Synthetic data generation for faults networks

Canning Anat, Vizeu Fernando, Zambrini Joao, Emerson*

Summary

Machine learning is transforming seismic interpretation, but its success depends on access to labeled training data - a resource that is often impossible to obtain. We present a methodology for generating synthetic 3D fault surface datasets that addresses this fundamental bottleneck. Point clouds with 6D attributes (x, y, z, probability, strike, dip) are generated on-the-fly, incorporating realistic geological complexity including curvature, irregular edges, discontinuities, and noise. Systematic randomization of all parameters, from grid dimensions and fault orientations to surface geometry and noise distributions, ensures comprehensive coverage of geological scenarios while eliminating the need for massive data storage. Although developed for fault surface extraction, the methodology serves broader AI-based applications such as seismic attribute generation, seismic image segmentation, foundation model training and more.

Introduction

In recent years, machine learning has transformed seismic interpretation, with deep learning models demonstrating remarkable success in automating seismic interpretation including fault detection, horizon tracking, salt body delineation (Huang et al., 2017; Gao et al., 2022) and more. These data-driven approaches promise significant gains in efficiency and consistency, replacing labour-intensive manual interpretation with automated systems. However, supervised learning models are fundamentally limited by the availability of labeled training data. With seismic data, ground truth lies buried in the subsurface and remains mostly unknown. Obtaining fault labels requires expert interpretation - a process that is expensive, time-consuming, and subjective. Different interpreters often produce inconsistent results, introducing human bias into any manually labeled dataset. Synthetic data generation offers a practical solution, enabling creation of unlimited training examples with perfect ground truth labels while introducing physical and geological constraints directly into the learning process (e.g., Gao et al., 2022 and Choi et al., 2025). Additionally, synthetic data can be generated on-the-fly during training, providing the diverse geological scenarios and systematic randomization critical for developing AI models that generalize well to real data, while eliminating massive data storage requirements.

The need for synthetic fault data spans multiple applications in seismic interpretation. ML-based fault attribute generation methods like FaultNet3D (Wu et al., 2019b) require training datasets with known fault locations and orientations. Foundation models for seismic interpretation, while primarily trained on un-labeled real data, increasingly incorporate complementary synthetic data to broaden the scope and diversity of training datasets (Dou et al., 2025). Full seismic simulation workflows require realistic fault geometries as input to their forward modeling engines to generate synthetic seismic volumes for various training objectives.

Automatic fault interpretation itself is typically divided into two stages: fault attribute generation and fault surface extraction. Fault attribute generation methods - both deterministic (Hale, 2013) and machine learning-based (Wu et al., 2019a) - process seismic volumes to produce three 3D attributes at each location: P (probability of being a fault), θ (local dip angle of the fault), and φ (local strike angle of the fault). These attributes provide rich geometric information about potential fault locations and orientations. Fault surface extraction then takes these attributes and constructs a fault network where each individual fault surface is uniquely identified (Wu and Hale, 2016; Dong et al., 2025).

We developed the methodology presented in this paper specifically for training a fault surface extraction algorithm described in Canning et al. (2026). That method uses a point cloud representation because the typical 3D fault attribute volume is very sparse - only a small fraction of samples have high fault probability - making volume-based techniques computationally wasteful. We extract high-probability points (above a probability threshold) after a fault thinning process (Wu and Hale, 2016), creating a sparse point cloud dataset (x, y, z, P, θ, φ). The Synthetic data generation process is designed to mimic such datasets while adding a “true label” (x, y, z, P, θ, φ, L) to allow for training.

This paper presents a methodology for generating synthetic 3D fault surface geometries as point clouds with 6-dimensional attributes (x, y, z, P, θ, φ) - directly simulating the output of fault attribute generation algorithms. The scope of this paper is limited to fault surface geometry generation; we do not address here the seismic simulation itself (for synthetic seismic data generation incorporating fault geometries, see Vizeu et al., 2022). Our approach incorporates geological fault complexity including 3D curvature (both along strike and dip directions), irregular surface edges, discontinuities and gaps, and random noise. Data is generated on-the-fly during training, producing hundreds to thousands of unique examples with comprehensive geological variability.

While developed for our specific surface extraction application, this methodology can serve multiple purposes: training fault attribute generation networks, providing input geometries for seismic forward modelling, benchmarking different fault interpretation algorithms, or any application requiring realistic 3D fault surfaces with known ground truth labels.

Method

The generation of synthetic fault datasets is designed to train neural networks to automatically interpret seismic data. The objective is pre-training models that can be applied to any seismic data, not customized to specific surveys. Therefore, we need to generalize across all aspects of fault surface characteristics as well as data acquisition parameters.

Each synthetic dataset contains multiple fault surfaces at randomly selected positions. Parameter ranges are first defined for all variables: grid dimensions, sampling intervals, fault positions, orientations, fault dimensions, curvature characteristics, and noise levels. A random set of parameters is then selected from these ranges for each dataset. Each fault surface is generated sequentially by integrating point-by-point from a central geometric midline. After all surfaces in the dataset are built, noise is added. Lastly, all points are snapped to grid positions with duplicates removed, to mimic the real seismic data structure. The following is the data generalization strategy:

Data Size and Sampling: Synthetic surfaces are generated on regular 3D grids to match different seismic acquisition geometries. Grid dimensions (*nx, ny, nz*) vary widely between surveys, and we randomly select these to cover a realistic range. Grid spacing exhibits anisotropy typical of seismic data – inline, crossline and vertical spacing are not necessarily equal and can differ significantly between surveys. All coordinates are normalized by a reference length to maintain consistent scaling across different grid configurations, enabling the model to learn scale-invariant features.

Spatial Distribution and Fault Geometry: In each dataset, several fault surfaces are randomly distributed in 3D space, with each surface defined by randomly selected center positions and geometric parameters. Strike angles are randomly selected from the full 0-360° range, while dip angles are randomly selected from ranges reflecting typical structural geology. To mimic realistic attribute estimation uncertainty, small random errors are introduced to the strike and dip values at each point, simulating the imperfect attribute predictions common in real seismic data. Surface dimensions vary randomly: lateral extent varies from small segments to surfaces spanning the entire domain, while width along the dip direction varies from narrow to broad features.

Parallel-Perpendicular Geological Preference*:* Real fault networks exhibit geological bias—faults often form in semi-parallel sets, conjugate pairs, or orthogonal systems. We replicate this by biasing each new surface's strike toward existing surfaces. A random preference is established for each dataset, weighting the probability of generating surfaces that are: parallel (same direction ± small deviation), perpendicular (90° apart), opposite (180° apart), or random. Strike deviations of 2-10° add natural variation to parallel sets.

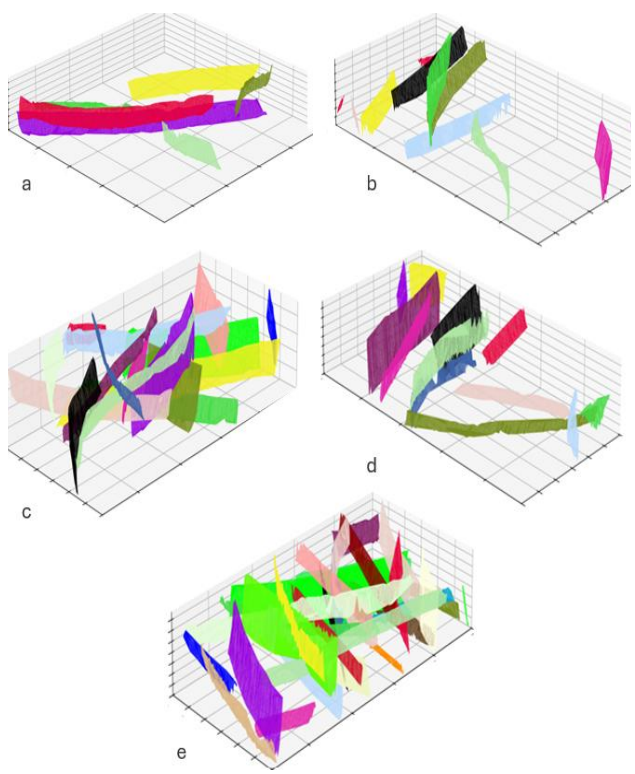

Figure 1: Several examples of synthetic fault networks. Each surface has a different label (color).

Curvature and Irregular Shape: Surfaces incorporate realistic 3D curvature through two mechanisms. Strike curvature introduces multiple bends along the horizontal direction, with bend positions distributed along the surface length and transition zones spanning several grid steps. Curvature is implemented using a parametric piecewise approach where bend angles are defined at specific positions and transitions occur sharply over ~15 grid steps, rather than using smooth analytic functions like splines. Dip curvature adds gentler bends along the vertical direction, representing steepening or shallowing. Cumulative bending is limited by proportional scaling: when total angular change exceeds geological limits, individual bend amplitudes are reduced to maintain plausible geometry. Surfaces are constructed by integrating point-by-point from a central spine, with local strike and dip updated at each step according to the curvature profile. Surface edges are made irregular through rugosity - an analytical boundary filter creates smooth, bumpy edges rather than rectangular boundaries, better representing typical seismic fault terminations.

Gaps and Discontinuities**:** Real fault surfaces often show gaps from erosion, limited seismic resolution, poor imaging quality or interpretation artifacts. We randomly add elliptical holes with perturbed boundaries creating irregular edges. Holes are preferentially placed near surface edges to create realistic fragmentation and isolated fault segments, mimicking the incomplete fault traces common in seismic data.

Random Noise: 3D blob-like clusters represent interpretation artifacts or unrelated geological features. These irregular ellipsoids with random strike/dip are generated with varying elongation and shape irregularity. Noise is generated in small blobs, simulating scattered misclassified points in real fault attribute data.

Most parameters are randomized independently to avoid introducing artificial correlations, except where geological constraints require coupling (e.g., curvature–extent limits or strike relationships). Randomization is applied systematically across all parameters - not just geometric properties but also grid spacing, fault counts, surface dimensions, curvature characteristics, hole sizes and positions, and noise levels. Each parameter is drawn from its defined range for every generated dataset, ensuring comprehensive coverage of the geological scenario space without repetitive patterns.

This generation strategy provides the network with diverse, realistic fault network scenarios covering the expected range of geological configurations. Figure 1 displays several examples of synthetic fault networks.

Examples

The synthetic data generation methodology produces highly diverse fault networks, as demonstrated in Figure 1. Each image shows a different randomly generated dataset, illustrating variations in grid dimensions, fault count, spatial distribution, surface orientations, curvature complexity, rugosity patterns, and discontinuities.

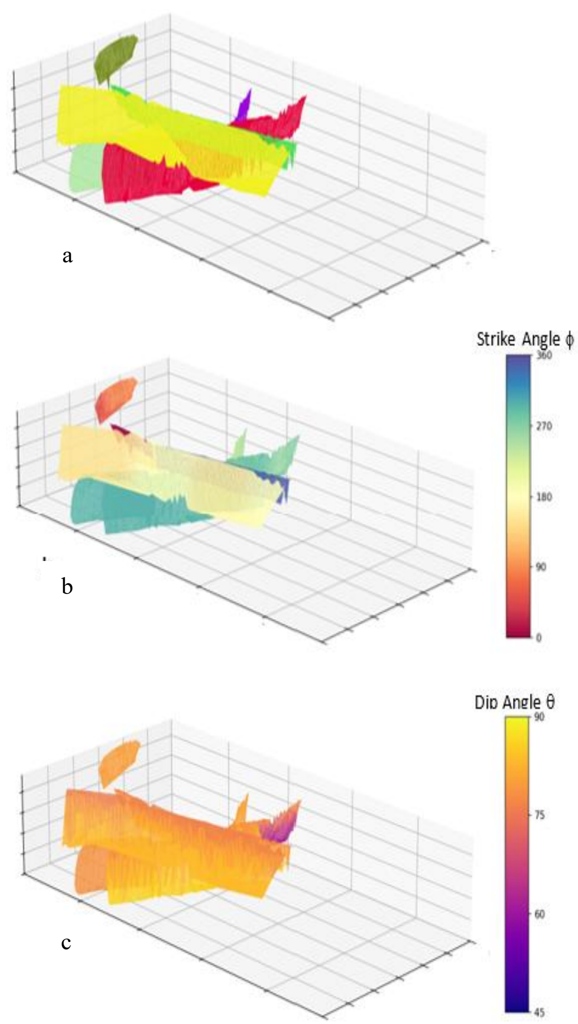

Figure 2: Example of an input data. Attributes, generated as part of the synthetic data generation process. a) fault network – color represent label, b) same dataset - color represents strike angle, c) color represents dip angle.

Figure 2 demonstrates that the synthetic engine generates complete multi-dimensional point cloud data (x, y, z, θ, φ, L). The same fault network is displayed with colors representing (a) fault surface labels, (b) local strike angles spanning the full 0-360° range, and (c) local dip angles. Note the spatial variability of strike and dip within individual surfaces, reflecting the 3D curvature incorporated into the generation process. These local orientation attributes are essential for training algorithms that leverage geological constraints during fault surface extraction.

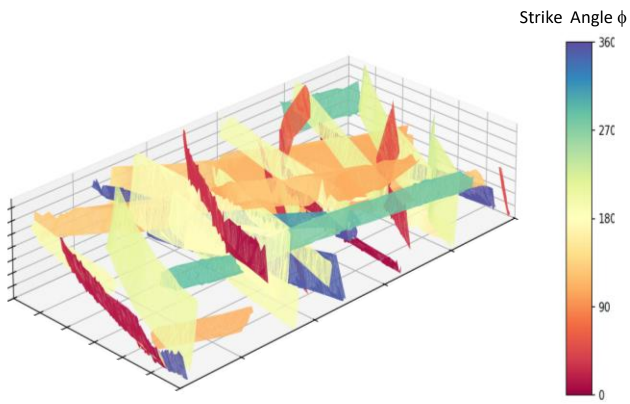

Figure 3: Detailed view of strike angle distribution (represented by color) for the model shown in Figure 1e,

Figure 3 provides a detailed view of strike angle distribution for the model shown in Figure 1e, illustrating the continuous variation of strike orientation across curved fault surfaces and between different faults in the network.

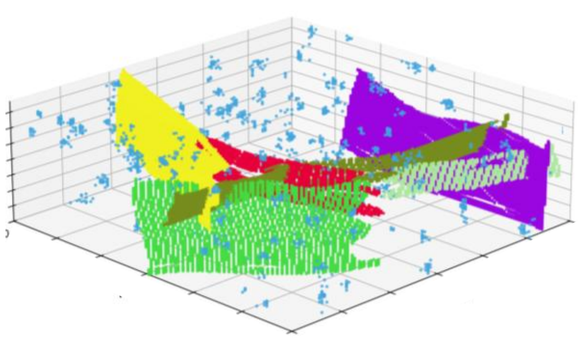

Figure 4: Example of multiple fault surfaces (colored) with randomly distributed noise points (blue label) scattered throughout the volume.

Figure 4 shows an example dataset with noise. Noise is labeled (-1) and represents interpretation artifacts or misclassified points that the model must learn to distinguish from actual fault surfaces.

These examples demonstrate the capability of the synthetic data engine to generate realistic, diverse training datasets on-the-fly, with each generated example exhibiting unique geometric characteristics while maintaining geological plausibility.

Conclusions

This methodology provides a practical solution to the labeled training data bottleneck in ML-based fault interpretation. By generating synthetic fault surfaces on-the-fly with comprehensive geological realism, we reduce the need for expensive manual labeling and ensure training datasets cover a large range of fault configurations. The synthetic data engine produces realistic, diverse examples with each dataset exhibiting unique geometric characteristics and maintaining geological plausibility. Developed specifically for training point cloud-based fault surface extraction algorithms, the approach can also serve multiple applications including training fault attribute generation networks, providing geometries for seismic forward modeling, benchmarking interpretation algorithms and more.