Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Development of Neural Networks for Mineral Layer Surface Reconstruction and Assessment of Key Geological Characteristics Spatial Distribution

Mikhail Nikolaitchik
Artemij Lysionak
Vladimir Shtyk
shtykvladimir86@gmail.com

Research Article

Keywords: surface reconstruction, Signed Distance Function, machine learning, geological surface, neural networks, spatial data

Posted Date: November 4th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11251047/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Development of Neural Networks for Mineral Layer Surface Reconstruction and Assessment of Key Geological Characteristics Spatial Distribution

Nikolaitchik M.A.[a*], Lysionak A.P.[a**], Shtyk V.V.[a***](corresponding author)

[a]Belarusian State University, Minsk, 220030 Belarus

[*]*e-mail*: *nikolaitchikMA@bsu.by*

[**]*e-mail*: *lysjonokartemij@gmail.com*

[***]*e-mail*: *shtykvladimir86@gmail.com*

Abstracts

This article examines the sequential reconstruction of a mineral productive-layer surface and the qualitative geological parameter distributed across it from incomplete and noisy spatial data. A software package has been developed to generate synthetic surfaces and parameter fields, introduce controlled distortions, form training samples and compare neural network results with those obtained using classical interpolation methods. In the first stage a fully connected neural network approximates the signed vertical distance. The surface is determined by the zero level of the resulting implicit function constructed according to the Signed Distance Function (SDF) principle. The maximum number of training epochs was 200. In the second stage a coordinate-based multilayer perceptron (MLP) uses sparse measurements to reconstruct a continuous qualitative-parameter field on the reconstructed surface. The effects of number of observations, coordinate noise, parameter noise, local data gaps and hidden regions were investigated. Classical methods yielded lower Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) values on a smooth surface. For a vertical discontinuity, the SDF model had the lowest MAE across the entire range studied and achieved a lower RMSE when the number of points was small. In the case of local gaps SDF model showed the lowest RMSE and MAE. In the experiment with a large hidden region, linear interpolation yielded the best overall RMSE. The second neural network maintained a constant error as parameter noise increased and enabled rapid re-computation of the field after training. However, smoothed radial basis function (RBF) interpolation yielded a lower RMSE than the neural network. The results of the study show that the neural network approach does not have a universal advantage over common interpolation

methods and should be based on the data structure, required accuracy, and expected recalculation frequency.

Keywords: surface reconstruction, Signed Distance Function, machine learning, geological surface, neural networks, spatial data

1 Introduction

In the fields of geological modelling, geomechanics and mining, geological surfaces are reconstructed from a finite set of coordinate, geodetic or geophysical measurements (Cheng et al. 2025). Such data may describe productive-layer roof or floor elevations, geological-body boundaries, mine-opening surfaces, or sections of rock mass where displacements and deformations are recorded (Li et al. 2024a). Modern technologies such as laser scanning, aerial surveying and photogrammetry enable the acquisition of highly detailed spatial data. However, the results of various monitoring methods may contain systematic and random discrepancies (Zhan et al. 2024). The resulting three-dimensional geological models are used to analyze deposit structure, ore-body locations, and the spatial distribution of qualitative geological parameters under investigation (Liu et al. 2024).

The raw measurement data are typically unevenly distributed, contain coordinate errors, outliers and local data gaps, and do not provide complete coverage of the study area. Data gaps, geological heterogeneity and errors in the mathematical model give rise to uncertainty, which accumulates during the data acquisition process (Lindi et al. 2024). Spatial reconstruction errors can affect object shape, reserve volume and component concentration assessments and consequently affect the quality of subsequent engineering decisions. Recent geotechnical research has combined sparse site measurements with data-driven models and domain-informed spatial-correlation representations to estimate properties at unsampled locations (Chen et al. 2024).

Research into ore quality assessment shows that machine learning methods can reduce geostatistical processing errors. However, the resulting benefit depends on the source-data structure, training-sample generation method, and spatial-validation accuracy (Olmos-de-Aguilera et al. 2023). Therefore a controlled study of the effects of point density, noise, data gaps, and observation-region limits is a prerequisite for a valid comparison between methods.

Classical methods for reconstructing geological surfaces and parameter fields include the nearest-neighbour method, irregular triangulation, linear and cubic interpolation, the weighted inverse distance method, radial basis functions

and kriging (Kebede et al. 2024; Sadeghi et al. 2024). Comparative studies show that there is no universally superior interpolation method. Relative accuracy depends on point density and configuration, field heterogeneity, and the selected evaluation metric (Bronowicka-Mielniczuk and Mielniczuk 2023; Cui et al. 2025; Kebede et al. 2024; Sadeghi et al. 2024). Classical methods do not require prior training and usually have a transparent mathematical interpretation. However, their results depend directly on the local distribution of measurements. Outside the convex hull of the input points or in regions where measurements are distributed too irregularly significant errors may arise, necessitating the additional interpolation of uncertain values.

A comparative analysis of modern surface reconstruction algorithms also shows that machine learning-based methods can outperform traditional ones under consistent training and testing conditions, whilst classical algorithms often retain greater robustness to defects in real-world data that are not present in the training set (Ouasfi and Boukhayma 2024; Schirmer et al. 2024; Sulzer et al. 2025). More broadly, data scarcity, interpretability, and reliable generalization remain central challenges for the practical adoption of artificial intelligence in geotechnical engineering (Sheil et al. 2026).

Instead of constructing a surface explicitly, it can be represented implicitly as a level set of a continuous scalar function. One of the best-known approaches in this class is DeepSDF, in which a neural network approximates the signed distance function, and the object's boundary is defined by its zero level (Park et al. 2019). Implicit representation is not tied to a fixed discretization and allows function values and derivatives to be computed at any spatial point. The Implicit Geometric Regularization method demonstrated that an implicit surface can be trained directly from a point cloud using the zero-level condition and a unit-gradient-norm constraint (Gropp et al. 2020). The SIREN architecture has demonstrated that periodic activation functions allow for a more accurate representation of high-frequency spatial dependencies and the derivatives of implicit functions (Sitzmann et al. 2020).

Generating training targets for the Signed Distance Function (SDF) is a significant challenge as the reference surface is usually unknown for real-world measurements. The SAL (Sign Agnostic Learning) approach enables a latent representation to be learned from unlabeled data without pre-computed signed values (Atzmon and Lipman 2020). In Neural-Pull, the network is trained to map spatial queries to the nearest surface points by jointly utilizing the predicted distance and its gradient (Ma et al. 2021). These methods demonstrate that a neural implicit surface can be trained not only through direct regression on known SDF values, but also via geometric constraints that define the behavior of the function in the neighborhood of the observed point cloud.

Recent research is developing this approach with a view to improving robustness against noise, sparsity and geometric ambiguity. In the (p)-Poisson method, the distance-function properties are defined via differential constraints and the condition that the gradient field is potential (Park et al. 2023). Neural-Singular-Hessian utilizes the distance-function Hessian properties to suppress spurious geometry and recover details from unoriented point clouds (Wang et al. 2023). The use of sparse convolutional networks allows computations to be concentrated near the surface and improves reconstruction robustness to measurement noise (Wang et al. 2024). In SDFConnect, topological constraints are additionally utilized for reconstruction from a sparse point cloud (Jignasu et al. 2024). However, for non-closed sheet-like surfaces, the standard partitioning of space into interior and exterior regions is not always natural. An open surface is defined as one that has a boundary but does not form a closed shell. Research by IPVNet shows that, when reconstructing such surfaces, SDF can artificially close data gaps. Therefore, the choice of SDF, unsigned distance or directed distance relative to the surface must be justified (Arshad and Beksi 2023).

Reconstructing a scalar field over the surface is a separate task. Such a parameter may be the valuable-component concentration, density, moisture content, stress, strain, displacement, mining convergence or displacement velocity. Neural implicit representations are used not only for geometry but also to reconstruct continuous physical fields from sparse and irregular observations (Luo et al. 2024). Coordinate neural networks have also been shown to reconstruct scalar geophysical fields from sparse measurements and compute their spatial derivatives scalar geophysical fields from sparse measurements (Smith et al. 2025). This provides the basis for developing a sequential system in which the first model reconstructs the geological surface, whilst the second model reconstructs the parameter field on either the reference or reconstructed geometry.

A single metric is insufficient for quantitative model evaluation. The Mean Absolute Error (MAE) characterizes the average absolute deviation, whilst the Root Mean Squared Error (RMSE) places greater weight on large errors. The maximum absolute error indicates the greatest local deviation. The coefficient of determination R^2 reflects the proportion of the variability in the reference values explained by the model. The correlation coefficient characterizes the consistency of spatial variations. Error maps identify regions of reduced accuracy (Bronowicka-Mielniczuk and Mielniczuk 2023; Sulzer et al. 2025). When hidden regions are present, the metrics are also calculated separately for the observation region and for hidden regions.

Despite the active development of neural reconstruction methods, most studies consider either geometric reconstruction or the interpolation of a single

physical field. How geometry reconstruction errors affect subsequent parameter-field reconstruction has received considerably less attention.

This study proposes a sequential two-stage approach that first reconstructs the productive-layer surface and then estimates the parameter field distributed across it.

In the first stage, an implicit neural network representation of the surface is constructed from a sparse and distorted point cloud. In the second stage, the continuous parameter field is reconstructed on the resulting geometry using sparse parameter values. The additional second-stage error introduced by using reconstructed rather than reference geometry is also evaluated.

The aim of this study is to develop and investigate a neural network method for the sequential reconstruction of the geological surface geometry and the parameter field defined on the surface, based on sparse, incomplete and distorted measurements, as well as to determine the conditions under which such an approach outperforms classical interpolation methods.

To achieve this objective, synthetic reference surfaces and parameter fields are generated. Coordinate noise, parameter noise, sparsity, local data gaps, outliers and observation-region constraints are simulated.

As part of the study, a neural network model for implicit geometry representation is trained and a parameter reconstruction model is developed. The results obtained using neural networks are compared with common interpolation methods. Reconstruction quality is evaluated separately for the geometry and the target-parameter field on both reference and reconstructed surfaces. In the final stage error propagation across the full two-stage pipeline is investigated.

2 Problem Statement

The problem considered in this research is the sequential reconstruction of the productive-layer surface and its continuous qualitative-parameter field from a limited set of observations (Figure 1). The input is a discrete point cloud of spatial measurements:

$$\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{N}, \qquad \mathbf{p}_i = (x_i, y_i, z_i, t_i) \quad (1)$$

where P is the set of spatial measurements (point cloud); p_i is the measurement vector at the i-th observation point; N is the number of observation points; x_i, y_i, and z_i are the spatial coordinates of the i-th point; and t_i is the measured value of the qualitative parameter.

Two neural network models are applied sequentially to solve the problem.

At the first stage, the first neural network uses the coordinates x_i, y_i, and z_i to reconstruct the continuous surface as the zero level of the SDF.

At the second stage, the sparse measured values t_i are associated with the points of the reconstructed surface. These values are used only as targets when training the second network.

After training, the second network computes the function $\hat{t}(x, y, z)$ at all surface points, thereby producing a smooth continuous parameter field. This separation prevents reference-value leakage between the two stages.

The complete two-stage reconstruction workflow is shown in Fig. 1

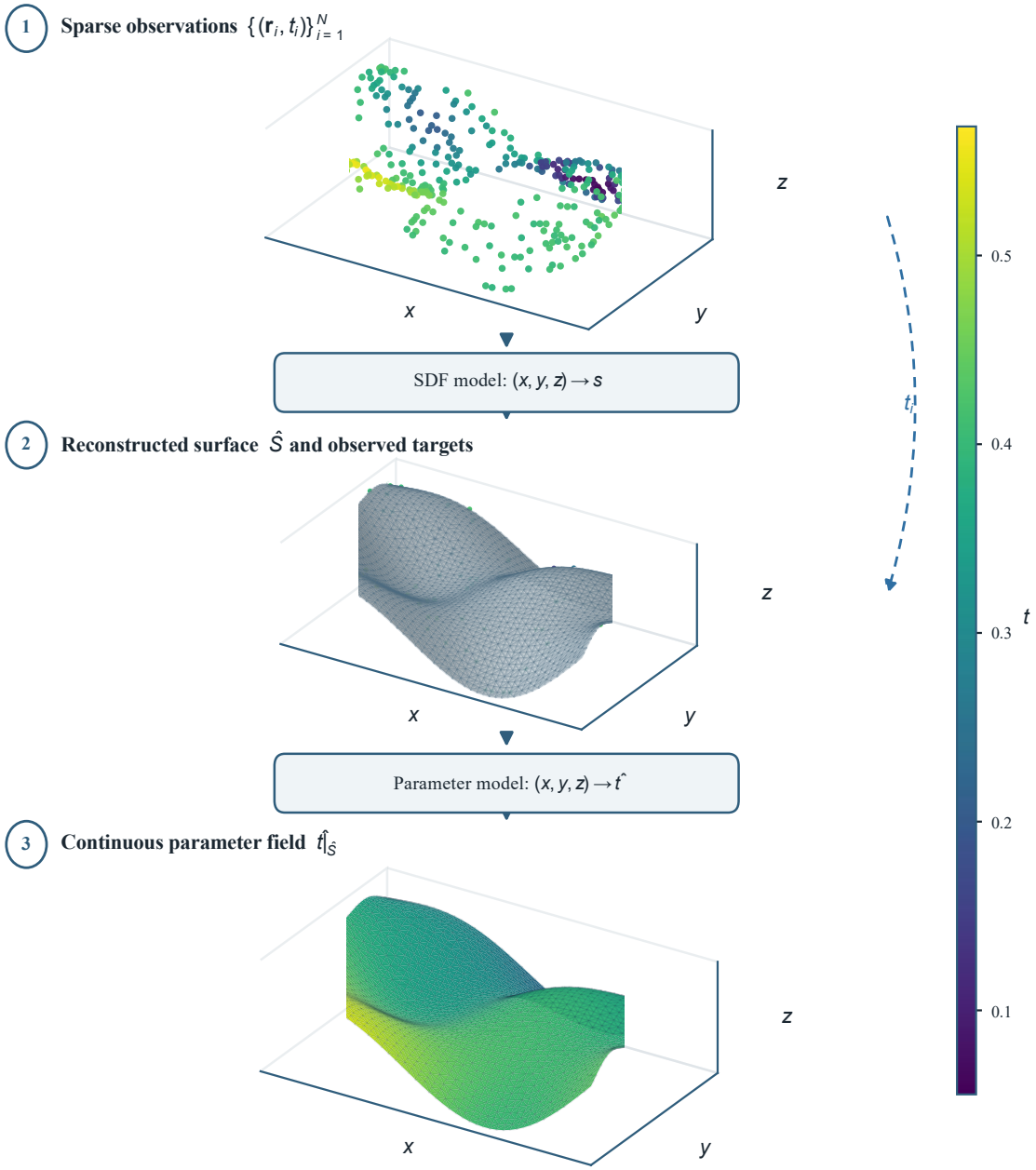


Fig. 1 - Sequential reconstruction of the geological surface and qualitative-parameter field

3 Neural Network Architectures and Training Methods

3.1 First Neural Network: Geological Surface Reconstruction

The first neural network reconstructs only the geological surface. An SDF-like implicit representation is used in the proposed implementation. The network predicts a signed vertical distance whose zero level defines the surface. The

qualitative parameter t is not provided as an input to the first model. Its sparse measured values are retained for training the second network.

A multilayer fully connected perceptron (MLP) that approximates the signed vertical distance is used to reconstruct the surface (Gropp et al. 2020; Park et al. 2019). The model receives a three-dimensional vector as input:

$$\mathrm{r} = (\mathrm{x}, \mathrm{y}, \mathrm{z}) \quad (2)$$

where r is the input vector and x, y, and z are the spatial coordinates of a point.

The network output is the predicted signed vertical distance to the reference surface:

$$\mathrm{s} = \mathrm{f}_\theta(\mathrm{x}, \mathrm{y}, \mathrm{z}) \quad (3)$$

The reconstructed surface is defined as the set of points for which the predicted distance is equal to zero:

$$\mathrm{S} = \{(\mathrm{x}, \mathrm{y}, \mathrm{z}) | \mathrm{f}_\theta(\mathrm{x}, \mathrm{y}, \mathrm{z}) = 0\} \quad (4)$$

Before training, the coordinates x, y, and z and the target values s are standardized separately. The mean calculated from the training set is subtracted from each value, and the result is divided by the corresponding standard deviation. The same transformation is applied during validation and inference. Only the means and standard deviations calculated from the training set are used. This procedure brings the variables to comparable scales and facilitates network training.

Positional encoding is not used. The standardized coordinates x, y, and z are supplied directly to the network. The nonlinear relationship between the coordinates and the function value is learned through a sequence of fully connected layers with the Softplus activation function (Gropp et al. 2020).

The MLP architecture consists of seven fully connected hidden layers with 256 neurons each and one output neuron. The smooth nonlinear Softplus activation function is used in the hidden layers:

$$\mathrm{f}_\beta(\mathrm{u}) = \frac{1}{\beta} \ln\left(1 + \mathrm{e}^{\beta \mathrm{u}}\right) \quad (5)$$

where $\mathrm{f}_\beta(\mathrm{u})$ is the value of the Softplus function for the input argument u, and $\beta > 0$ is a parameter controlling the steepness of the function.

The output layer remains linear because the predicted signed vertical distance can take both positive and negative values.

The architecture of the first neural network is shown in Figure 2.

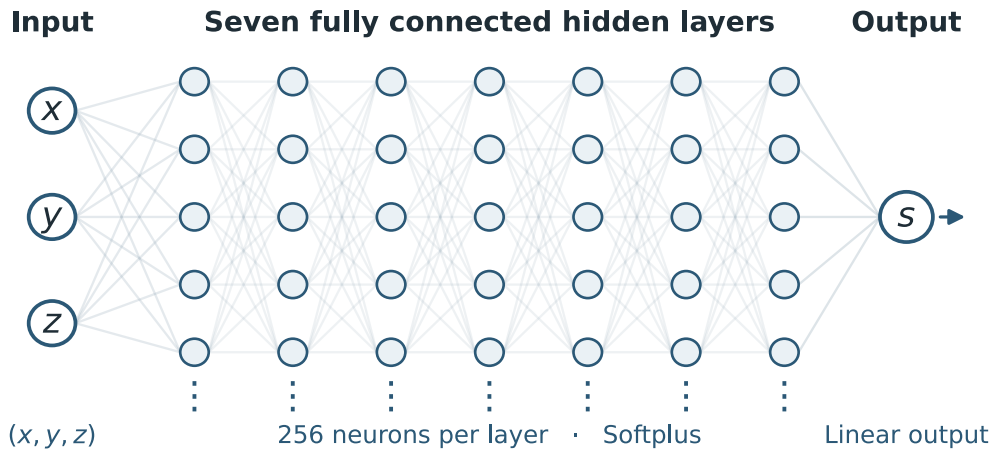


Fig. 2 - Architecture of the first neural network for geometry reconstruction

The network architecture was selected based on preliminary computational experiments. Seven hidden layers provide sufficient capacity to represent the nonlinear relationship between the coordinates and the signed vertical distance while maintaining an acceptable training time.

The Softplus function was selected because of its smoothness, which is required for approximating a continuous surface.

The hidden-layer activations and the output are computed as follows:

$$h^{(1)} = \varphi(W^{(1)}r + b^{(1)}) \quad (6)$$

$$h^{(l)} = \varphi(W^{(l)}h^{(l-1)} + b^{(l)}) \quad (7)$$

$$s = W_{out}h^{(L)} + b_{out} \quad (8)$$

where r is the input vector; h is the output of a hidden layer; l is the layer index; W and b are the weight matrix and bias vector of the corresponding layer, respectively; L is the number of hidden layers; W_{out} and b_{out} are the parameters of the linear output layer; φ is the Softplus function. In this study β is assumed to be equal to 1.

The geometry-model training set consists of spatial points paired with their reference vertical distances:

$$D = \{(r_i, s_i), i = 1, \dots, N\} \quad (9)$$

where D is the first-model training set; r_i is the coordinate vector of the i-th point; s_i is the corresponding reference signed vertical distance; N is the number of training samples.

Two additional groups of spatial points are generated for training (Figure 3). The first group consists of points located in the vicinity of the surface. These points are generated by adding a small spatial perturbation to randomly selected points on the reference surface.

The additional training points are divided into two groups. The first group, comprising 85% of the points, is generated near the surface, where the zero level must be estimated most accurately. The remaining 15% form the second group and are distributed uniformly throughout the three-dimensional bounding box of the computational domain, thereby constraining the function away from the zero level.

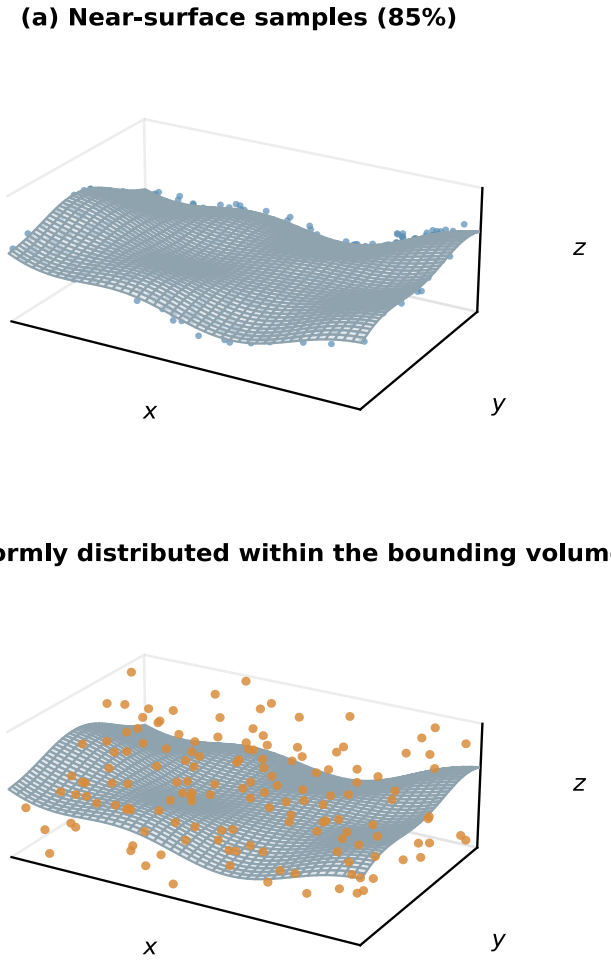


Fig. 3 - Generation of additional training points: (a) in the vicinity of the reference surface; (b) uniformly within the bounding volume

The data for each object is divided into training, validation, and test subsets using non-overlapping spatial blocks. The training subset is used to up the network parameters. The validation subset is used to select the best model checkpoint. The test subset is used only for final accuracy assessment.

The first network is trained using backpropagation with the Adam optimizer (Kingma and Ba 2015) and the mean squared error loss function. A mini-batch size of 8192 is used and the maximum number of training epochs is set to 200. The validation error is calculated after each epoch. The model state corresponding to the minimum validation error is retained for subsequent use.

After training, the model is applied to a regular computational grid. For each horizontal coordinate pair (x,y), the function is evaluated at a sequence of grid points along the vertical z-direction. The reconstructed elevation is determined from the position of the zero level. In the discrete search procedure, the grid node with the minimum absolute predicted value is selected. This procedure allows the neural network reconstruction, the reference surface, and the classical methods to be compared at the same grid nodes.

Predictions at the computational grid nodes are evaluated in batches. The full point set is divided into batches, which are sequentially passed through the model. The resulting predictions are then combined into a single array. This batch inference procedure limits GPU memory consumption without changing the calculated values.

An intermediate surface reconstruction is saved every 20 epochs for diagnostic purposes. The final model state is selected exclusively according to the minimum validation error. Test data are not used for checkpoint selection.

In the first-network implementation the target value is defined as the signed vertical distance between the input-point elevation and the reference-surface elevation at the same horizontal coordinates.

This formulation corresponds to a surface of the form $z = f(x, y)$. A positive distance indicates that the point is located above the reference surface, whereas a negative residual indicates that it is located below the surface.

$$s_i = z_i - z_{ref}(x_i, y_i) \quad (10)$$

where x_i, y_i, and z_i are the coordinates of the i-th input point; $z_{ref}(x_i, y_i)$ is the reference elevation at the same horizontal coordinates and s_i is the reference signed vertical distance.

The network parameters are determined by minimizing the mean squared error:

$$L = \frac{1}{N} \sum_{i=1}^{N} (f_\theta(x_i, y_i, z_i) - s_i)^2 \quad (11)$$

where N is the number of training samples; f_θ is the function represented by the neural network; θ denotes the trainable model parameters and s_i is the reference signed vertical distance for the i-th training sample.

3.2 Second Neural Network: Reconstruction of the Geological Parameter Distribution

After the surface geometry has been reconstructed, the surface $\hat{S}$ is passed to the second stage. The input data consist of sparse measurements of the parameter t at spatial locations. The values of t are not provided as inputs to the network during prediction. They are used only as target values at the observed points:

$$\mathcal{D}_t = \{(\mathbf{r}_i, t_i)\}_{i=1}^{N}, \qquad \mathbf{r}_i = (x_i, y_i, z_i) \quad (12)$$

where D_t is the set of observations used in the second stage; N is the number of observations; i is the observation index; $r_i = (x_i, y_i, z_i)$ is the coordinate vector of the observed point and t_i is the measured qualitative-parameter value.

Because measurement locations may not coincide with reconstructed-surface points, each observation is matched to the nearest point on the reconstructed surface of the same object:

$$\mathbf{q}_i = \arg\min_{\mathbf{q}\in\hat{S}} \|\mathbf{q} - \mathbf{r}_i\|_2 \quad (13)$$

where q_i is the point on the reconstructed surface corresponding to the i-th observation; q is an arbitrary point on the surface $\hat{S}$; $||q - r_i||_2$ is the Euclidean distance and argmin selects the point with the minimum distance.

This projection yields the training pairs (q_i, t_i). The second model is a coordinate-based multilayer perceptron (MLP) that approximates a continuous parameter field over the surface:

$$\hat{t}(\mathbf{q}) = g_\varphi(x, y, z), \qquad \mathbf{q} \in \hat{S} \quad (14)$$

where $\hat{t}(q)$ is the predicted value of the continuous parameter field at point q; g_φ is the function represented by the second neural network and φ denotes its trainable parameters.

The second network has four fully connected hidden layers with 128 neurons each. The hidden representations and the output value are computed as follows:

$$\mathbf{h}^{(l)} = \mathrm{GELU}\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right), \qquad l = 1, \ldots, 4 \quad (15)$$

$$\hat{t} = \mathbf{W}_{out}\mathbf{h}^{(4)} + b_{out} \quad (16)$$

where $h^{(l)}$ is the output of the l-th hidden layer; $h^{(0)}$ is the standardized input vector; l is the layer index; $W^{(l)}$ and $b^{(l)}$ are the weight matrix and bias vector of the corresponding layer; GELU is the activation function; $h^{(4)}$ is the output of the fourth hidden layer and W_{out} and b_{out} are the weight matrix and bias of the linear output layer.

No activation function is applied in the output layer because the predicted parameter may take arbitrary real values. After inverse standardization, the predicted values are expressed in the original units of the parameter. The GELU activation function is used in the hidden layers (Hendrycks and Gimpel 2016):

$$GELU(u) = u\Phi(u) \quad (17)$$

where u is the input value of the activation function, and Φ(u) is the cumulative distribution function of the standard normal distribution.

Before training, the coordinates x, y, and z and the target values t are standardized separately using the training set. Each variable is centered using its training-set mean and scaled by its training-set standard deviation. The same statistics are used for validation, testing and inference. The data is divided using non-overlapping spatial blocks so that neighboring points from the same region are not simultaneously included in the training and evaluation subsets (validation and test).

The Huber loss function is used to reduce the influence of outliers (Huber 1964):

$$\mathcal{L}_{\mathrm{Huber}} = \frac{1}{N}\sum_{i=1}^{N} \ell_\delta\,(\hat{t}_i - t_i), \qquad \ell_\delta(e) = \begin{cases} \frac{e^2}{2}, & |e| \le \delta, \\ \delta\left(|e| - \frac{\delta}{2}\right), & |e| > \delta \end{cases} \quad (18)$$

where $\boldsymbol{\mathcal{L}}_{\mathrm{Huber}}$ is the mean loss value; $e_i = \hat{t}_i - t_i$ is the prediction error at the i-th training point; $\hat{t}_i$ is the value predicted by the network; ℓ_δ is the piecewise-defined Huber loss function; e is its argument and δ is the threshold at which the loss changes from the quadratic to the linear regime.

Optimization is performed using the Adam optimizer with an initial learning rate of $7 \cdot 10^{-4}$. The maximum number of training epochs is set to 1200. The learning rate is reduced when the validation error stops improving and early

stopping is applied after 120 epochs without improvement. The batch size is 256 observations.

After training, the model takes the coordinates of all reconstructed-surface points and produces a smooth field $\hat{t}(x, y, z)$.

The performance of the second stage is evaluated using MAE, RMSE, R^2, the correlation coefficient, and the maximum absolute error. The coefficient of determination R^2 indicates how well the model reproduces the variability of the reference values. The correlation coefficient characterizes the consistency of spatial variations. The maximum absolute error indicates the largest local deviation.

To evaluate the propagation of the geometry reconstruction error, the parameter field is reconstructed in two identical modes. In the first model the parameter field is reconstructed on the reference surface. In the second model the parameter field is reconstructed on the surface reconstructed by the first network. For RMSE, the contribution of the first stage is defined as $\Delta RMSE_t = RMSE_t^{rec} - RMSE_t^{ref}$, where $RMSE_t^{ref}$ is the parameter reconstruction error obtained using the reference surface. $RMSE_t^{rec}$ is the corresponding error obtained using the reconstructed geometry. Both quantities are calculated relative to the reference values of t at the same evaluation points.

4 Synthetic Data Generation for the Two-Stage Model

The overall computational workflow is shown in Figure 4. Synthetic data for both neural networks are generated from a common reference model.

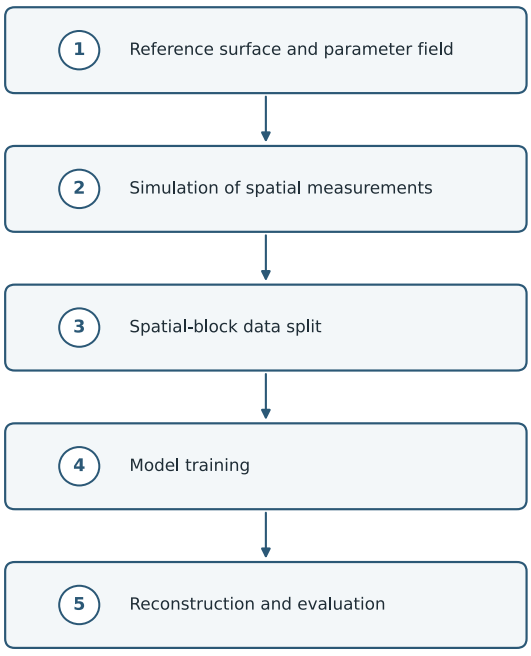


Fig. 4 - Main stages of the computational workflow

Synthetic data generation begins by defining the surface geometry and qualitative parameter field. Data sparsity, noise, local gaps, observation-region restrictions, and outliers are then simulated.

Synthetic data provides exact reference surface and parameter fields. Therefore, all methods can be compared under identical conditions and the errors at both reconstruction stages can be calculated relative to the corresponding reference data. By varying the distortion parameters systematically, we can investigate the individual and combined effects of point count, noise level, and spatial data gaps.

All experiment parameters are stored in a single configuration file to make the computational experiments reproducible.

At the first stage, a reference three-dimensional surface is generated. It represents the overall curvature, wave-like variations, and local irregularities of a geological surface. The surface is defined by the continuous function

$$z = f(x, y) \quad (19)$$

where x and y are the horizontal coordinates, z is the surface elevation, and f defines the synthetic-object geometry.

Figure 5 shows how several spatial components are combined to generate the reference surface. This representation makes it possible to define geometric features at different spatial scales independently and to control their parameters during the computational experiments.

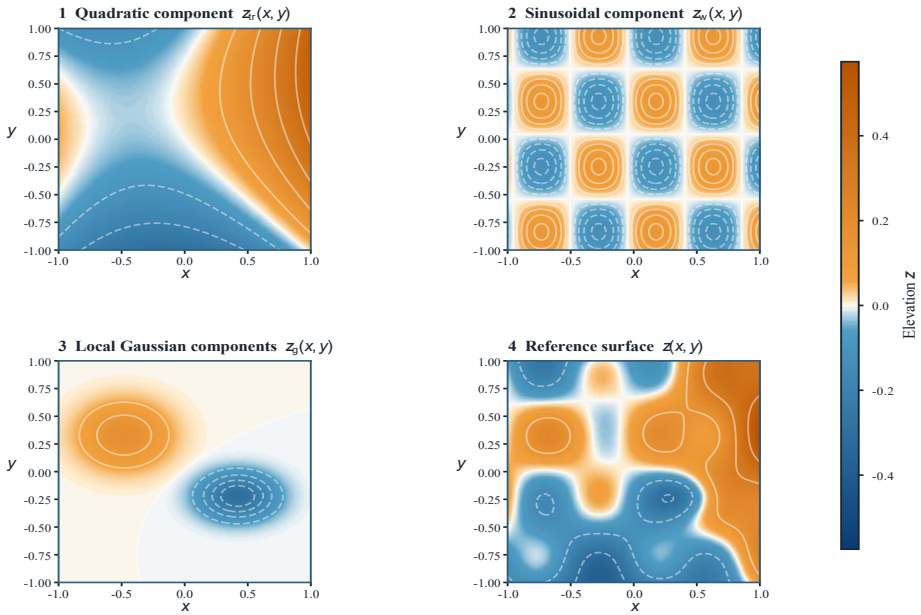


Fig. 5 - Generation of the synthetic surface as a sum of components

The synthetic surface can be represented as the sum of the following components.

1) The quadratic component defines a smooth variation in elevation over the entire computational domain. It represents the global surface trend, including its overall direction, slope and curvature.

$$z_{tr}(x,y) = a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 \quad (20)$$

where z_{tr} is the global trend component of the surface; $a_0, \dots, a_5$ are the coefficients of the quadratic function; x and y are the spatial coordinates.

2) The sinusoidal component generate smooth undulations with specified amplitudes and spatial periods.

$$z_w(x,y) = \sum_{k=1}^{M} A_k \sin\left(\frac{2\pi x}{\lambda_{x,k}} + \varphi_{x,k}\right) \sin\left(\frac{2\pi y}{\lambda_{y,k}} + \varphi_{y,k}\right) \quad (21)$$

where z_w is the wave component of the surface; M is the number of sinusoidal components; A_k is the amplitude of the k-th harmonic; $\lambda_{x,k}$ and $\lambda_{y,k}$ are the spatial periods along the x- and y -coordinates, respectively; $\varphi_{x,k}$ and $\varphi_{y,k}$ are the corresponding phase shifts.

3) Local Gaussian components generate spatially confined elevations and depressions. Local features are spatially confined deviations from the overall surface geometry.

$$z_g(x,y) = \sum_{j=1}^{L} B_j \exp\left(-\frac{(x-x_j)^2+(y-y_j)^2}{2\sigma_j^2}\right) \quad (22)$$

where z_g is the local Gaussian component of the surface; L is the number of local Gaussian components; B_j is the amplitude of the j-th component; x_j and y_j are the coordinates of the local feature center; σ_j is the parameter that determines its characteristic size.

4) The final surface is obtained by summing the components described above. It therefore combines the overall curvature, smooth wave-like variations and local geometric deviations.

$$z(x,y) = z_{tr}(x,y) + z_w(x,y) + z_g(x,y) \quad (23)$$

where $z(x, y)$ is the elevation of the reference surface; z_{tr} is the global trend component; z_w is the wave component; z_g is the local Gaussian component.

The coefficients and parameters of the reference-surface components are generated randomly within predefined ranges. To ensure the reproducibility of each experiment, the initial state of the pseudorandom number generator is fixed.

This generation procedure produces smooth surfaces with variations at different spatial scales. It also allows features typical of mineral layers roof and floor models, mine-working geometry and spatial deformation patterns within mining objects to be reproduced (Cui et al. 2025; Zhan et al. 2024).

A scalar field is generated alongside the surface geometry to represent a qualitative geological or geomechanical parameter distributed over the surface. The field is defined using a set of local influence centers:

$$t(x, y) = t_0 + \sum_{k=1}^{K} A_k \exp\left(-\frac{(x-x_k)^2+(y-y_k)^2}{2\sigma_k^2}\right) \quad (24)$$

where t_0 is the background value of the parameter; K is the number of influence centers; x_k and y_k are the coordinates of the k-th influence center; A_k is its amplitude; σ_k is the parameter that determines the spatial scale of influence.

The field value at each point is determined by summing the Gaussian components. The resulting distribution is then normalized to a predefined range. This approach produces smooth spatial variations in the parameter and makes it possible to define regions of locally increased or decreased values without introducing artificial sharp boundaries.

Depending on the specific application, the parameter t may represent the concentration of a valuable component, density, moisture content, or ash content of the mineral. In typical geomechanical applications, it may correspond to Young's modulus, Poisson's ratio, strength, strain, displacement magnitude, or another scalar quantity characterizing the state of the rock mass. Such spatial variability is accounted for when constructing geomechanical models of mining objects (Li et al. 2024b).

The parameter t is generated using the procedure illustrated in Figure 6.

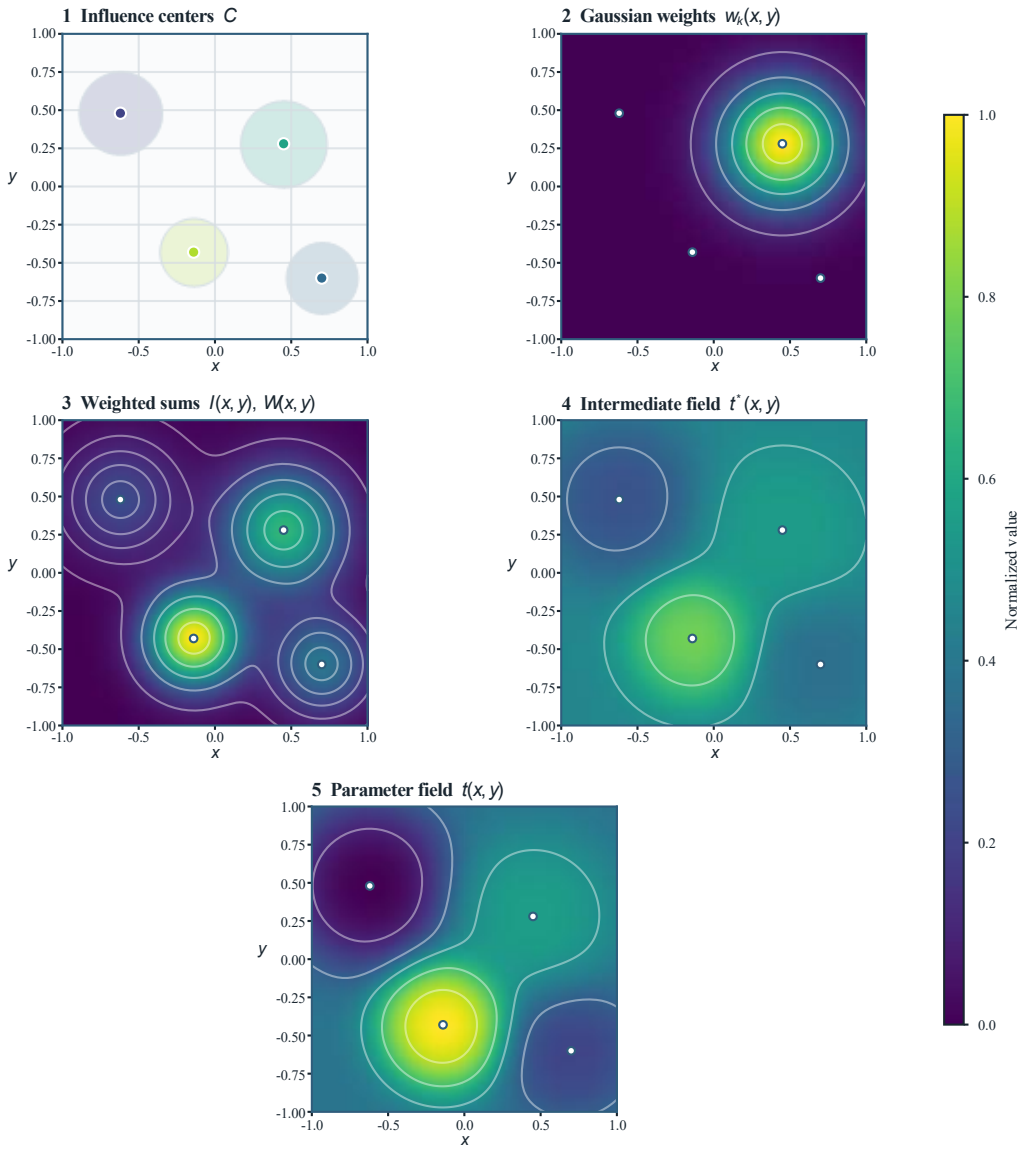


Fig. 6 - Sequence of qualitative parameter field generation

1) At the first stage, the influence centers, their values and radius of influence are specified.

$$\mathcal{C} = \{(\mathbf{c}_k, t_k, \sigma_k)\}_{k=1}^{K}, \quad \mathbf{c}_k = (x_k, y_k) \quad (25)$$

where $\mathcal{C}$ is the influence centers set of parameters; K is the number of influence centers; $\mathbf{c}_k$ is the position vector of the k-th influence center; x_k, y_k are its coordinates; t_k is the parameter value at the center; and σ_k is the radius of influence.

2) At the second stage, each influence center's Gaussian weight is evaluated at point (x, y).

$$w_k(x,y) = \exp\left(-\frac{(x-x_k)^2+(y-y_k)^2}{2\sigma_k^2}\right) \quad (26)$$

where $w_k(x,y)$ is the weight of the k-th influence center.

3) At the third stage, the contributions of all influence centers are summed.

$$I(x,y) = \sum_{k=1}^{K} t_k\, w_k(x,y), \quad W(x,y) = \sum_{k=1}^{K} w_k\,(x,y) \quad (27)$$

where $I(x,y)$ is the total weighted contribution of the influence centres and $W(x,y)$ is the sum of their weights.

4) At the fourth stage, a background term is added to the intermediate field to ensure stable behavior far from the influence centers.

$$t^*(x,y) = \frac{I(x,y)+w_0 t_{bg}}{W(x,y)+w_0} \quad (28)$$

where $t^*(x,y)$ is the intermediate field; t_{bg} is the background value of the parameter; $w_0 > 0$ is the weight of the background contribution, which prevents instability when the influence of the centers is small.

5) At the fifth stage, the field is normalized and scaled to the target t range.

$$t(x,y) = t_{min} + (t_{max} - t_{min})\frac{t^*(x,y)-t^*_{min}}{t^*_{max}-t^*_{min}} \quad (29)$$

where $t(x,y)$ is the final parameter field; t_{min}, t_{max} are the specified bounds of the final range; t^*_{min}, t^*_{max} are the minimum and maximum values of the intermediate field, respectively.

5 Simulation of Spatial Measurements

After generating the reference model, field data acquisition for the qualitative parameter distribution over the productive layer is simulated.

A subset of reference-model points is selected to represent field measurements collected during geomechanical monitoring.

Measurement density is a key experimental parameter and it is varied over a wide range. This makes it possible to investigate how input-data sparsity affects reconstruction quality.

To make the synthetic data more representative of real geological measurements and sampling conditions, representative measurement distortions are introduced into the model. These include random errors in the spatial coordinates, noise in the additional parameter t, local regions without measurements, random thinning of the point cloud, and outliers caused by measurement equipment errors or data-processing errors.

The cases where measurements cover only part of the investigated surface are also considered in this research. This makes it possible to evaluate the ability of the methods to reconstruct the geometry beyond the observation region.

Local data gaps are generated by removing points from selected regions of the surface, thereby simulating the absence of observations. Point-cloud thinning is performed by removing a specified fraction of measurements throughout the entire domain. The observation region is varied by restricting measurement coverage. This allows interpolation accuracy within the observed region and extrapolation accuracy outside it to be evaluated separately.

Figure 7 shows, in sequence, coordinate noise, noise in parameter t, a local data gap, point-cloud thinning, exclusion of part of the domain, and the addition of outliers.

Figure 7 presents the individual types of distortions. In specific scenarios, they are applied either separately or in predefined combinations. This makes it possible to generate training and test data for both neural networks under controlled conditions.

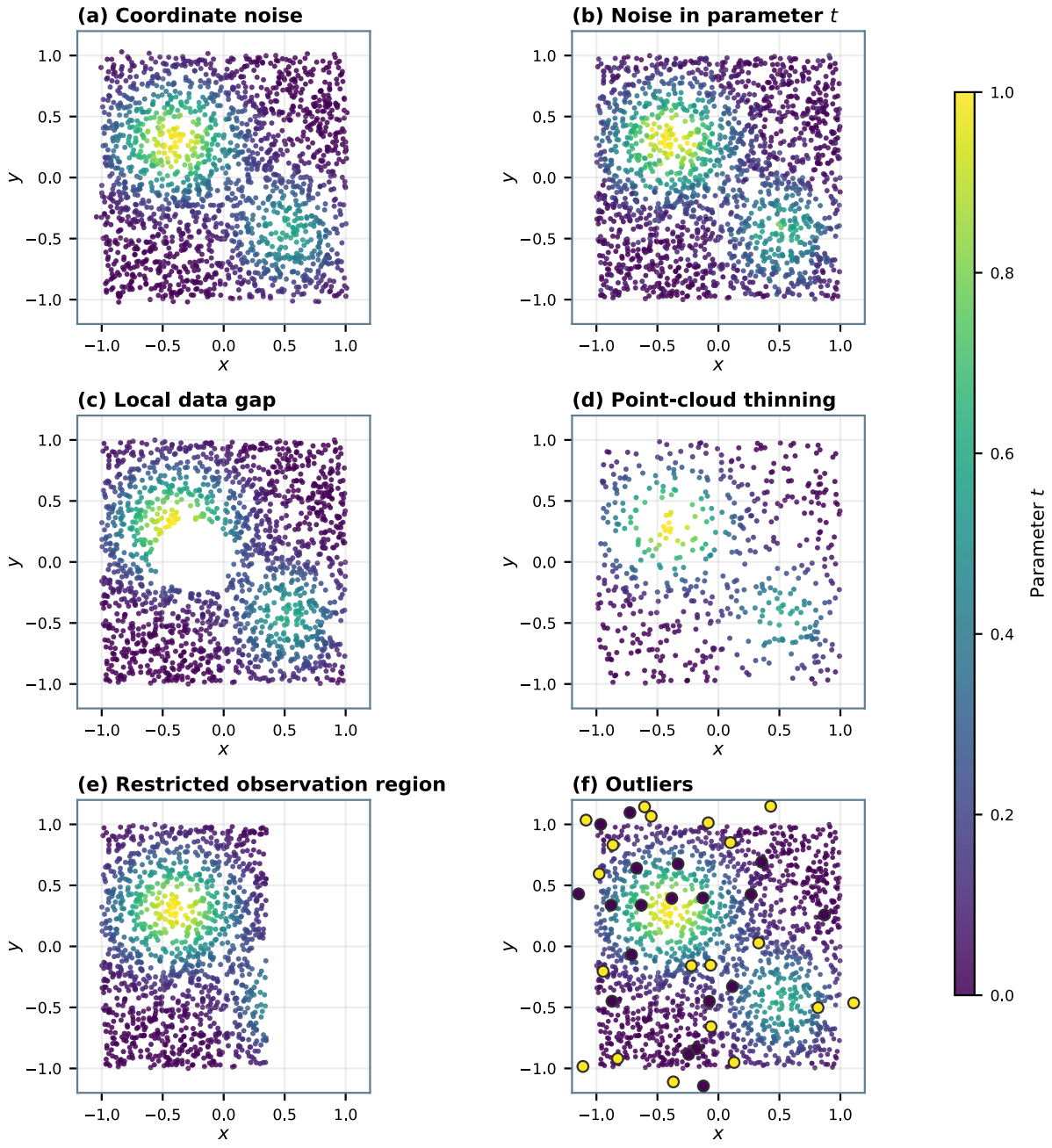


Fig. 7 - Simulation of distortions and incompleteness in the input point cloud

As mentioned above, distortions can be applied independently or in combination. This makes it possible to simulate a wide range of conditions characteristic of practical geomechanical monitoring problems. All simulation parameters are specified centrally and can be varied during a series of computational experiments.

6 Comparison of Neural Network Results with Classical Interpolation Methods

The results obtained with each neural network are compared with classical interpolation methods.

For geological surface reconstruction, the nearest-neighbor method, linear interpolation, inverse distance weighting (IDW) and a triangulated irregular network (TIN) are used (Cui et al. 2025; Kebede et al. 2024; Sadeghi et al. 2024).

The parameter-field reconstruction stage is benchmarked against cubic interpolation and smoothed radial basis function (RBF) interpolation (Bronowicka-Mielniczuk and Mielniczuk 2023; Kebede et al. 2024; Sadeghi et al. 2024). RBF interpolation is not used by the first neural network and serves only as an independent benchmark at the parameter-field reconstruction stage.

At each stage, all methods use the same input observations, and their results are evaluated at the same computational grid points.

7 Results of Computational Experiments

7.1 Geological Surface Reconstruction

For each set of parameters, the surface was reconstructed using the neural-network-based SDF approach and classical interpolation methods.

Three tests were conducted to evaluate neural network performance.

In the first test, coordinate noise was added to a smooth reference surface.

In the second test, both noise and a vertical discontinuity along the boundary x = 0 were introduced. The modified surface was defined piecewise as follows:

$$z = \begin{cases} f(x,y) + c_1, & x < 0, \\ f(x,y) + c_2, & x \geq 0 \end{cases} \quad (30)$$

where $f(x, y)$ is the original smooth surface; c_1 and c_2 are constant vertical offsets applied to the left and right parts of the surface, respectively; x and y are the horizontal coordinates; z is the elevation of the modified surface. The difference $c_2 - c_1$ determines the magnitude of the discontinuity.

In the third test, the point cloud was thinned and all points within two local regions were removed. This scenario evaluates surface reconstruction under non-uniform observation density and in regions without direct measurements. The corresponding distortions are illustrated in the previous section.

For each test, the number of input points was varied and the resulting reconstructions were compared using RMSE and MAE calculated relative to the same reference values.

Surface-reconstruction accuracy was evaluated using the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) (Bronowicka-Mielniczuk and Mielniczuk 2023; Cui et al. 2025; Kebede et al. 2024; Sadeghi et al. 2024)

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{z}_i - z_i^{\mathrm{ref}}\right| \quad (31)$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{z}_i - z_i^{\mathrm{ref}}\right)^2} \quad (32)$$

where N is the number of computational grid points; $\hat{z}_i$ is the reconstructed elevation at the i-th grid point; z_i^{ref} is the corresponding reference elevation. MAE characterizes the mean absolute deviation, whereas RMSE assigns greater weight to large local errors.

RMSE and MAE versus number of points are shown for the three tests in Figures 8–10. In all legends, the term "SDF model" refers to the first neural network. TIN, IDW and the nearest-neighbor method are represented using the same colors and markers throughout the figures.

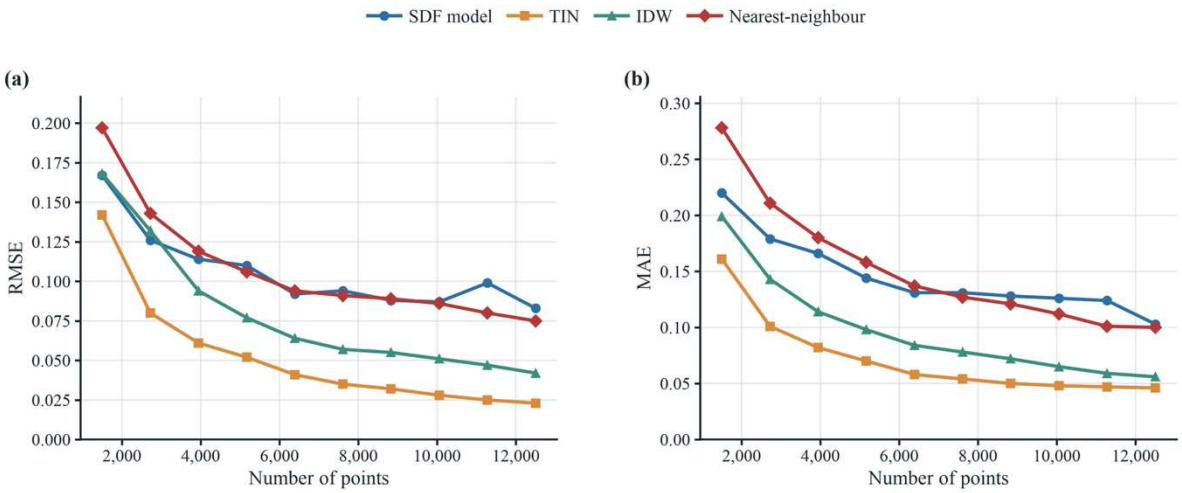


Fig. 8 - Dependence of RMSE (a) and MAE (b) on the number of points in the first test

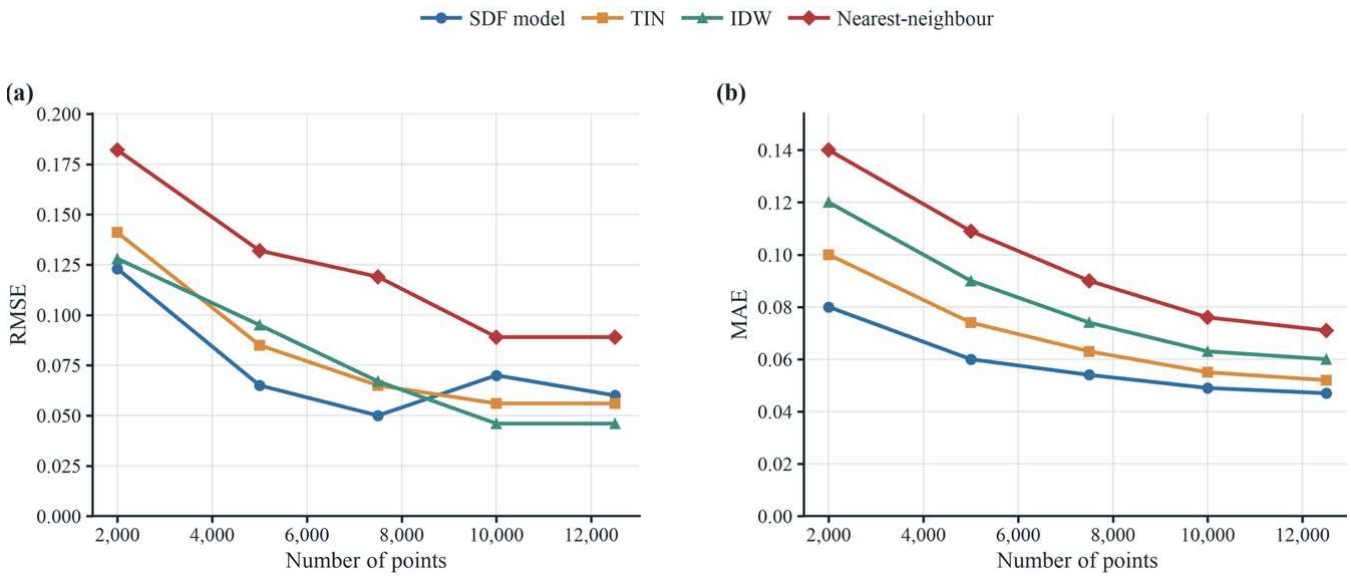


Fig. 9 - Dependence of RMSE (a) and MAE (b) on the number of points in the second test

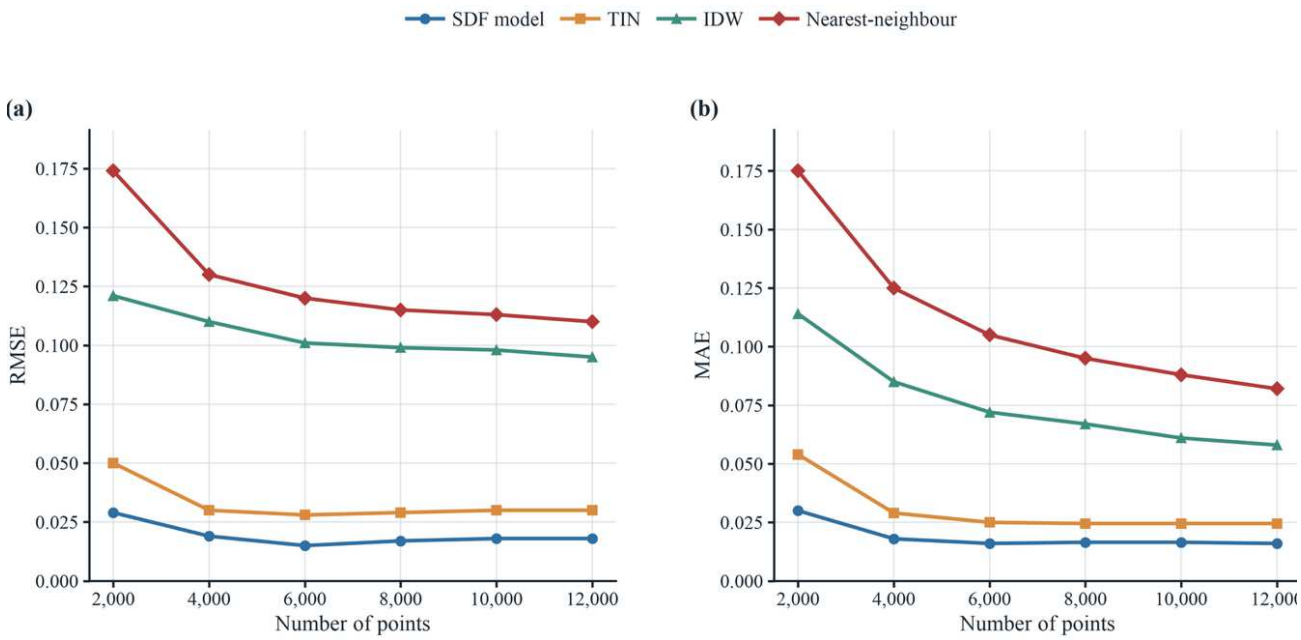


Fig. 10 - Dependence of RMSE (a) and MAE (b) on the number of points in the third test

In the first test, classical methods, particularly TIN, produced lower RMSE and MAE values for the smooth and sufficiently densely sampled surface. Under these conditions, local interpolation directly uses a sufficient number of nearby measurements. Therefore, preliminary training of the neural network model does not provide an advantage in terms of accuracy.

In the second test, the advantage of the classical methods decreased after the vertical discontinuity was introduced. In terms of MAE, the neural network

model produced the lowest error over the entire investigated range of the number of point. Its advantage in terms of RMSE was most pronounced when the amount of input data was small. Therefore, within this test, the global neural network approximation was more robust to the combination of the discontinuity and a sparse point cloud.

In the third test, which included point-cloud thinning and local data gaps, the neural network model produced the lowest RMSE and MAE values for every tested number of points. This result applies only to the specific configuration of local data gaps tested here and does not imply an automatic advantage when an arbitrarily large hidden region is present.

Additional tests reveal a limitation of the method. When part of the domain was withheld, linear interpolation produced the lowest overall RMSE of 0.004445, whereas the SDF model produced an RMSE of 0.005069. Within the observed region, the SDF model was more accurate, with an RMSE of 0.001348, compared with 0.002126 for linear interpolation. In the hidden region, however, the SDF model produced an RMSE of 0.008348, whereas both the nearest-neighbor method and linear interpolation produced an RMSE of 0.006518.

The SDF approach therefore performs best mainly for noisy data and local irregularities when observation coverage is sufficient. When substantial data gaps are present, hidden-region error should be analyzed separately because a low overall error may mask poor extrapolation.

7.2 Reconstruction of the Qualitative Parameter Field

Two second-stage experiment series were conducted on a reconstructed surface with a central hidden region.

In the first series, the parameter noise level was varied from 0 to 20% of the reference parameter field range using 400 observations.

In the second series, the number of observations was varied from 50 to 1200 at a noise level of 2%. At each setting six independent synthetic realizations were performed. In each realization, the surface, parameter field, observation locations, and noise were generated independently. The plots report the six-run mean RMSE across 219 computational grid points within the hidden region. The

term “MLP model” refers to the second neural network. The results are presented in Fig. 11 (a–c).

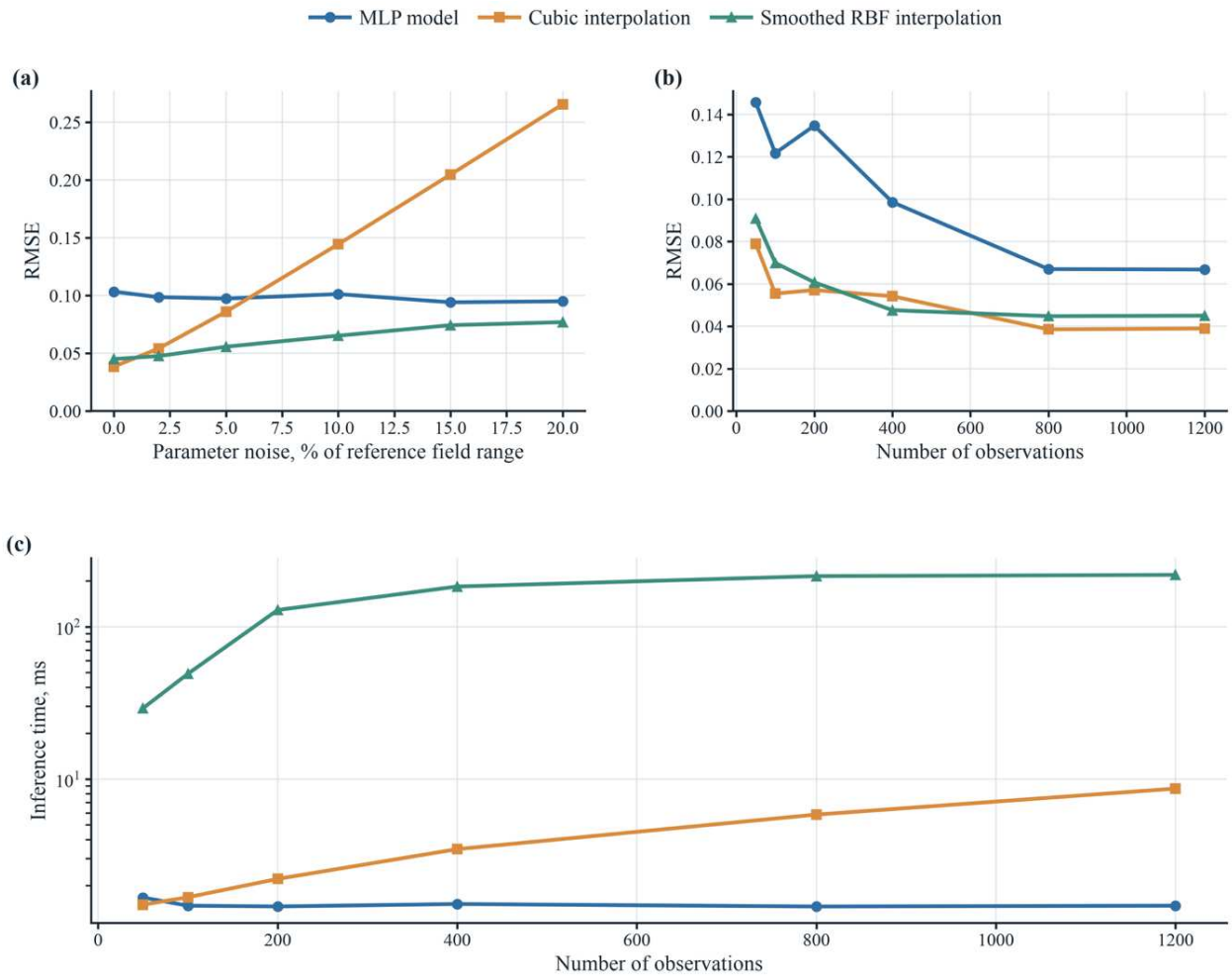


Fig. 11 - Effect of the noise level on RMSE (a), the number of observations on RMSE (b) and inference time (c)

As the noise level increased from 0 to 20%, the mean RMSE of the MLP model remained between 0.09416–0.10340 and was 0.09504 at a noise level of 20%. At 20% noise, the MLP RMSE was 64.2% lower than that of cubic interpolation (0.26565) but 23.5% higher than that of smoothed RBF interpolation (0.07699). Thus, in this scenario, the MLP model was more robust to increasing noise than cubic interpolation. However, the MLP model did not outperform smoothed RBF interpolation in accuracy.

As the number of observations increased from 50 to 800, the mean RMSE of the MLP model generally decreased from 0.14567 to 0.06702. The error then reached a plateau, with an RMSE of 0.06683 obtained at 1200 observations. At 1200 observations, cubic interpolation and smoothed RBF interpolation remained more accurate, with RMSE values of 0.03896 and 0.04502, respectively. Thus,

the MLP advantage in this series was computational time rather than accuracy-based.

Inference time was measured on a macOS (ARM64) CPU for 2500 computational grid points. Figure 11 reports the median across repeated runs. In the implemented scenario, for the MLP, timing covered prediction by an already loaded model. For the classical methods, timing included interpolator construction and field evaluation. At 1200 observations, the median inference time was 1.47 ms for the MLP, 8.65 ms for cubic interpolation, and 218.73 ms for RBF interpolation with smoothing. The median MLP training time increased from 0.21 s at 50 observations to 1.80 s at 1200 observations. The training time of the first SDF model was not recorded separately. This timing advantage therefore applies only to repeated use of the trained second network, not to the full training-and-computation cycle.

These conclusions apply to synthetic parameter fields. In a separate series of five runs, the effect of the specified synthetic surface distortion in the extrapolation scenario was $\Delta RMSE_t = 0.00348 \pm 0.00395$ with a 95% confidence interval. No statistically significant error increase was detected. In this experiment, however, the geometry distortion was defined analytically rather than obtained from the actual first-network output. Full-pipeline error propagation should therefore be evaluated separately using SDF-reconstructed surfaces.

8 Conclusion

A two-stage approach for sequentially reconstructing a productive-layer geological surface and its qualitative parameter field was proposed and investigated. The first fully connected neural network maps the coordinates x, y, and z to signed vertical distance. The second coordinate-based neural network reconstructs the continuous field $\hat{t}(x, y, z)$ from sparse parameter samples mapped to reconstructed surface points.

The first neural network produced lower errors in the tested scenarios involving local data gaps and a vertical discontinuity. However, when the hidden region was sufficiently large, linear interpolation provided the best result. In the second series of experiments, the MLP model was more robust than cubic interpolation to increasing noise in the input data but was less accurate than smoothed RBF interpolation. Repeated inference with a loaded MLP was faster than building and evaluating an RBF interpolator. This comparison does not include the neural network training time.

The results obtained show that neural networks are therefore not a universal substitute for interpolation when reconstructing geological surfaces and their associated qualitative parameter fields.

For smooth and sufficiently densely sampled surfaces, classical methods provide lower reconstruction errors and do not require training. Therefore, they are preferable under such conditions. The neural network approach was

advantageous in the tested scenarios involving a vertical discontinuity, point-cloud sparsity, and local data gaps.

The use of neural networks for reconstructing qualitative geological parameter fields demonstrated greater robustness to increasing noise in the input data compared with the interpolation approaches considered in this study. The trained model also provided fast repeated inference.

Method selection should therefore consider data structure, hidden-region size, required accuracy and expected recalculation frequency.

The developed software package provides a basis for further validation of the proposed two-stage approach using geological and geomechanical monitoring data. These results may also support the processing of large geological, mine-surveying, and geomechanical spatial datasets.

Declarations

Funding
The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing interests
The authors have no relevant financial or non-financial interests to disclose.

Data availability
The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Code availability
The source code used in this study is available from the corresponding author upon reasonable request.

Ethics approval
Not applicable.

Consent to participate
Not applicable.

Consent for publication
Not applicable.

References

Atzmon M, Lipman Y (2020) SAL: sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 2565–2574. https://doi.org/10.1109/CVPR42600.2020.00264

Arshad MS, Beksi WJ (2023) IPVNet: learning implicit point-voxel features for open-surface 3D reconstruction. J Vis Commun Image Represent 97:103970. https://doi.org/10.1016/j.jvcir.2023.103970

Bronowicka-Mielniczuk U, Mielniczuk J (2023) New indices to quantify patterns of relative errors produced by spatial interpolation models: a comparative study by modelling soil properties. Ecol Indic 154:110551. https://doi.org/10.1016/j.ecolind.2023.110551

Chen W, Ding J, Shi C, Wang T, Connolly DP (2024) Geotechnical correlation field-informed and data-driven prediction of spatially varying geotechnical properties. Comput Geotech 171:106407. https://doi.org/10.1016/j.compgeo.2024.106407

Cheng N, Ismail MAM, Muztaza NM, Pauzi FNA, Zakaria MT, Yokota Y (2025) From 2-D seismic refraction to 3-D subsurface characterization: unpacking the role of univariate spatial interpolation techniques with borehole validation. Phys Chem Earth Parts A/B/C 141:104113. https://doi.org/10.1016/j.pce.2025.104113

Cui M, Hou E, Lu T, Hou P, Feng D (2025) Study on spatial interpolation methods for high precision 3D geological modeling of coal mining faces. Appl Sci 15(6):2959. https://doi.org/10.3390/app15062959

Gropp A, Yariv L, Haim N, Atzmon M, Lipman Y (2020) Implicit geometric regularization for learning shapes. In: Proceedings of the 37th International Conference on Machine Learning, vol 119, pp 3789–3799

Hendrycks D, Gimpel K (2016) Gaussian error linear units (GELUs). arXiv:1606.08415. https://arxiv.org/abs/1606.08415

Huber PJ (1964) Robust estimation of a location parameter. Ann Math Stat 35(1):73–101. https://doi.org/10.1214/aoms/1177703732

Jignasu A, Balu A, Sarkar S, Hegde C, Ganapathysubramanian B, Krishnamurthy A (2024) SDFConnect: neural implicit surface reconstruction of a sparse point cloud with topological constraints. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp 5271–5279. https://doi.org/10.1109/CVPRW63382.2024.00536

Kebede H, Demissie Z, Tadesse H, Eshetu A (2024) Spatial interpolation techniques comparison and evaluation: the case of ground-based gravity and elevation datasets of the central Main Ethiopian Rift. Heliyon 10(12). https://doi.org/10.1016/j.heliyon.2024.e32806

Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations. https://arxiv.org/abs/1412.6980

Li J, Chen J, Ma B (2024) Study on coal mine tunnels surface displacement monitoring based on 3D laser point cloud comparison. Saf Coal Mines 55(7):219–225. https://doi.org/10.13347/j.cnki.mkaq.20231494

Li J, Yang T, Liu F, Zhao Y, Liu H, Deng W, Gao Y, Li H (2024) Modeling spatial variability of mechanical parameters of layered rock masses and its application in slope optimization at the open-pit mine. Int J Rock Mech Min Sci 181:105859. https://doi.org/10.1016/j.ijrmms.2024.105859

Lindi OT, Aladejare AE, Ozoji TM, Ranta J-P (2024) Uncertainty quantification in mineral resource estimation. Nat Resour Res 33:2503–2526. https://doi.org/10.1007/s11053-024-10394-6

Liu X, Wang G, Lv J, Peng Y, Zhao X (2024) Three-dimensional modeling and geostatistical structural analysis for ore deposit prospecting trend and wisdom mining: a case study in the Jiaoxibei gold field, China. Front Earth Sci 12:1217016. https://doi.org/10.3389/feart.2024.1217016

Luo X, Xu W, Ren Y, Yoo S, Nadiga BT (2024) Continuous field reconstruction from sparse observations with implicit neural networks. In: International Conference on Learning Representations

Ma B, Han Z, Liu Y-S, Zwicker M (2021) Neural-Pull: learning signed distance function from point clouds by learning to pull space onto surface. In: Proceedings of the 38th International Conference on Machine Learning, vol 139, pp 7246–7257

Olmos-de-Aguilera C, Campos PG, Risso N (2023) Error reduction in long-term mine planning estimates using deep learning models. Expert Syst Appl 217:119487. https://doi.org/10.1016/j.eswa.2022.119487

Ouasfi A, Boukhayma A (2024) Unsupervised occupancy learning from sparse point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 21729–21739. https://doi.org/10.1109/CVPR52733.2024.02053

Park JJ, Florence P, Straub J, Newcombe R, Lovegrove S (2019) DeepSDF: learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 165–174. https://doi.org/10.1109/CVPR.2019.00025

Park Y, Lee T, Hahn J, Kang M (2023) p-Poisson surface reconstruction in curl-free flow from point clouds. Adv Neural Inf Process Syst 36:60077–60098

Sadeghi B, Eleish AM, Morrison SM, Klump J (2024) Inverse radius weighting and its Python package “IRWPy”: a new topography-informed interpolation to enhance geological interpretations. Ore Geol Rev 172:106206. https://doi.org/10.1016/j.oregeorev.2024.106206

Schirmer L, Novello T, da Silva V, Schardong G, Perazzo D, Lopes H, Gonçalves N, Velho L (2024) Geometric implicit neural representations for signed distance functions. Comput Graph 125:104085. https://doi.org/10.1016/j.cag.2024.104085

Sheil B, Anagnostopoulos C, Buckley R, Ciantia MO, Febrianto E, Fu J, Gao Z, Geng X, Gong B, Hanley K, He P, Kolomvatsos K, Lopes BCF, Ninic J, Previtali M, Rezania M, Ruiz-Lopez A, Sun J, Suryasentana S, Taborda D, Utili S, Whyte S, Zhang P (2026) Artificial intelligence transformations in geotechnics: progress, challenges and future enablers. Comput Geotech 189:107604. https://doi.org/10.1016/j.compgeo.2025.107604

Sitzmann V, Martel JNP, Bergman AW, Lindell DB, Wetzstein G (2020) Implicit neural representations with periodic activation functions. Adv Neural Inf Process Syst 33:7462–7473

Smith LT, Horrocks T, Akhtar N, Holden E-J, Wedge D (2025) Implicit neural representation for potential field geophysics. Sci Rep 15:9799. https://doi.org/10.1038/s41598-024-83979-z

Sulzer R, Marlet R, Vallet B, Landrieu L (2025) A survey and benchmark of automatic surface reconstruction from point clouds. IEEE Trans Pattern Anal Mach Intell 47(3):2000–2019. https://doi.org/10.1109/TPAMI.2024.3510932

Wang T, Wu J, Ji Z, Lai Y-K (2024) Sparse convolutional networks for surface reconstruction from noisy point clouds. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 3200–3209. https://doi.org/10.1109/WACV57701.2024.00318

Wang Z, Zhang Y, Xu R, Zhang F, Wang P-S, Chen S, Xin S, Wang W, Tu C (2023) Neural-Singular-Hessian: implicit neural representation of unoriented point clouds by enforcing singular Hessian. ACM Trans Graph 42(6):274:1–274:14. https://doi.org/10.1145/3618311

Zhan X, Zhang X, Wang X, Diao X, Qi L (2024) Comparative analysis of surface deformation monitoring in a mining area based on UAV-LiDAR and UAV photogrammetry. Photogramm Rec 39(186):373–391. https://doi.org/10.1111/phor.12490