem of seismic source inversion—continually training models through neural networks, interspersed with various methods.

Each iteration of our model represented a step forward, with the process of moving to the next step being independent of the previous one. Essentially, our training improved the model incrementally, and this improvement was achieved using different, relatively independent methods at each step. Consequently, the model could be refined to a very high degree of accuracy through unlimited iterations. Each iterative model remained independent of the others, and the methods employed in each step was different and independent. Therefore, it was feasible to develop a modular iterative inversion model training approach.

This iterative process allowed us to incorporate the acquired velocity knowledge to enhance the model while also reducing the computational cost associated with training. It ultimately yielded a high-precision model suited to a specific scenario—in this case, mining-induced seismicity—although the methodology is also applicable to hydropower, bridges, and tunnels.

2.1. Data Used for Model Training and Testing

2.1.1. True Synthetic Data and Regular Training Data

Our data are categorized into two types—the “true” data and the training data. Though both data are, in principle, synthetic data and their calculation methods are the same (see text below), an important distinction must be stressed here. The ordinary training data were obtained by calculating the initial arrival times at stations. These data are paired with their corresponding velocity models and used as the initial dataset for the first iteration of the training of the inversion models. The true data, on the other hand, were a series of arrival times at receiver stations calculated using different velocity models. Considering that in real engineering environments, we do not have access to the velocity models required for training models, it is not possible to directly associate true data with any specific training velocity models or seismic source data.

During model training, we used true data as input to calculate the corresponding velocity model seismic sources (as described in the following sections), thereby forming a complete training dataset.

2.1.2. Synthetic Data Generations

Our synthetic data, that include both training data and true data, were constructed through forward modeling with our randomly generated velocity models. In generating these models, we adhered to fundamental geological principles. Firstly, P-wave velocities typically ranged from 1500 m/s to 3200 m/s, a range established by the site engineers’ past experiences. Secondly, velocities generally increased with greater depth.

To create our velocity models, we started with an initial vertical velocity profile model and introduced random variations. The model was divided into 30 depth layers, covering

a depth range of 3000 m and each layer was assigned with a constant velocity initially. Vertically, the velocities ranged from 2000 m/s to 3000 m/s, increasing linearly from the bottom layer to the top. For each layer, we added a random variation of 50 m/s to simulate natural heterogeneity (Figure 2a).

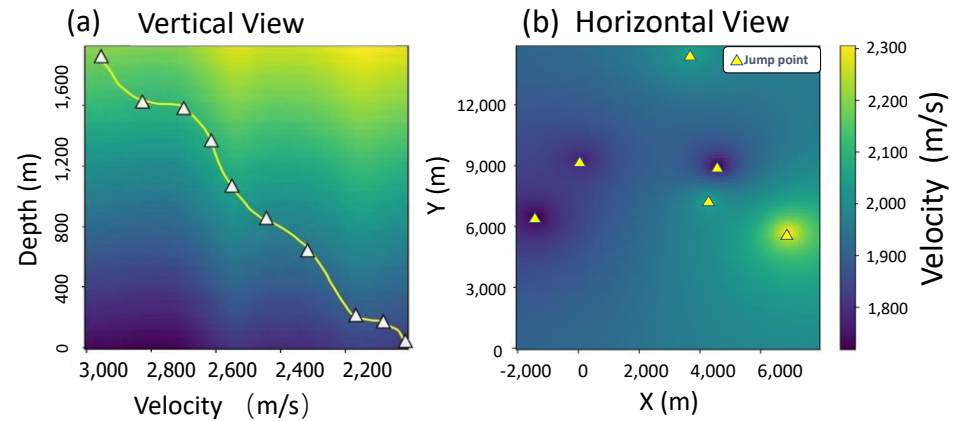


Figure 2. (a) Vertical velocity profile and vertical direction line graph on the x-z plane at Y = 0; (b) Horizontal velocity profile on the x-y plane at Z = 0 (triangles denote points of abrupt velocity change).

Horizontally, the model spans 8000 by 12,000 m, structured as a 30 by 30 grid. We employed two methods to introduce variance into a plane with constant velocity, which was calculated based on the previously discussed vertical velocity profile. Both methods start with this constant velocity plane that is defined by the vertical velocity profile discussed in the aforementioned text. For the first method, we used the following equation to add variance to our velocities on the horizontal plane:

$$V(x,y) = a_1 \sin(a_2 x) + a_3 \cos(a_4 y) + a_5$$

where x and y are the horizontal coordinates, $a_i (i = 1-5)$ are random variables. We either start with $a_1 = 60$, $a_2 = 0.25$, $a_3 = 50$, $a_4 = 0.15$, $a_5 = 0.20$, or we randomly select six points and add random variables to their velocities; for the remaining grid points on the horizontal plane, we calculate velocities using inverse distance weighting. Both methods use 180 parameters (30 vertical layers and each layer has 6 parameters) to characterize our velocity model.

These methods ensure that our forward modeling process incorporates realistic horizontal velocity variations, enhancing the accuracy and applicability of our synthetic data for training seismic inversion models. This approach ensures that the forward modeling process realistically reflects the geological characteristics of the study area, thereby enhancing the reliability and applicability of the synthetic data for training the seismic inversion models.

With the generated velocity models, we hypothetically placed earthquake sources within the model and then used ray-tracing methods to calculate the P-wave arrival times at each receiver. These arrival times and earthquake locations were subsequently used as the training data.

For the models, the event locations were randomized. For each model, 4000 events were generated. We employed two methods. First, events happening in locations that were spatially continuous; the coordinates of the first earthquake were (X, Y, Z) . For the second earthquake, we adjusted the coordinates by adding random values to X , Y , and Z (i.e., $X + \text{random}(-0.5, 0.5)$). The same approach was also applied to the Y and Z coordinates. This method creates a series of events that are roughly connected, simulating real-world scenarios where earthquakes predominantly occur along the same fault plane. We hereby denote this as event generation methods I. Or, when we have entirely randomly scattered events, we denote this as event generation methods II.

Ray tracing is utilized in both linear and nonlinear inversion methods to model the paths taken by seismic waves as they travel through the Earth. Ray-tracing algorithms simulate how these paths bend and twist in response to varying subsurface properties, which is essential for accurately predicting the travel times and amplitudes of seismic waves. This technique is crucial for understanding wave propagation in complex geological settings, including those with varying velocity structures and anisotropies.

The ray tracing is performed using the `tccrpy` package, which is designed for computing travel times and ray tracing with geophysical applications in mind. This package includes code for computations on 2D and 3D rectilinear grids, as well as 2D triangular and 3D tetrahedral meshes. The following three different algorithms are implemented in `tccrpy`: the Fast-Sweeping Method, the Shortest-Path Method, and the Dynamic Shortest-Path Method. For this study, we used the Shortest-Path Method (SPM).

3. Model Training

The goal was to leverage deep learning techniques to accurately predict earthquake parameters based on seismic arrival times and geological features inferred from velocity models. This implementation used PyTorch 2.4 with CUDA support for GPU acceleration, designed to harness the computational power of GPUs through CUDA, enabling faster training and optimization of the model parameters.

We imported the `torch.nn` module from Python 3.12 and built a four-layer neural network model (Figure 3). The first layer was the input layer with 27 neurons, corresponding to the 27 processed arrival times. The activation function used was ReLU. The second layer was a hidden layer with 64 neurons, and the third layer was another hidden layer with 64 neurons, also using the ReLU activation function. The fourth layer was the output layer with 34 neurons, including the source coordinates (X, Y, Z), earthquake time t_{ts} , and velocity model parameters. The activation function for the output layer was also ReLU. We note here, for inversion, we only inverted for a linear depth velocity profile—a horizontal variation of velocities is not accounted for here.

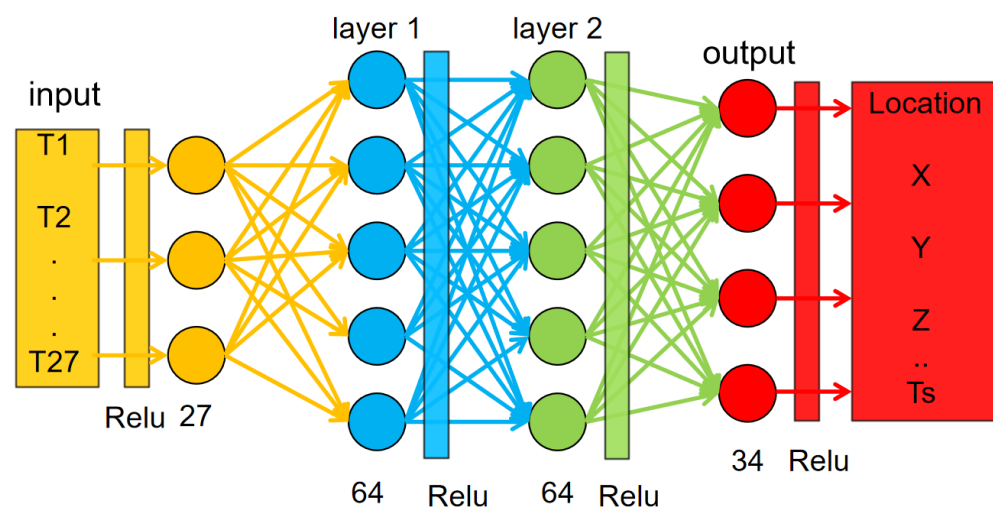


Figure 3. Neural network architecture used for training in this study.

Next, we imported the `train_test_split` function from the `sklearn.model_selection` module to divide the data into training and test sets using an 80:20 ratio. This means that 80% of the data were allocated to the training set, which was used to train the neural network model. The remaining 20% formed the test set, which served to evaluate the model's performance and assess its generalization capabilities on unseen data. We used `TensorDataset` and `DataLoader` from PyTorch to create datasets and data loaders, with a batch size of 128.

For optimization, we employed the Adam optimizer and utilized the Mean Squared Error (MSELoss) function as our loss metric. The learning rate dynamically ranged from 0.001 to 0.0001 to facilitate efficient convergence during training. Each model is trained for 10,000 iterations and, during each iteration, data undergo one forward and one backward propagation pass, with the loss function's gradient descent optimizing the neural network model parameters. After the iterations are completed, the model is saved for subsequent inversion capability testing and further iterative upgrades.

It is important to note that the test data used here are different from the 'true' data that will be used later to validate and assess the performance of our model. The test data here include both arrival times and a velocity model, whereas the 'true' data only have the arrival times without a velocity model.

Using a ray-tracing model, we obtained 27 source arrival times (t_1 – t_{27}) that count as one set of our training data. We define the minimum value of these 27 arrival times as t_s . We subtract t_s from the arrival times at each station and use these 27 adjusted times as the input data X (Figure 4) for the neural network. The output data Y of the neural network model includes the earthquake source coordinates (X, Y, Z), the earthquake time t_s , and six velocity parameters for each layer (a total of 180 parameters).

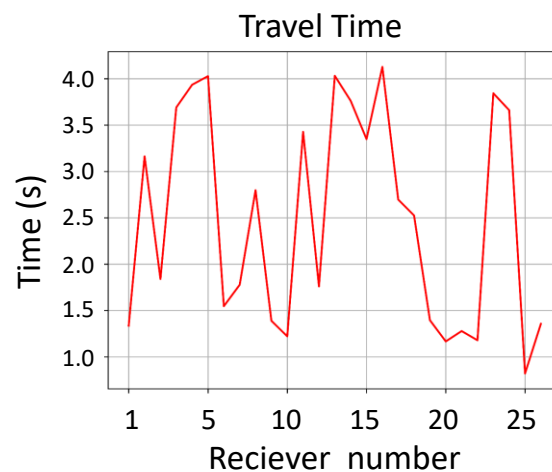


Figure 4. Initial arrival times for the 27 stations (after subtracting the minimum value).

We use an iterative training process to train our model with the inclusion of synthetically generated data that are supposed to represent real field data (Figure 5). We first construct our initial training model by inputting 8000 sets of seismic events. This is in accordance with the past engineering experiences that roughly 5000–10,000 events would be recorded per year. Each set in these 8000 data sets has source coordinates that follow a linear pattern, as previously described. These 8000 sets are divided into four subgroups, with each subgroup consisting of 2000 data sets generated from the coordinates (4000, 10, 100, 690). Each subgroup employs a different velocity model, with each model described by 180 different parameters (as detailed in the earlier texts). This constitutes the first dataset (Dataset I), which includes a total of 8000 seismic events and four different velocity models.

The dataset is divided into a training set and a test set using an 80:20 ratio. This means that 80% of the data are allocated to the training set, which is used to train the neural network model. The remaining 20% of the data form the test set, which serves to evaluate the model's performance and assess its generalization capabilities on unseen data. This partitioning strategy helps in assessing the model's effectiveness in predicting earthquake source parameters and subsurface velocities accurately, while also validating its robustness and reliability.

Using training dataset I, we conducted neural network training (refer to the previous sections). After training, we obtained model I. We then applied model I to invert 2000 sets of true data, resulting in a new velocity model. These 2000 sets of true data, along with the

velocity model obtained from the inversion and the source coordinates derived based on this velocity model and true data, were packaged and converted into training data.

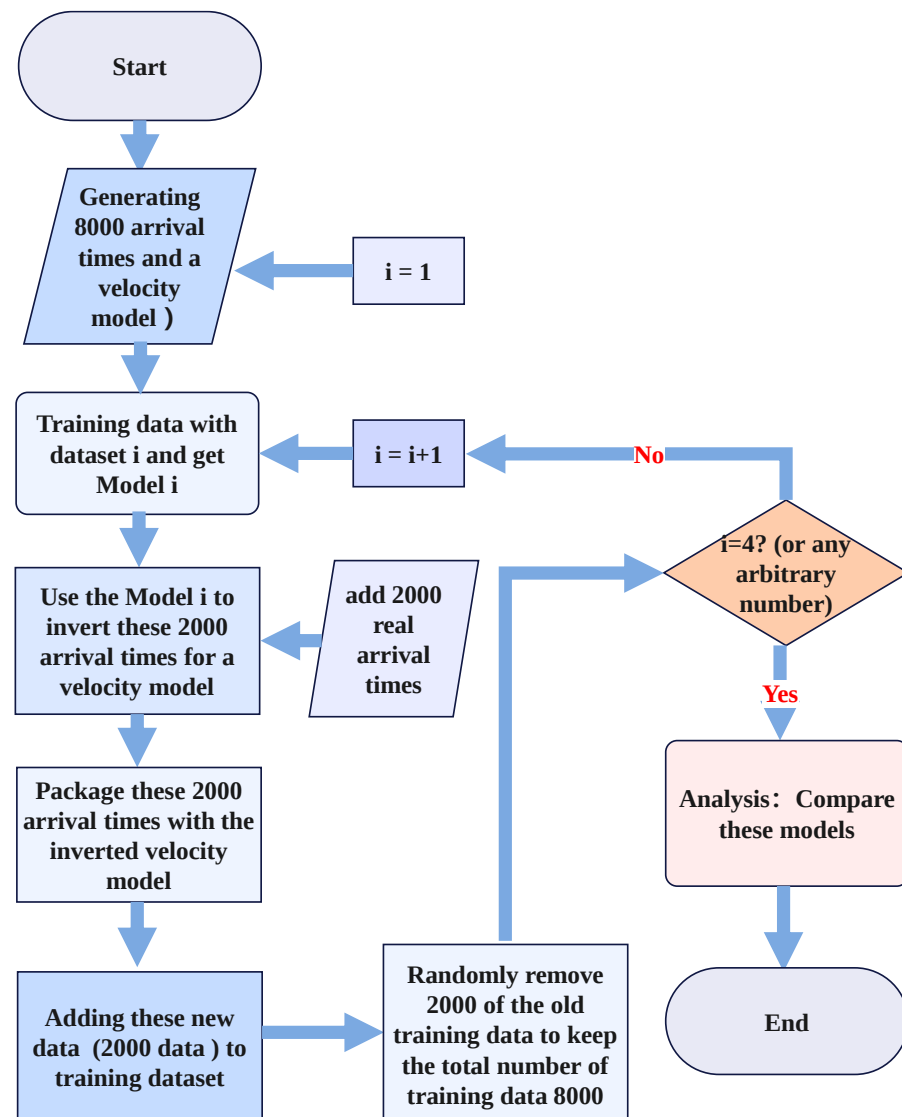


Figure 5. Flowchart for the first 4 iterations. The later iterations can continue with $i > 4$.

It is important to emphasize that the source coordinates in our package were obtained through inversion using the velocity model and initial arrival times, not the actual source coordinates. In practical engineering scenarios, we do not know the exact source locations; we only have the initial arrival times at the stations. At this stage, since we have the velocity model (predicted by model I) and the initial arrival times ('true' data), we can also calculate the source coordinates using conventional algorithms. Here, we employed a standard genetic algorithm, which we will not elaborate on further. The true initial arrival times, the velocity model derived from model I, and the source locations obtained from the conventional algorithm form our new 2000 sets of training data.

We then added these 2000 sets of data to dataset I, resulting in dataset II, which contains a total of 10,000 datasets. This completes one cycle of our training process. Subsequently, we began training again, using dataset II to obtain model II. We then added a new batch of training data (2000 sets) composed of true data, the velocity model derived from model II, and the source coordinates obtained using the velocity model and conventional algorithms. This process is theoretically repeatable as long as we have sufficient true data. In practical engineering, true data will continue to accumulate over time.

In this study, during iterations I–IV, we employed a linear method to construct the initial model and true data. This approach was designed to enhance our model’s ability to infer the relative positions of seismic sources. Starting from iteration V to VIII, we used completely random seismic sources (within the ranges X: 4000–7313 m, Y: 10,100–13,411 m, Z: 690–1352 m) to obtain the initial arrival times as true data and to calculate the additive training dataset. The DNN neural network architecture remained consistent throughout the entire study.

4. Results and Discussion

4.1. Linearly Distributed Events

We have developed four models (designated as Model I to Model IV) capable of inversely predicting velocity models and the locations and times of earthquakes that are spatially distributed linearly (generated using event generation method I). Using an iterative training approach, we conducted tests to evaluate the efficacy of these models in earthquake relocation. Our evaluation involved generating new datasets using the same methods employed for training data generation, then utilizing our trained models to predict earthquake source relocations. The primary focus of this assessment was comparing the predicted source locations against the actual source locations to assess the performance of our DNN models. It is important to emphasize that the true data used in the testing phase is generated independently and is not related to the true data used during training.

Firstly, we examined linearly distributed earthquake locations to simulate scenarios where earthquakes occur along a linear faulting plane (Figure 6).

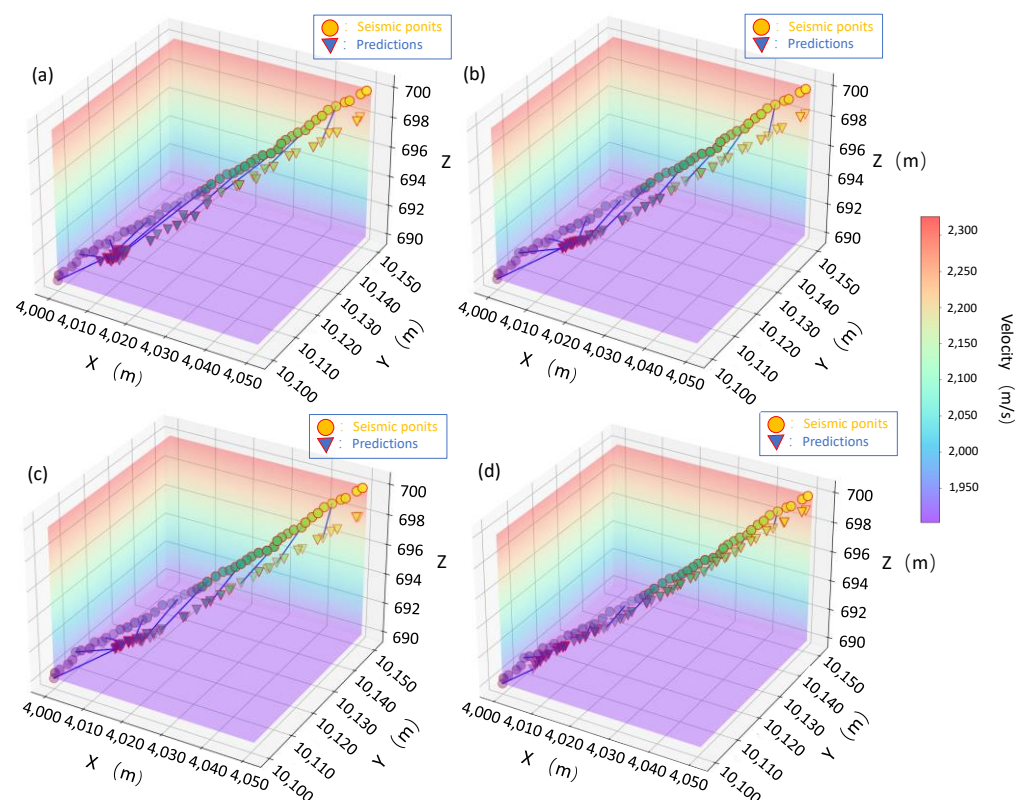


Figure 6. Performance test of the first four models with linear data. Panels (a–d) correspond to models I–IV, respectively, where the crosses represent the true data and triangles denote predicted coordinates.

We also performed analysis on the discrepancies between the actual source locations and our predicted locations of earthquakes, using models I–IV (Figure 7), and we found the following:

- Model I: Average distance error in earthquake source location is approximately 20 m. The trend in predicted locations for the first 50 events varied widely, showing a lack of linear trend in most cases. However, the vertical velocity model achieved a close fit of 99%.
- Model II: Average distance error in earthquake source location is around 18 m. There was some linear trend observed in the first 50 events, although the coordinates clustered together. Later events showed a clearer linear trend, with the vertical velocity model also achieving a 99% match.
- Model III: Average distance error in earthquake source location ranges from 12 to 16 m. A majority of the events exhibited a linear trend.
- Model IV: Distance errors mostly ranged between 5 to 10 m, indicating the closest proximity between predicted and actual earthquake coordinates.

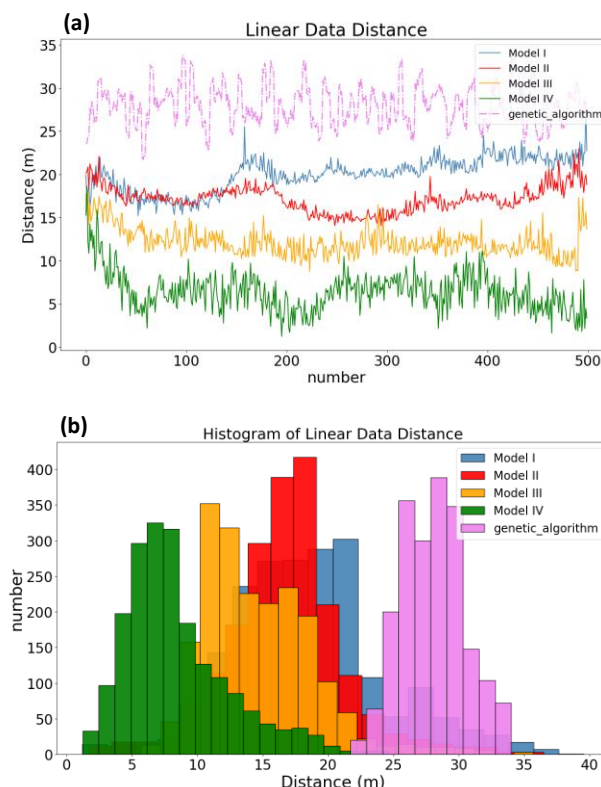


Figure 7. Performances of models I–IV. (a) Error in predicted coordinates compared to actual source coordinates, and (b) histogram showing the distribution of errors.

These findings highlight Model IV as the most accurate in predicting earthquake source locations, demonstrating significant improvement in localization precision compared to the other models.

Also, for the sake of comparison and assessing our model's performance against traditional relocation techniques, we made a comparison with the results obtained from a genetic algorithm. The genetic algorithm starts with a pre-assumed earthquake location at (x, y, z) and a velocity model similar to the one used for generation training data in this study. Each iteration of the genetic mutation allows the (x, y, z) and parameters that describe the velocity model (30 layers with no account for horizontal variation) to vary by a certain percentage (10% in this study). The seismic wave arrival times are calculated with these parameters and compared with station recorded arrival times. L2 norm is used as the metric to determine if this mutation yields better results; if the L2 norm is smaller than the previous iteration, these mutated parameters are kept and allowed to further mutate in the next iteration. The iterative process stops when the L2 norm stops improving/decreasing.

Figure 7 shows that the results produced by this traditional genetic algorithm is less accurate than our models, even the initial model I. Considering that our model can be further improved with more iterations and the input of more field ‘true’ data, the accuracy disparities between our model and the genetic algorithm will only be greater. That being said, we should still acknowledge that our model is particularly tailored for linearly distributed events—the linearity is somehow accounted for in our ‘black box’ DNN model. In the section below, we will continue to test our model, against its own variations trained in iterations V–VIII, and this genetic algorithm.

4.2. Scattered Events

Additionally, we conducted tests on our models using earthquake locations randomly scattered within a region defined by x (4000–5000), y (10,100–11,100), and z (650–900) (generated using event generation method II, with entirely randomly scattered earthquakes). This scenario simulates earthquake occurrences across a broader and more irregular spatial distribution, challenging the models’ ability to accurately predict source locations under diverse conditions (Figure 8).

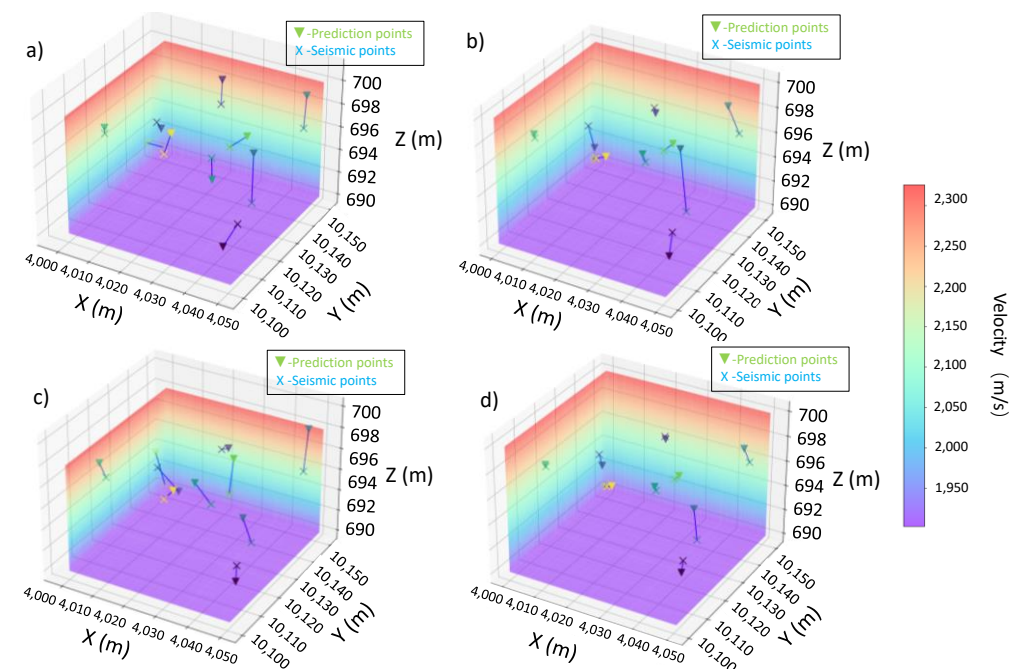


Figure 8. Performance test of the last four models with linear data. Panels (a–d) correspond to models V, VI, VII, and VIII, where crosses mark the true data and triangles mark the predicted coordinates.

We also performed analysis on the discrepancies, with model V–VIII, between the actual source locations and our predicted locations of earthquakes (Figure 9), and we found that Model V had a source distance error of 30–40 m, with most coordinates randomly distributed within a 360-degree range around the source. Model VI had a source distance error of 30–35 m, Model VII had a source distance error of 20–25 m, and Model VIII had a source distance error of 5–15 m, with predicted coordinates closely matching the actual source coordinates.

The significant improvement in Model VIII is due to the iterative upgrading process, during which the number of data points in the training dataset gradually increased. This enhanced the model’s ability to handle the randomly distributed data points within the entire seismic range beneath the stations. In other words, Model VIII contains more and more comprehensive scatter data, resulting in better data fitting and more accurate inversion of the true coordinates.

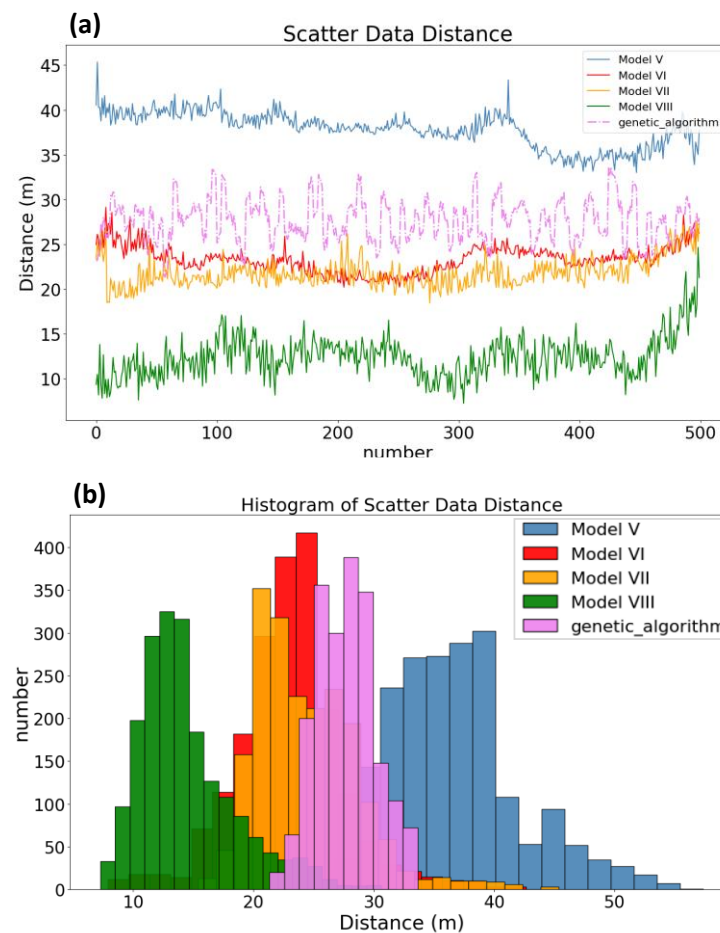


Figure 9. Performances of models V–VIII. (a) Error in predicted coordinates compared to actual source coordinates, and (b) histogram showing the distribution of errors.

This shows that the later input data can continue to improve the model and different data can provide different improvements. Scattered data input can improve the models' capability in relocating scattered events, and linear data can improve its performance in linearly distributed events.

Finally, our models were also compared with the results obtained through the genetic mutation algorithm (see details in section above). For the scattered events, our models' advantages against the genetic mutation algorithm were less significant and only iterations VII and VIII produced models surpassing the genetic mutation algorithm. Nevertheless, we can continue our training to make better models in the later iterations that can locate seismic events more accurately, even in the most generic application scenarios.

We would also like to point out that, although the training of our model takes a tremendous amount of time, even with our triple GPU (NVidia 4090 or equivalent) rig, our model takes a minimal amount of time to perform deductions. On the other hand, the genetic mutation algorithms, that require the constraining of 34 parameters (30 parameters for velocity model, 3 for event location, and 1 for the timing of the seismic event), can take more than 100 times more time (up to half hour or 30 min) than our ML-trained model (often in less than tens of seconds). This distinction is important to be noted as field calculation can often be strained by the computational resource, and our approach fits better with such a limitation.

4.3. Final Remarks

Through this iterative process, each subsequent model consistently showed incremental improvement over the previous one. Particularly with the linear data, the improvements

were clearly reflected in both error plots and histograms. Thus, this research proves effective, essentially conducting two types of experiments—comparing four models using linear data and four models using scattered data.

Our approach centers on training a model that, regardless of initial training conditions, progresses to the next iteration. Each iteration may employ different methods, but scientifically sound approaches consistently yield better results than the previous model. Therefore, we proposed the following methodology for solving earthquake source inversion problems: training a neural network model with varied methods interspersed throughout the process, ultimately achieving a high-precision model applicable to specific scenarios (such as mining-induced seismicity, with broader applicability to hydroelectric and infrastructure studies).

Each iteration of our model represents a step forward, with each step independent yet contributing to overall improvement. This training method is inherently geared towards enhancing model accuracy through successive iterations, leveraging the flexibility to apply diverse techniques at each step.

In practice, our approach can facilitate real-time velocity and source inversion. While training the model can be time-consuming, the actual deduction/prediction process is significantly faster. The model supports continuous training, allowing for seamless continuation at any point, even with different DNN parameters. We incorporate the true data, that can be accumulated during the real engineering and monitoring process, that helps improve the accuracy of our model's prediction. This iteration can continue to improve and eventually result in a highly capable earthquake source relocation model for the site.

This modular iterative inversion model training framework thus enables robust and adaptable seismic source analysis, essential for various geophysical applications.

5. Conclusions

In our study, we utilized Deep Neural Network (DNN) to train a model that is capable of predicting both velocity profiles and earthquake locations. We leveraged real deployed seismometer locations, along with expected and estimated earthquake locations, to generate a substantial volume of training data.

Our approach was iterative, allowing us to continuously refine the model using prior knowledge. By incrementally adding new data, we built upon the existing model rather than starting training anew. This iterative strategy not only saved time but also optimized computational resources.

Currently, our model demonstrates effective performance in predicting events that occur linearly in space, which is particularly relevant for real-world scenarios where earthquakes often cluster near fault lines. For events distributed in a completely scattered manner, uncertainties are higher initially. However, our iterative training process demonstrates potential for improving accuracy over time, especially if future efforts focus on refining predictions for scattered events. The computational efficiency of our model for prediction tasks is notably high, contributing to its practical applicability.

This study underscores the effectiveness of using DNNs in training models for microseismicity relocation. By incorporating prior geological knowledge and foundational principles of earthquake seismology, we streamline the training process, reducing both time and computational costs significantly.

Author Contributions: Software, C.W.; Formal analysis, C.W.; Data curation C.W.; Writing—original draft, L.S.; Writing—review & editing, L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by NSFC (grant # 42227805) and Fundamental Research Funds for the Central Universities, China University of Geosciences (Wuhan) (grant # CUG240619).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Maxwell, S.C.; Cipolla, C. What does microseismicity tell us about hydraulic fracturing? In Proceedings of the SPE Annual Technical Conference and Exhibition, Denver, CO, USA, 30 October–2 November 2011; p. SPE-146932-MS.
2. Shapiro, S.; Dinske, C.; Rothert, E. Hydraulic-fracturing controlled dynamics of microseismic clouds. *Geophys. Res. Lett.* **2006**, *33*. [CrossRef]
3. Eneva, M. In search for a relationship between induced microseismicity and larger events in mines. *Tectonophysics* **1998**, *289*, 91–104. [CrossRef]
4. Pasten, D.; Estay, R.; Comte, D.; Vallejos, J. Multifractal analysis in mining microseismicity and its application to seismic hazard in mine. *Int. J. Rock Mech. Min. Sci.* **2015**, *78*, 74–78. [CrossRef]
5. Young, R.; Maxwell, S.C.; Urbancic, T.; Feignier, B. Mining-induced microseismicity: Monitoring and applications of imaging and source mechanism techniques. *Pure Appl. Geophys.* **1992**, *139*, 697–719. [CrossRef]
6. Li, Z.; Zhou, L.; Duan, M.; Zhao, C. Deep Learning-Based Microseismic Detection and Location Reveal the Seismic Characteristics and Causes in the Xiluodu Reservoir, China. *Bull. Seismol. Soc. Am.* **2024**, *114*, 806–822. [CrossRef]
7. Piccinelli, F.; Mucciarelli, M.; Federici, P.; Albarello, D. The microseismic network of the Ridracoli Dam, North Italy: Data and interpretations. *Pure Appl. Geophys.* **1995**, *145*, 97–108. [CrossRef]
8. Lacroix, P.; Helmstetter, A. Location of seismic signals associated with microearthquakes and rockfalls on the Séchilienne landslide, French Alps. *Bull. Seismol. Soc. Am.* **2011**, *101*, 341–353. [CrossRef]
9. Yang, Y.; Chen, G.; Meng, X.; Bian, S.; Chong, Y.; Shi, W.; Jiang, W.; Jin, J.; Li, C.; Mu, X. Analysis of the microseismicity characteristics in landslide dam failure flume tests: Implications for early warning and dynamics inversion. *Landslides* **2022**, *19*, 789–808. [CrossRef]
10. Huang, L.-Q.; Li, X.-B.; Dong, L.-J.; Zhang, C.-X.; Dong, L. Relocation method of microseismic source in deep mines. *Trans. Nonferrous Met. Soc. China* **2016**, *26*, 2988–2996. [CrossRef]
11. Kustu, T.; Taskin, A. Deep learning and stereo vision based detection of post-earthquake fire geolocation for smart cities within the scope of disaster management: İstanbul case. *Int. J. Disaster Risk Reduct.* **2023**, *96*, 103906. [CrossRef]
12. Nicoletti, V.; Arezzo, D.; Carbonari, S.; Gara, F. Detection of infill wall damage due to earthquakes from vibration data. *Earthq. Eng. Struct. Dyn.* **2023**, *52*, 460–481. [CrossRef]
13. Cheng, J.; Song, G.; Sun, X.; Wen, L.; Li, F. Research developments and prospects on microseismic source location in mines. *Engineering* **2018**, *4*, 653–660. [CrossRef]
14. Qian, J.-W.; Anyiam, U.O.; Wang, K.-D. Machine learning-based microseismic catalog and passive seismic tomography evaluating the effect of grouting in Zhangji coal mine, China. *Appl. Geophys.* **2023**, *20*, 167–175. [CrossRef]
15. Shi, P.; Grigoli, F.; Lanza, F.; Beroza, G.C.; Scarabello, L.; Wiemer, S. MALMI: An automated earthquake detection and location workflow based on machine learning and waveform migration. *Seismol. Soc. Am.* **2022**, *93*, 2467–2483. [CrossRef]
16. Zhou, L.; Zhao, C.; Zhang, M.; Xu, L.; Cui, R.; Zhao, C.; Duan, M.; Luo, J. Machine-learning-based earthquake locations reveal the seismogenesis of the 2020 M w 5.0 Qiaojia, Yunnan earthquake. *Geophys. J. Int.* **2022**, *228*, 1637–1647. [CrossRef]
17. Li, T.; Cai, M.; Cai, M. A review of mining-induced seismicity in China. *Int. J. Rock Mech. Min. Sci.* **2007**, *44*, 1149–1171. [CrossRef]
18. Hasegawa, H.S.; Wetmiller, R.J.; Gendzwill, D.J. Induced seismicity in mines in Canada—An overview. *Pure Appl. Geophys.* **1989**, *129*, 423–453. [CrossRef]
19. Fritschen, R. Mining-induced seismicity in the Saarland, Germany. *Pure Appl. Geophys.* **2010**, *167*, 77–89. [CrossRef]
20. Bischoff, M.; Cete, A.; Fritschen, R.; Meier, T. Coal mining induced seismicity in the Ruhr area, Germany. *Pure Appl. Geophys.* **2010**, *167*, 63–75. [CrossRef]
21. Blake, W. Rock Burst Mechanics. 1970–1979-Mines Theses Diss. 1971. Available online: https://repository.mines.edu/bitstream/handle/11124/16965/Blake_10796009.pdf?sequence=1 (accessed on 1 July 2024).
22. Zhu, G.-a.; Dou, L.-m.; Li, Z.-l.; Cai, W.; Kong, Y.; Li, J. Mining-induced stress changes and rock burst control in a variable-thickness coal seam. *Arab. J. Geosci.* **2016**, *9*, 365. [CrossRef]
23. Chen, Y.; Zhang, J.; Chen, J.; Deng, X. Rock burst disasters in coal mines. *Energies* **2022**, *15*, 4846. [CrossRef]
24. Chen, B.; Barboza, B.R.; Sun, Y.; Bai, J.; Thomas, H.R.; Dutko, M.; Cottrell, M.; Li, C. A review of hydraulic fracturing simulation. *Arch. Comput. Methods Eng.* **2021**, *29*, 1–58. [CrossRef]
25. Wang, S.; Zhu, G.; Zhang, K.; Yang, L. Study on characteristics of mining earthquake in multicoal seam mining under thick and hard strata in high position. *Shock Vib.* **2021**, *2021*, 6675089. [CrossRef]
26. Zhang, M.; Hu, X.; Huang, H.; Chen, G.; Gao, S.; Liu, C.; Tian, L. Mechanism and prevention and control of mine earthquake in thick and hard rock strata considering the horizontal stress evolution of stope. *Shock Vib.* **2021**, *2021*, 6680928. [CrossRef]
27. Orlecka-Sikora, B.; Lasocki, S.; Lizurek, G.; Rudziński, Ł. Response of seismic activity in mines to the stress changes due to mining induced strong seismic events. *Int. J. Rock Mech. Min. Sci.* **2012**, *53*, 151–158. [CrossRef]

28. Chen, B. Stress-induced trend: The clustering feature of coal mine disasters and earthquakes in China. *Int. J. Coal Sci. Technol.* **2020**, *7*, 676–692. [[CrossRef](#)]
29. Russell, B.H. *Introduction to Seismic Inversion Methods*; SEG Books: Houston, TX, USA, 1988.
30. Wang, Y. *Seismic Inversion: Theory and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2016.
31. Yilmaz, Ö. *Seismic Data Analysis: Processing, Inversion, and Interpretation of Seismic Data*; Society of Exploration Geophysicists: Houston, TX, USA, 2001.
32. Spikin, S.A. Estimation of earthquake source parameters by the inversion of waveform data: Global seismicity, 1981–1983. *Bull. Seismol. Soc. Am.* **1986**, *76*, 1515–1541. [[CrossRef](#)]
33. Weston, J.; Ferreira, A.M.; Funning, G.J. Systematic comparisons of earthquake source models determined using InSAR and seismic data. *Tectonophysics* **2012**, *532*, 61–81. [[CrossRef](#)]
34. Duputel, Z.; Rivera, L.; Fukahata, Y.; Kanamori, H. Uncertainty estimations for seismic source inversions. *Geophys. J. Int.* **2012**, *190*, 1243–1256. [[CrossRef](#)]
35. Liang, C.; Yu, Y.; Yang, Y.; Kang, L.; Yin, C.; Wu, F. Joint inversion of source location and focal mechanism of microseismicity. *Geophysics* **2016**, *81*, KS41–KS49. [[CrossRef](#)]
36. Chen, W.; Ni, S.; Kanamori, H.; Wei, S.; Jia, Z.; Zhu, L. CAPjoint, a computer software package for joint inversion of moderate earthquake source parameters with local and teleseismic waveforms. *Seismol. Res. Lett.* **2015**, *86*, 432–441. [[CrossRef](#)]
37. Zhao, X.; Wang, C.; Zhang, H.; Tang, Y.; Zhang, B.; Li, L. Inversion of seismic source parameters from satellite InSAR data based on deep learning. *Tectonophysics* **2021**, *821*, 229140. [[CrossRef](#)]
38. Wu, K.; Zou, J.; Jiao, Y.-Y.; Zhang, X.; Wang, C. Focal mechanism of strong ground seismicity induced by deep coal mining. *Rock Mech. Rock Eng.* **2023**, *56*, 779–795. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.