remote sensing

Article

Improving Seismic Fault Recognition with Self-Supervised Pre-Training: A Study of 3D Transformer-Based with Multi-Scale Decoding and Fusion

Zeren Zhang [†], **Ran Chen** [†] **and Jinwen Ma** [*]

Department of Information and Computational Sciences, School of Mathematical Sciences, Peking University, Beijing 100871, China; eric_zhang@stu.pku.edu.cn (Z.Z.); chenran@stu.pku.edu.cn (R.C.)

[*] Correspondence: jwma@math.pku.edu.cn

[†] These authors contributed equally to this work.

Abstract: Seismic fault interpretation holds great significance in the fields of geophysics and geology. However, conventional methods of seismic fault recognition encounter various issues. For example, models trained on synthetic data often exhibit inadequate generalization when applied to field seismic data, and supervised learning is heavily dependent on the quantity and quality of annotated data, being susceptible to the subjectivity of interpreters. To address these challenges, we propose applying self-supervised pre-training methods to seismic fault recognition, exploring the transfer of 3D Transformer-based backbone networks and different pre-training methods on fault recognition tasks, thereby enabling the model to learn more powerful feature representations from extensive unlabeled datasets. Additionally, we propose an innovative pre-training strategy for the entire segmentation network based on the characteristics of seismic data and introduce a multi-scale decoding and fusion module that significantly improves recognition accuracy. Specifically, during the pre-training stage, we compare various self-supervision methods, like MAE, SimMIM, SimCLR, and a joint self-supervised learning approach. We adopt multi-scale decoding step-by-step fitting expansion targets during the fine-tuning stage. Ultimately merging features to refine fault edges, the model displays superior adaptability when handling narrow, elongated, and unevenly distributed fault annotations. Experiments demonstrate that our proposed method achieves state-of-the-art performance on Thebe, the currently largest publicly annotated dataset in this field.

Keywords: seismic faults detection; 3D segmentation; self-supervised; pre-training

check for updates

Citation: Zhang, Z.; Chen, R.; Ma, J. Improving Seismic Fault Recognition with Self-Supervised Pre-Training: A Study of 3D Transformer-Based with Multi-Scale Decoding and Fusion. *Remote Sens.* **2024**, *16*, 922. https://doi.org/10.3390/rs16050922

Academic Editor: Shuying Li

Received: 4 February 2024
Revised: 28 February 2024
Accepted: 4 March 2024
Published: 6 March 2024

1. Introduction

1.1. Background

Seismic data obtained through reflection surveys represent an indirect depiction of the intricate physical structures within the Earth's subsurface [1]. Three-dimensional seismic data offer a more detailed imaging and complex mapping of sub-surface structures than 2D because of denser sampling [2]. In 3D seismic images, geological strata create consistent reflections, whereas faults, due to brittle deformation, cause reflection discontinuities [3,4]. This contrast enables identifying and extracting geological features like strata and faults from the data. Seismic fault interpretation is crucial in geophysics and geology for tasks like stratigraphic analysis, examining ancient landscapes, reservoir characterization, well planning, resource assessment, and geological hazard risk mitigation [4–8].

Traditionally, manual and experiential seismic fault interpretation is susceptible to the influence of data quality and interpreter subjectivity [1,9]. Moreover, in large 3D seismic images, identifying stratum and fault mapping is a time-consuming and labor-intensive process [3]. Various automated fault recognition methods are proposed in response to these limitations, including phase unwrapping [10,11], ant-tracking [12,13], waveform classification [14], slope techniques [15], and Hough transform [16]. However, most feature-based methods impose high computational costs and display susceptibility to noise [1]. These methods often require manual adjustment and heuristic parameter tuning to adapt to different datasets.

The emergence of machine learning [17] has spurred the advancement of intelligent seismic fault recognition algorithms [18,19]. End-to-end fault recognition methods based on convolutional neural networks (CNNs) frame fault recognition as a classification or segmentation task [20–22]. Wu et al. [22] employed synthetic data to train an end-to-end 3D-Unet network named FaultSeg3D for fault recognition tasks. Furthermore, An et al. [23] utilized a 2D segmentation network, Mobile DeepLabV3+, for per-pixel segmentation to extract faults and introduced Thebe, the largest currently publicly available field seismic fault dataset. However, existing models trained on small synthetic datasets demonstrate performance degradation and limited generalization when applied to field seismic data. This is due to the heavy reliance on supervised deep learning on accurate annotations. Consequently, these models need improvement in dealing with data with significant feature variations, quality issues, high annotation costs, and subjectivity.

1.2. Related Work

Transformer [24] models, which rely on attention mechanisms, have advantages over CNNs in capturing global context information and obtaining full-scale features. Self-supervised learning methods using the Transformer structure, such as Vision Transformer (ViT) [25], show excellent performance across various visual tasks. Additionally, the Swin-Transformer [26], optimized through hierarchical and sliding window strategies, further enhances the performance of the ViT. On the other hand, masked autoencoders (MAEs) [27] demonstrate commendable image understanding capabilities through self-supervised learning. The simplified mask image modeling (SimMIM) [28], tailored specifically for the Swin-Transformer, uses a straightforward masked image modeling (MIM) approach, a lightweight decoder, and a more adaptable token processing mechanism, making it well-suited for downstream tasks requiring multi-scale modeling. As a self-supervised technique that learns similarities in data through contrastive learning, a simple framework for contrastive learning of representations (SimCLR) [29] also demonstrates significant competitiveness in image understanding tasks.

Seismic data processing tasks are now increasingly utilizing self-supervised learning techniques, such as seismic velocity inversion, stratigraphic phase semantic segmentation, seismic data denoising, etc. [30–32], which demonstrates the applicability of self-supervised pre-training methods in seismic data feature extraction. However, only a few studies apply the Transformer framework to fault recognition tasks. Yang et al. [33] presented a multitask learning network capable of simultaneous stratigraphy extraction and fault detection. They utilized Transformer architecture for stratigraphy estimation, fortifying the model's robustness with expert geological interpretation. Tang et al. [34] proposed a novel method that amalgamates the Unet architecture with a Transformer encoder for 2.5D fault detection. The model, trained on synthetic datasets, demonstrates superior performance on the Netherlands F3 dataset [35] compared to the complete 3D Unet model. The Dual Unet with Transformer model proposed by Wang et al. [36] combines the traditional U-Net with the Transformer U-Net, and they found that the binary cross-entropy loss (BCE) performed best after comparing six different loss functions. Finally, FaultSSL [37] is a semi-supervised fault recognition framework. Its supervised learning component uses synthetic data and a small number of 2D label data, while the unsupervised learning component relies on two proxy tasks—PaNning Consistency (PNC) and PaTching Consistency (PTC). The approach reduces the reliance on synthetic data and purely supervised learning methods but still needs a certain amount of labeled data support. In summary, recent research has made progress in applying the Transformer architecture to solve fault recognition problems. However, training on synthetic datasets limits the generalization of field datasets.

Although the potential advantages of semi-supervised learning have been preliminarily explored, more quantitative evaluations are still needed.

1.3. Motivation

As previously discussed, seismic fault recognition technology has evolved significantly, transitioning from traditional manual interpretation to advanced deep learning models. Despite these advancements, supervised learning models, heavily dependent on accurate labeling, exhibit limitations when processing field seismic data. In response to these limitations, self-supervised learning has become a powerful solution to compensate for the limitations of supervised learning. However, applying the Transformer framework in fault recognition tasks and using self-supervised learning methods within this context are still largely undeveloped, and the resulting performance has yet to reach the expected level. Consequently, this study explores the potential gains of using self-supervised pre-training methods in seismic fault recognition. We adopt a 3D Transformer-based backbone network, and during the pre-training phase, we conduct a detailed exploration of various self-supervised pre-training techniques, including MAE, SimMIM, and SimCLR, allowing the model to learn more robust feature representations from large-scale unlabeled data. Our findings reveal that combining the Swin UNEt TRansformer (Swin-UNETR) [38] backbone with the SimMIM pre-training task significantly enhances seismic fault recognition capabilities. We specially design our model architecture to handle the complexity inherent in seismic fault recognition patterns. Considering the sparse distribution features of seismic fault data, we innovatively improve the Swin-UNETR architecture, realizing multi-scale decoding and fusion. Furthermore, our findings reveal that the Swin-UNETR model's decoder has significantly more parameters than its backbone network, which suggests that pre-training the entire segmentation model can further enhance fault detection accuracy. Our method, referred to as FaultSeg Swin-UNETR, demonstrates excellent adaptability and precision in detecting seismic faults.

2. Datasets

In their comprehensive 2023 review on fault recognition [39], An and colleagues synthesized information from 73 seismic datasets, of which only three field datasets and four synthetic datasets open-sourced seismic data and labels, providing a public baseline for research. Only two open-sourced datasets with annotations are 3D, including the synthetic dataset FaultSeg [22] created by Wu and the field dataset Thebe [23] collected by An's team. FaultSeg simulates seismic signal patterns, providing detailed explanations and opening its dataset and code. Although synthetic datasets can partially reduce the reliance on expert annotations, the quality difference of synthetic data significantly influences the model's performance when dealing with practical data. When choosing the dataset type, researchers must consider that the ultimate purpose of all models is to solve real earthquake data-related problems. Hence, testing the model's performance through field earthquake datasets is a straightforward approach to gauging its performance. Currently, the Thebe dataset is the largest publicly available earthquake fault dataset, providing many detailed pixel-level expert annotations, allowing researchers to compare the performance of different models more accurately and identify their advantages and limitations.

2.1. Datasets Employed

Earthquake fault recognition models trained with data from actual work areas can fully understand and handle the complexity and uniqueness of field earthquake data. Such models have better generalization abilities than models trained on simulated datasets. They can adapt to varying conditions and maintain high accuracy even when encountering previously unseen data. Therefore, considering the practical applicability of seismic fault recognition models, our approach primarily relies on seismic data obtained from actual working areas. Specifically, we use a large number of unlabeled private data for pre-training.

Then, we fine-tune it on the labeled field dataset Thebe and verify the model's effectiveness on both the publicly available synthetic dataset FaultSeg and the field dataset Thebe.

- Synthetic dataset
  A synthetic dataset was used in the earliest work using deep learning methods for fault recognition [22]. This dataset consists of 220 seismic volumes of 128 × 128 × 128 and corresponding fault labels. The training set slice images are depicted in Figure 1.

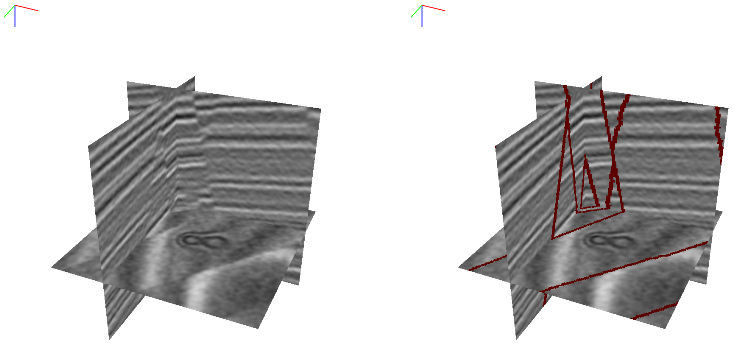


Figure 1. Three-dimensional visual examples of the synthetic dataset. A synthetic seismic image of the synthetic dataset and the corresponding true fault image (with labeling ones on faults and zeros elsewhere) is overlaid with the seismic image.

- Field dataset
  Most relevant works on fault recognition are trained on this synthetic dataset and then qualitatively analyzed on field fault datasets. However, deep learning methods typically require much training data to mitigate overfitting. Deep networks are prone to learning patterns specific to the synthetic data, which may not apply to real-world scenarios. Therefore, during the fine-tuning process, we conducted significant experiments using Thebe [40], the currently largest publicly available seismic faults dataset. The dataset, originally from a seismic survey called Thebe Gas Field in the Exmouth Plateau of the Carnarvan Basin on the NW shelf of Australia, is represented in Python Numpy format. The Thebe dataset has a size of 1803 × 1537 × 3174. Because of the high correlation between adjacent slices, random partitioning along the crossline or inline direction was not a reasonable choice. Following the approach proposed by An et al. [40], we divided the data along the crossline direction. The first 900 slices were used for training, the next 200 for validation, and the remaining 703 for testing. The 640 × 640 × 640 cube segmented from the Thebe dataset is depicted in Figure 2. A comparison between Figures 1 and 2 reveals that the field seismic data are comparatively more complex than the synthetic seismic data. The field seismic dataset exhibits a more intricate fault distribution and finer fault annotation lines, making transferring the model trained on synthetic data to field data challenging.
- Pre-training dataset
  In addition, as self-supervised pre-training does not require annotations, we collected private datasets from 15 different working areas for pre-training tasks. These datasets are diverse in their geological features and size, and notably, they all lack fault annotations. Our goal was for the network to independently discover and learn the intrinsic characteristics of fault data across various contexts from various working areas during the self-supervised learning phase. We visualize some of the data used for pre-training in Figure 3.

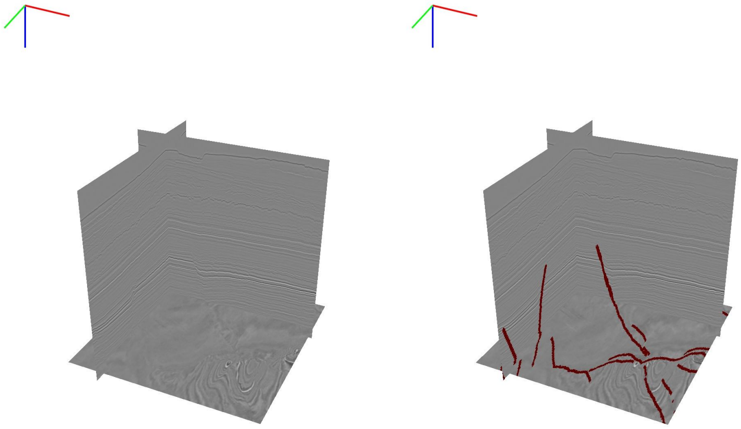


Figure 2. Three-dimensional visual examples of the Thebe dataset. A synthetic seismic image is cropped from the Thebe dataset, and the corresponding true fault image (with labeling ones on faults and zeros elsewhere) is overlaid with the cropped seismic image.

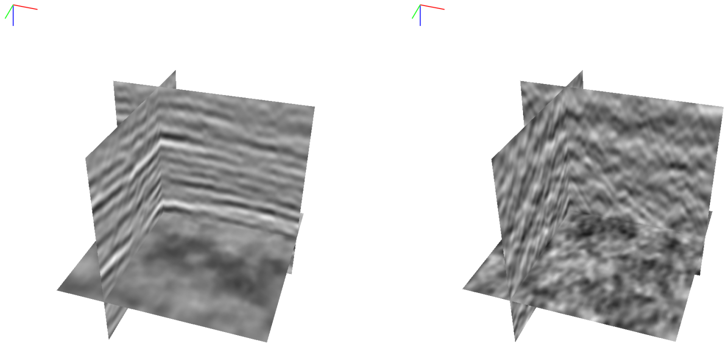


Figure 3. Three-dimensional visual examples of the pre-training dataset. The figure illustrates 128 × 128 × 128 segmentations from the pre-training datasets of various working areas.

2.2. Data Processing

To ensure consistency during training and comparability during predictions of seismic data, we initially performed z-score standardization on the data. Assuming S represents the original three-dimensional seismic data and μ and σ respectively denote its mean and standard deviation, we utilized Equation (1) to standardize the original data.

$$S_{\text{standardized}} = \frac{S - \mu}{\sigma} \tag{1}$$

These preprocessing steps prevent data from being overly concentrated or dispersed, mitigating the influence of varying feature scales. This enhances the stability and convergence speed of the model, facilitating a more straightforward acquisition of fault distribution characteristics. The distributions of standardized fault data and annotated data are illustrated in Figure 4. The Thebe dataset is publicly available. Expert interpreters from the Fault Analysis Group at University College Dublin manually labeled the faults in the seismic data. Because of the slanted nature of the work area, preprocessing involved creating a bounding rectangle to form a 3D cube and filling in blank areas with zeros. This resulted in a higher kurtosis for the Thebe dataset than synthetic and field data typically used for pre-training, which usually resemble a normal distribution. The field seismic cube shows skewness, with many zeros and near-zero values, making its distribution notably different from the synthetic and pre-training data. Therefore, fine-tuning post-pre-training on unlabeled datasets was crucial to addressing these distributional discrepancies.

Figure 5a is a histogram showing the statistical distribution of 0–1 fault labels for 10,000 data points randomly sampled from the Thebe and synthetic datasets; on the x-axis, False represents non-fault points, i.e., points with a label of 0, while True represents fault points, i.e., points with a label of 1. The y-axis represents the total number of voxel points with a label of 0 or 1 in different datasets. It reveals substantial label imbalance, where fault annotations are significantly fewer than non-fault annotations. The field dataset, in particular, exhibits a more pronounced imbalance, making fault interpretation tasks more challenging. Consequently, models trained on synthetic datasets demonstrate limited feature extraction capabilities when applied to field datasets. Figure 5b displays the distribution of fault and non-fault points after merging the Thebe and synthetic datasets. The distinct difference in fault distribution between the synthetic and the Thebe datasets highlights the crucial significance of the model's simultaneous learning to identify seismic fault and non-fault features.

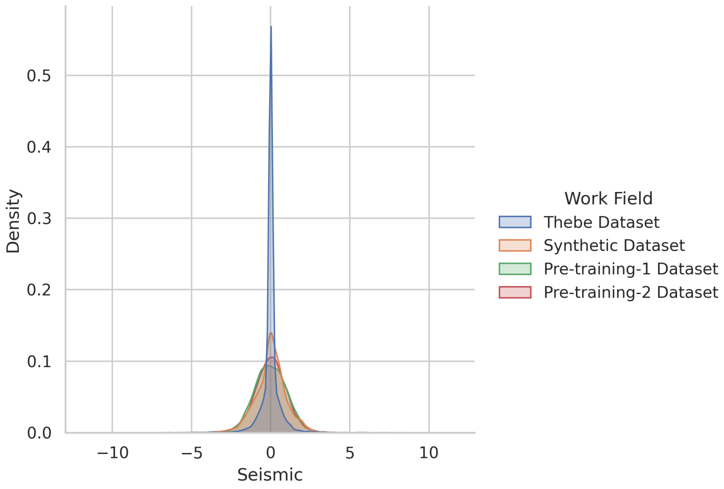
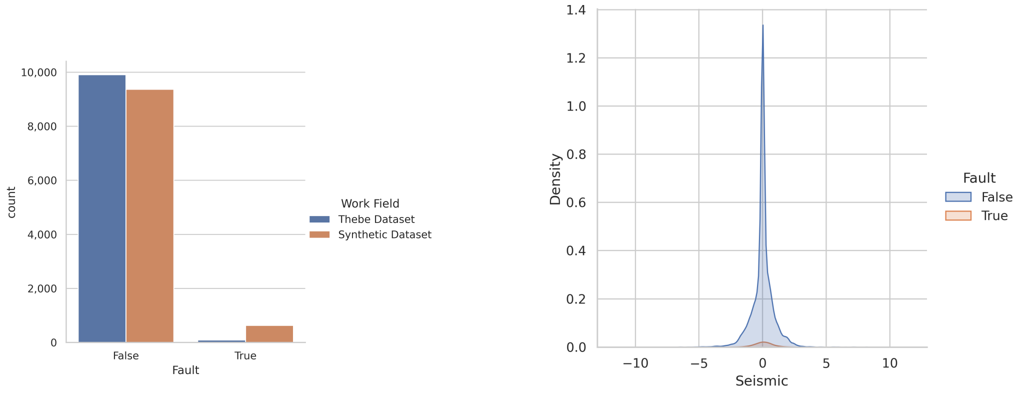


Figure 4. The amplitude distribution of seismic datasets after z-score standardization.



(**a**) (**b**)

Figure 5. (**a**) The distribution of labels in synthetic seismic data and field seismic data exhibiting a 0–1 distribution. (**b**) Labels distribution of faults and non-faults after merging synthetic and field datasets.

Figure 6 is a boxplot of the data distribution. The values on the vertical axis represent the seismic data values after z-score normalization. The False part on the left shows the distribution of non-fault data for Thebe and synthetic data, and the True part on the right shows fault data. The distribution of non-fault data in Thebe on the far left even loses the shape of the box because of the over-concentration of zero values and includes many outliers, in which the maximum outlier even reaches a positive 12.5. However, the synthetic data values are evenly distributed on both sides of the 0 axis after standardization. The annotation of fault data is relatively uniform, with both Thebe, in blue, and the synthetic data, in orange, being standard and symmetrical. The boxplot vividly illustrates the highly concentrated distribution of non-fault points within the Thebe dataset and the considerable presence of outliers. This challenges the model, potentially leading to biases toward specific features or regions during training, neglecting other crucial information. Such biases can impede the model's comprehensive understanding of geological structures.

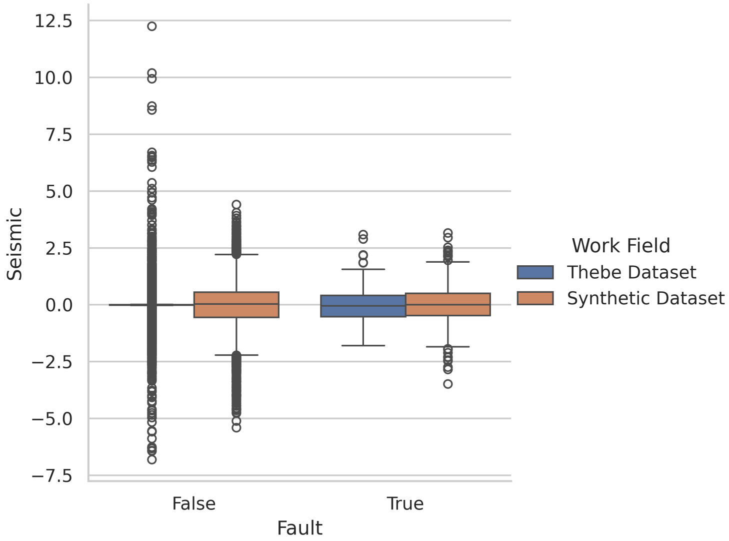


Figure 6. The boxplots of fault and non-fault data in the Thebe dataset and synthetic dataset.

Additionally, random flipping, 90-degree random rotations, and random cropping are common data augmentation techniques in the computer vision domain. They expand the training dataset, enabling the model to learn diverse features better. Random cropping effectively extracts feature information of different scales and positions, bolstering the model's generalization ability. Random flipping and 90-degree random rotations aid the model in learning features from various orientations, enhancing its robustness. This combination of techniques augments the model's capability to extract seismic data features and generalizability, bolstering its effectiveness and stability in real-world applications.

3. Seismic Fault Recognition with Self-Supervised Pre-Training

Inspired by successful self-supervised pre-training schemes in image processing, we explored various pre-training tasks on a large volume of unlabeled seismic field data using 3D Transformer-based backbones. These backbones include U-Net TRansformers (UNETR) [41] and Swin-UNETR, with pre-training strategies comprising MAE, SimMIM, SimCLR, as well as three joint self-supervised tasks used in [38]: masked volume inpainting, image rotation, and contrastive coding. Following this pre-training phase, the network learned rich features within the seismic domain. We fine-tuned and tested it on field datasets, discovering that the strategy combining Swin-UNETR with SimMIM yields the best results. Building on this insight, to address challenges such as the scarcity of seismic fault data and the narrow and uneven distribution of seismic faults, we further refined the Swin-UNETR model. We propose a full-network pre-training scheme named FaultSeg Swin-UNETR specifically for seismic fault segmentation tasks. We observe a notable disparity between the size of parameters in the backbone network and the decoder head of Swin-UNETR. To address this, we employed full-network self-supervised pre-training with SimMIM, which achieved better outcomes. Additionally, we found that introducing multi-scale decoding and fusion modules significantly enhanced identification precision. Section 3.1 introduces the backbone network and pre-training schemes, while Section 3.2 discusses the full-network pre-training of FaultSeg Swin-UNETR.

3.1. Backbone and Pre-Training Methods

3.1.1. 3D Transformer-Based Backbone

Our research uses a Transformer-based model, different from many past studies that used CNNs [23,42], for fault recognition in seismic images. It is adept at capturing multi-scale contexts and managing tasks that require understanding long-distance relationships, which is crucial for segmenting images by pixel relationships. Because of the difficulty of annotating seismic faults, it is common for researchers to annotate faults slice by slice along a specific direction (inline or crossline) within a particular working area. Therefore, previous studies often used 2D segmentation networks for fault recognition, but these could not capture the full spatial context of seismic faults. While 2.5D data have been employed to address this partially, proper integration of 3D spatial information necessitates a 3D segmentation network. Therefore, our model uses a 3D Transformer backbone as the main network.

UNETR and Swin-UNETR [38], as classic 3D Transformer-based frameworks, redefine the task of 3D cube fault recognition as a sequence-to-sequence prediction problem. These architectures divide the input 3D seismic image into volumetric patches. Each patch is linearly embedded and then processed through a series of Transformer or Swin-Transformer blocks, enabling the models to learn rich, context-aware data representations. UNETR combines the architecture of U-Net with the advantages of Transformers and typically features an encoder–decoder structure: the encoder utilizes Transformer blocks to capture global context information, while the decoder employs conventional convolutional layers to reconstruct the segmentation output of seismic faults intricately. Swin-UNETR builds upon the UNETR concept but incorporates the Swin-Transformer as its backbone. "Swin" in Swin-Transformer stands for "shifted window", a novel attention mechanism that adopts a window-based approach to handle local information in images better. As shown in Figure 7, the Swin-Transformer consists of multiple Swin blocks. In Swin-UNETR, the Swin-Transformer's shifted windowing approach is adapted to handle 3D images, allowing the model to dynamically adjust its focus between smaller, local areas and the broader global context. This mechanism provides a more nuanced understanding of the spatial relationships within volumetric data, which is crucial for accurate segmentation and analysis in seismic imaging. For more details about Swin-Transformer, please refer to Appendix A.

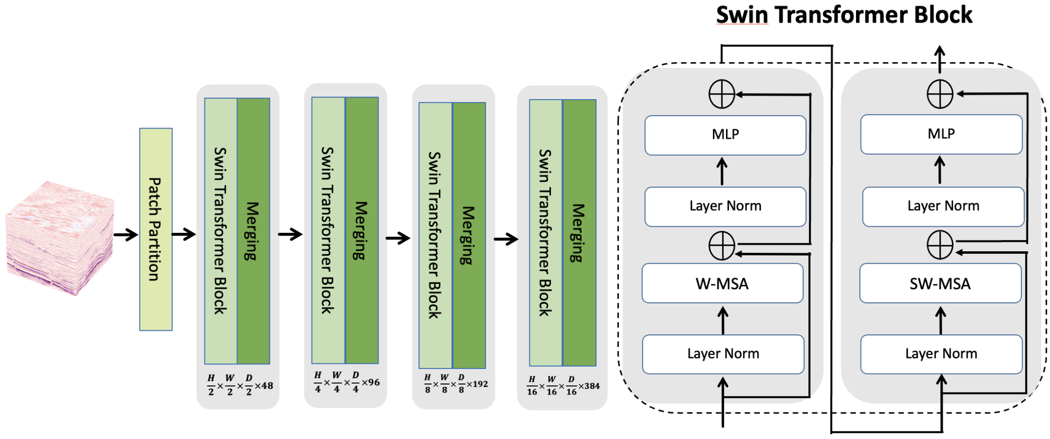

Swin Transformer Block
Patch Partition
Swin Transformer Block
Merging
$\frac{H}{2}\times\frac{W}{2}\times\frac{D}{2}\times48$
Swin Transformer Block
Merging
$\frac{H}{4}\times\frac{W}{4}\times\frac{D}{4}\times96$
Swin Transformer Block
Merging
$\frac{H}{8}\times\frac{W}{8}\times\frac{D}{8}\times192$
Swin Transformer Block
Merging
$\frac{H}{16}\times\frac{W}{16}\times\frac{D}{16}\times384$
MLP
Layer Norm
W-MSA
Layer Norm
MLP
Layer Norm
SW-MSA
Layer Norm

Figure 7. Architecture of 3D Swin-Transformer backbone.

3.1.2. Self-Supervised Pre-Training of Seismic Data

We collected many unlabeled field seismic data, covering many scenarios. In order to explore the potential information within seismic data, we compare several mainstream self-supervised pre-training methods, including MAE, SimMIM, SimCLR, and the masked volume inpainting, image rotation, and contrastive coding implemented within the Swin-UNETR architecture. MAE and SimMIM focus on improving the model's prediction capabilities for masked parts within the data, thereby strengthening the model's overall understanding of seismic data. Meanwhile, SimCLR enhances the model's ability to discriminate between different seismic features via contrastive learning. Masked volume inpainting trains the model on how to infer the overall structure from partially visible seismic data; image rotation compels the model to learn the direction invariance of seismic images; contrastive coding encourages the model to identify and understand the differences and similarities within different parts of the images. Our goal was to comprehensively evaluate the applicability of each method in the field of earthquakes to determine which method performs better on seismic datasets.

3.2. FaultSeg Swin-UNETR

This research introduces FaultSeg Swin-UNETR, a self-supervised pre-training model for seismic fault segmentation based on Transformer. This model innovatively combines the Swin-UNETR architecture with the SimMIM pre-training method, optimizes the uneven distribution of parameters through full-network pre-training, and incorporates multi-scale decoding fusion modules to adapt to the peculiarities of seismic data.

3.2.1. Full-Network Pre-Training

Our findings highlight that the Swin-UNETR backbone, particularly when combined with the SimMIM pre-training strategy, outperforms traditional tasks like masked volume inpainting, image rotation, and contrastive coding in terms of earthquake fault recognition capability. This empirical evidence led us to adopt the Swin-UNETR backbone in conjunction with the SimMIM pre-training approach in our proposed FaultSeg Swin-UNETR method.

Figure 8 illustrates the workflow of using SimMIM in seismic data. Given a fixed size $(128 \times 128 \times 128)$ seismic dataset, it was first divided into 262,144 non-overlapping patches of size $2 \times 2 \times 2$. After specifying a mask ratio (usually 75%), a random portion of the seismic data was masked from these 262,144 patches to serve as the input for the backbone network, Swin-Transformer. However, it is not reasonable to randomly mask overly dense patches, as it would make the reconstruction task more similar to a super-resolution task. Therefore, we adopted the strategy of simulating larger patch sizes in ViT, where during random masking, we simulated the effect of $16 \times 16 \times 16$ large patches on $2 \times 2 \times 2$ small patches. In the specific implementation, after the image was divided into patches, it could go through a tokenization module to convert it into tokens inputted to the Transformer. The masking operation was then performed on these tokens (Figure 8 optional branch). Consider the 3D seismic dataset represented by S. The formulation for the masking operation can be elegantly expressed as Equation (2).

$$\text{MaskedInput} = \text{PE}(\text{Tokenization}(S) \odot \mathcal{M} + \text{Repeat}(\text{MaskToken}) \odot (1 - \mathcal{M})) \tag{2}$$

Herein, $\text{Tokenization}(\cdot)$ represents the operator that segments the input seismic data into blocks and then projects them into tokens that can be accepted by the Swin-Transformer. $\text{PE}(\cdot)$ denotes the application of positional embeddings to introduce spatial awareness. The term $\mathcal{M}$ represents a masking matrix, which is designed to obscure a random 75% of the input tokens, thereby facilitating a robust learning process. MaskToken, a tunable parameter, serves to supplant the elements obscured by the mask, thus preserving the integrity of the data's structural composition. After inputting the masked tokens into the Transformer, they went through its internal self-attention mechanism. Afterward, we obtained a feature map with a smaller spatial size at the output. We needed a simple upsampling network (composed of transpose convolutions) to upsample it to the same size as the input. Then, through the same patch-based masking operation, we calculated the l_1 loss between the masked regions and the original input. Despite the difficulty of annotating earthquake data, collecting it was relatively easier. This made it possible to perform SimMIM on many unlabeled seismic data. Compared to using pre-trained backbone networks for supervised tasks such as classification, segmentation, and object detection on natural scene data (such as Imagenet, COCO), directly using earthquake data for self-supervised learning could learn the essential features of the seismic domain, making it easier and more accurate in the downstream fine-tuning process.

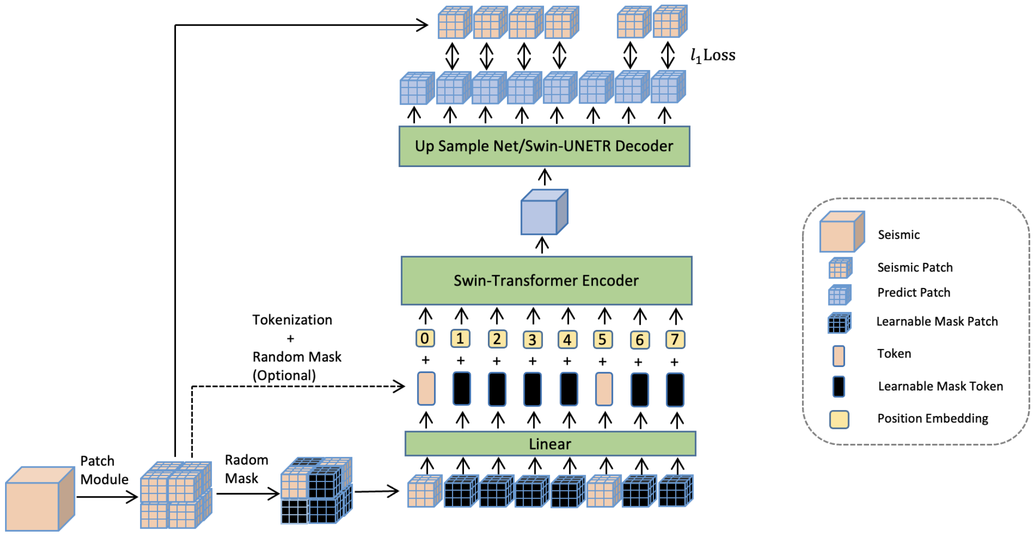


Figure 8. Overview of simplified mask image modeling.

Furthermore, we found a significant disparity in parameter proportion between the backbone network and the segmentation decoder network in the Swin-UNETR network used for seismic fault segmentation. This mainly arose from using the self-attention mechanism in the Swin-Transformer, resulting in far fewer parameters than the 3D convolutional layers in the segmentation decoder. Specifically, the parameter size of the Swin-Transformer backbone network is only one-eighth of the total parameter size of the segmentation network (see Table 1). This design contradicts the ideas of many self-supervised pre-training models. Generally, in self-supervised pre-training frameworks, the parameter size of the backbone network is much larger than that of the decoding heads of downstream tasks. The backbone network mainly accomplishes the learning of input data features, and it is preferable to have a simple design for the decoding heads. In light of these observations, we propose an innovative shift in strategy. Rather than confining the self-supervised learning task, such as SimMIM, to merely a network component, we advocate for utilizing the entire Swin-UNETR structure, including both the backbone and the segmentation head, for self-supervised pre-training. This method is particularly suitable for the inherently image-to-image nature of most seismic-related downstream tasks. To some extent, it can solve the problem of unreasonable parameter settings when Swin-UNETR integrates the Unit architecture. The original self-supervised pre-training method of Swin-UNETR only focuses on the model's backbone and needs to fully explore the potential of unlabeled data. Our training strategy not only allows the decoder to learn prior information from a large number of unlabeled data during the pre-training stage but also ensures that this pre-training information can be fully utilized in FaultSeg Swin-UNETR through the parameter replication of the three decoder heads downstream tasks.

Consequently, we engaged in extensive pre-training of the complete Swin-UNETR network on a vast corpus of seismic fault data. In response to this situation, we made certain modifications to the input processing. We performed patching and masking operations directly on the seismic data at the input stage, in contrast to the typical approach of manipulating tokens post the patch embedding layer. This reimagined strategy addresses the parameter distribution challenge and aligns more congruently with the intrinsic requirements of seismic fault recognition tasks. This direct masking operation on the input seismic

S can be described by Equation (3), where $F_{\text{patch}}(\cdot)$ denotes the operation of segmenting the input into patches and $\mathcal{M}$ represents the mask matrix.

$$\text{MaskedInput} = F_{\text{patch}}(S) \odot \mathcal{M} + \text{Repeat}(\text{MaskPatch}) \odot (1 - \mathcal{M}). \tag{3}$$

Additionally, the dimension of MaskPatch is also smaller than that of MaskToken. This approach is more in line with the original design of SimMIM. It allows the backbone network and decoder to learn some essential features from unlabeled data during the pre-training phase. The goal of the SimMIM pre-training task is to reconstruct masked seismic data. The learning objective for the task is

$$\mathcal{L}_{SimMIM} = \|\mathcal{M} \odot (S_{reconstruct} - S)\|_1. \tag{4}$$

During the pre-training phase, the network starts learning from weights initialized randomly. The chosen optimizer is AdamW, and a cosine annealing learning rate decay strategy is employed.

Table 1. Parameters of the Swin-Transformer-based model.

Model Name	Backbone	Decode Head	Total Params
Swin-Transformer + Up Sample Net	7.8 M	3.1 M	10.9 M
Swin-UNETR	7.8 M	54.1 M	61.9 M
FaultSeg Swin-UNETR	7.8 M	93.3 M	101.1 M

3.2.2. Multi-Scale Decoding and Fusion Module

Figure 9 illustrates the segmentation network FaultSeg Swin-UNETR that we designed for fault recognition. Based on our findings, earthquake faults are typically narrow and account for a deficient proportion, resulting in a serious class imbalance issue in training targets. In order to alleviate this phenomenon, we use multiple segmentation decoders to learn from fault segments that have been expanded at different levels. Specifically, let us assume our annotated samples are $\{S_j, F_j\}_{j=1}^{N}$, where $S_j \in R^{128\times128\times128}$ represents the input seismic and $F_j \in \{0,1\}^{128\times128\times128}$ represents the corresponding labeled faults. The Swin-Transformer encodes the input seismic and then decodes it separately by three segmentation decoders. These three decoders need to learn seismic faults that have been expanded by $1x$, $3x$, and $5x$, respectively. Our main loss function is defined as Equation (5), where $\text{Dilate}(F_j, 2i-1)$ represents the j-th labeled faults F_j after being dilated $2i-1$ times using the dilation operator, and $\text{Decode}_{i,j} = \text{Decoder}_\text{i}(\text{Backbone}(S_j))$ represents the predicted dilated fault result of seismic sample S_i after passing through the j-th decoder of FaultSeg Swin-UNETR. We conceptualize fault identification as a per-pixel binary classification problem for 3D seismic data, thereby configuring the loss function as binary cross-entropy (BCE) loss. The design of the main loss function is employed to supervise the three decoding branches of our FaultSeg Swin-UNETR, enabling them to independently learn the annotated fault information after being dilated at three different scales. Afterward, to restore the original fault body, we introduce a fusion module that re-calibrates the fault bodies learned from the three branches to their original thickness. The fusion module consists of a series of 3D convolutions. The three predictions of dilated faults $\{\text{Decode}_{i,j}\}_{i=1}^{3}$ are concatenated along the channel dimension and passed through the fusion module. We define the fusion loss function as Equation (6), where $\text{Fusion}(\cdot)$ denotes the fusion layer we designed, and $\text{Cat}(\cdot)$ represents the concatenation operator. Given that the decoded outputs $\text{Decode}_{i,j} \in \mathbb{R}^{H\times W\times D}$ are three-dimensional, we extend an additional dimension to the predicted results $\{\text{Decode}_{i,j}\}_{i=1}^{3}$ from the three decoding heads and concatenate them along this new dimension, resulting in $\text{Cat}(Decode_{1,j}, Decode_{2,j}, Decode_{3,j}) \in \mathbb{R}^{H\times W\times D\times 3}$. Since the fusion module is designed solely to learn how to restore the expanded fault body during gradient backpropagation, we disconnect it from the main segmentation network. The fusion loss function $\mathcal{L}_{fusion}$ does not participate in the updates of the main network. The total loss is composed of a weighted combination of the losses from two parts (Equation (7)), where λ is used to adjust the strength of the supervisory signal between the backbone network of FaultSeg Swin-UNETR and its fusion module.

$$\mathcal{L}_{main} = \frac{1}{3N}\sum_{i=1}^{3}\sum_{j=1}^{N}\text{BCE}(\text{Decode}_{i,j}, \text{Dilate}(F_j, 2i-1)) \tag{5}$$

$$\mathcal{L}_{fusion} = \frac{1}{N}\sum_{j=1}^{N}\text{BCE}(\text{Fusion}(\text{Cat}(Decode_{1,j}, Decode_{2,j}, Decode_{3,j})), F_j) \tag{6}$$

$$\mathcal{L}_{total} = \mathcal{L}_{main} + \lambda\mathcal{L}_{fusion} \tag{7}$$

The redesigned FaultSeg Swin-UNETR network can learn fault features more easily. It is worth noting that after pre-training with Swin-UNETR on SimMIM, we can create three copies of the parameters of the decoder part to initialize the weights of FaultSeg Swin-UNETR. This allows us to leverage the prior knowledge obtained from pre-training in the seismic fault domain, speeding up network convergence and improving recognition accuracy. In the finetuning phase, we start from the weights obtained during self-supervised pre-training. We continue to use the AdamW optimizer and employ a cosine annealing learning rate decay strategy to gradually decrease the learning rate, ensuring the convergence of the network.

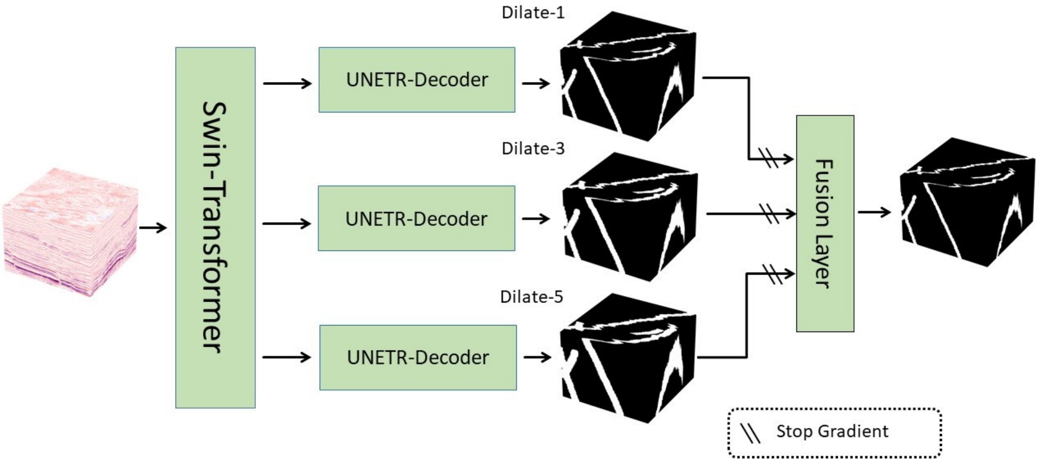


Figure 9. The architecture of the FaultSeg Swin-UNETR network.

4. Experimental Results and Analysis

This section introduced the experiments conducted and the results obtained. Section 4.1 details the evaluation metrics used for fault recognition tasks. Section 4.2 describes the training details of our algorithm. Finally, Sections 4.3 and 4.4 provide a detailed discussion of the results and an ablation study.

4.1. Evaluation Metrics

Fault recognition is a typical binary segmentation task; therefore, classic classification task metrics such as precision, recall, and F1 score can be used to measure the model's performance. However, evaluating the model solely based on precision or recall is not reasonable because of the low proportion of real earthquake faults. Therefore, following the work of An et al. [23], we use the per-image best threshold (OIS) and fixed contour threshold (ODS) metrics, calculated based on the F1 score, to assess the performance of the model. In the case of deep segmentation networks, the output is a 3D score volume with logits of the same size as the input seismic data. After applying the sigmoid activation function, the output is a 3D score volume ranging from 0 to 1. To calculate the OIS and ODS, the seismic faults in the test set are considered as multiple images along the crossline, $i = 1, 2, ..., N$. The F1 score is computed for the network's output score volume at 100 different thresholds. The best threshold is then selected either per image or for the entire volume to evaluate the model's performance. In this context, let us denote the model's output score as Y_{pred} and the ground truth labels for the faults as GT. Both Y_{pred} and GT are represented as 3D volumes with dimensions $H \times W \times D$. The evaluation metrics, OIS and ODS, are defined by the following equations (refer to Equation (8)):

$$\begin{aligned} OIS &= \frac{1}{N}\sum_{i=1}^{N} \max\left(\left\{ \mathrm{F1}\left(Y_{pred}[i,:,:] > \frac{j}{100}, GT[i,:,:]\right)\right\}_{j=1}^{100}\right) \\ ODS &= \max\left(\left\{ \frac{1}{N}\sum_{i=1}^{N} \mathrm{F1}\left(Y_{pred}[i,:,:] > \frac{j}{100}, GT[i,:,:]\right)\right\}_{j=1}^{100}\right) \end{aligned} \tag{8}$$

The formulations reveal that ODS, which determines the optimal threshold across the entire dataset, typically yields a value less than or equal to OIS. The latter metric computes the optimal threshold for each individual slice. Consequently, ODS offers a more comprehensive assessment of the model's efficacy across the whole seismic dataset.

4.2. Training and Validation

Regarding the datasets from Thebe and our collection, we divided the entire dataset into smaller chunks because of its large size. During the self-supervised stage, we used a patch size of 128 × 128 × 128 and a stride of 64 × 64 × 64 for chunking. Since we applied random masking to the input to the network each time, we did not perform excessive data augmentation. Instead, we only normalized each chunk of data by adjusting its mean to 0 and standard deviation to 1, making it easier for the neural network to process. During the fine-tuning stage, we divided the Thebe data into blocks of size 256 × 256 × 256 with a corresponding stride of 128 × 128 × 128. Before inputting each data block into the segmentation network, we applied random flips, random rotations by 90 degrees, and random cropping of seismic data from the blocks of size 256 × 256 × 256 to 128 × 128 × 128.

Additionally, we performed data normalization and other data augmentation operations. We utilized the slide window inference technique during the inference stage to generate predictions for the entire work area. This involved predicting a volume of size 128 × 128 × 128 at a time, with a sliding stride of 64 × 64 × 64. We calculated the average values in overlapping regions to address inconsistent predicted results at the boundaries. This choice of data augmentation method helped effectively mitigate the issue of inconsistent predictions at the junctions. The code for the entire pre-training and fine-tuning tasks was implemented using PyTorch 1.12.1 and accelerated using 4 × 4 NVIDIA A100 GPUs supported by the High-performance Computing Platform of Peking University. Because of memory limitations, we set the batch size for all tasks to two. The SimMIM self-supervised pre-training for the backbone network Swin-Transformer and the whole segmentation network Swin-UNETR was trained for 300 epochs. During the fine-tuning stage, all Transformer-based models ran for 100 epochs.

4.3. Results

Table 2 shows the OIS and ODS performance indicators of our FaultSeg Swin-UNETR model on the Thebe dataset. We primarily compare our results with An's work [42]. In that study, they found that the DeeplabV3 segmentation network pre-trained on ImageNet with ResNet-101 performed the best and even surpassed the foundational work of FaultSeg in 3D fault detection. Our experimental results show that our end-to-end SimMIM pre-training FaultSeg Swin-UNETR surpasses previous methods in both indicators, improving by at least 0.25 points compared to An's state-of-the-art work. In this section, we delve into the motivations behind the design of FaultSeg Swin-UNETR, validate the effectiveness of various pre-training strategies through a series of experiments, and confirm that every technique we propose significantly improves the detection accuracy of seismic faults.

Our research used UNETR and Swin-UNETR based on the 3D Transformer architecture as backbone networks. We conducted experiments on several self-supervised pre-training tasks, including MAE, SimMIM, SimCLR, and the masked volume inpainting, image rotation, and contrastive coding implemented within the Swin-UNETR. The results show that even in the absence of self-supervised pre-training, the performance of UNETR and Swin-UNETR is superior to the traditional 3D convolutional network-based FaultSeg3D, emphasizing the effectiveness of the combined Transformer and U-Net architecture for seismic fault segmentation tasks. However, the performance of the Swin-UNETR model without pre-training is similar to that of UNETR and needs improvement. Both backbone and entire segmentation networks can achieve better results than UNETR or Swin-UNETR under self-supervised pre-training. This indicates that self-supervised pre-training further enhances the ability of these models to extract seismic fault features. Despite this, we observe that after self-supervised pre-training, the performance improvement of UNETR is different than expected. This may be because, compared to CNN, the Transformer architecture requires more training samples to achieve optimal performance. Although we provided a large number of unsupervised data, the fault annotation data in the Thebe dataset still needs to be improved for the standard version of the Transformer. Relative to this, the Swin-Transformer structure with fewer parameters is more suitable for this task.

We improved the multi-task self-supervised pre-training method proposed by Tang et al. for Swin-UNETR in the field of medical imaging [38]. The previously designed tasks (masked volume inpainting, image rotation, and contrastive coding) have limited pre-training effects on seismic data, as shown in the 12th row of Table 2. Although these multi-task self-supervised pre-training tasks have a role in improving model performance, they are not as effective as the SimMIM method in seismic fault recognition tasks. Likewise, the effect of SimMIM is superior to MAE and SimCLR, indicating that the simple SimMIM is more suitable for fault recognition tasks. The original three-task-pre-trained task of Swin-UNETR may increase the complexity and difficulty of model training. Contrastive learning in SimCLR focuses more on global features, which may not be sufficient to capture tiny changes in seismic data. MAE is more suitable for ViT structures. In comparison, SimMIM, designed for Swin structures, provides a more straightforward and efficient method, focusing on the most critical learning tasks. Predicting the masked part encourages the model to pay attention to the subtle differences in the image, making it more effective in seismic fault detection.

Pre-training the entire segmentation network with SimMIM can yield the most significant improvements in downstream segmentation tasks. We attribute this phenomenon to the fact that all network components gain substantial prior knowledge during the pre-training stage and are not just limited to the Swin-Transformer backbone with fewer parameters. However, this may also be due to the irrational design of Swin-UNETR, whose backbone parameters are much smaller than those of the segmentation decoding head, only one-eighth the size. The design goal of most self-supervised pre-training networks is to make the backbone as complex and parameter-rich as possible, while smaller parameter networks usually handle downstream tasks. Nevertheless, pre-training the entire segmentation network can significantly alleviate this problem. As shown in the lower half of Table 2, both Swin-UNETR and FaultSeg Swin-UNETR, which incorporates the multi-scale decoding and fusion module, have greatly improved performance because of full-network pre-training.

Ultimately, our FaultSeg Swin-UNETR model is optimized explicitly for seismic faults' sparseness and slender characteristics. This model can restore the original fault structure more accurately by introducing multi-scale decoding objectives and integrated fusion modules. We deeply analyze the design of FaultSeg Swin-UNETR to demonstrate its rationality. Even without pre-training support, FaultSeg Swin-UNETR significantly outperforms the standard Swin-UNETR in detection performance. This enhancement allows us to further improve the performance of models based on the Swin-UNETR structure, making it reach the industry-leading level on the Thebe dataset.

Table 2. Performance on the Thebe dataset using OIS and ODS metrics, evaluating FaultSeg Swin-UNETR multi-scale decoding and fusion module design and SimMIM pre-training strategy for the entire segmentation network.

Model	Pre-Trained Method	OIS	ODS
UNet	-	0.769	0.766
HED	-	0.811	0.806
RCF	-	0.806	0.800
DeeplabM	-	0.759	0.756
DeeplabV3	Imagenet-pre-trained	0.849	0.845
FaultSeg3D	-	0.840	0.836
UNETR	-	0.845	0.841
UNETR	SimCLR	0.849	0.845
UNETR	MAE	0.847	0.844
UNETR	SimMIM	0.846	0.843
Swin-UNETR	-	0.844	0.840
Swin-UNETR	Three-task-pre-trained	0.852	0.845
Swin-UNETR	SimCLR	0.855	0.852
Swin-UNETR	MAE	0.859	0.857
Swin-UNETR	SimMIM	0.861	0.857
Swin-UNETR	Overall SimMIM	0.872	0.868
FaultSeg Swin-UNETR	-	0.851	0.847
FaultSeg Swin-UNETR	SimMIM	0.866	0.862
FaultSeg Swin-UNETR	Overall SimMIM	**0.875**	**0.870**

4.4. Ablation Study

Although neural networks based on the Transformer model require a large number of training data to achieve outstanding detection performance, we conducted fine-tuning experiments on a small-scale synthesized dataset to demonstrate that self-supervised pre-training can still achieve specific results under such stringent conditions. For our experiments, we used a synthetic dataset of 220 samples of size 128 × 128 × 128, as used by Wu et al. [21] in the FaultSeg3D model. We only used 200 samples to train the model, and the model selection was based on the performance of 20 validation samples from the synthetic dataset. After completing the fine-tuning, we directly evaluated the performance of the trained model on the test set of Thebe using the OIS and ODS metrics (Table 3). Because of the smaller scale and lower quality of the synthetic dataset compared to field seismic data, the model's performance showed a significant drop. However, compared to the FaultSeg3D model, the Swin-UNETR model performed significantly better, and further improvements can be achieved through the SimMIM self-supervised pre-training. Similar to previous conclusions, using the FaultSeg Swin-UNETR model combined with self-supervised pre-training yielded the best results. Moreover, the model could still achieve acceptable predictive performance on a relatively more minor annotated dataset, which indicates that our method can maintain decent performance in transferring between different work regions even with limited annotated data. This finding provides a baseline data volume reference for new work regions with less data. In addition, it also suggests our method provides an efficient pre-trained model. This pre-trained model offers an optimal starting point for fault prediction in new work regions. This experiment also significantly improves deep network performance by introducing large-scale annotated data. Though the Thebe dataset is currently the most extensive earthquake fault annotation dataset, it still needs more data compared to medical imaging to fully demonstrate the advantages of deep models.

Table 3. Testing results on the Thebe dataset using our proposed method fine-tuned on a small synthetic dataset.

Model	Pre-Trained Method	OIS	ODS
FaultSeg3D	-	0.441	0.431
Swin-UNETR	Three-task-pre-trained	0.482	0.473
Swin-UNETR	SimMIM	0.503	0.495
FaultSeg Swin-UNETR	Overall SimMIM	0.525	0.510

Because of the sizeable seismic data volume, it is often necessary to divide it into blocks for training data in the deep network. However, because of the limitation of GPU memory, the size of the input data given to the segmentation network is only 128 × 128 × 128. We used the sliding window method during prediction to infer the entire test data volume. To improve the consistency between sliding windows, we usually chose half of the window size as the sliding stride and took the average value of logits at the overlapping areas. In our deep network setup, the window size was 128 × 128 × 128, and the sliding stride was 64 × 64 × 64. However, the choice of block size during data partitioning was also worth considering. Since random cropping during the training process can serve as a powerful data augmentation technique and help the model learn fracture information at different positions in the data volume, it was necessary to set the block size larger than the input size that the network could accept. Therefore, we explored the block size of the data volume based on the Swin-UNETR baseline model. From Table 4, it can be seen that if the block size is set to 128 × 128 × 128, which means no random cropping during the training process, the performance is the worst in terms of OIS and ODS indicators. As we gradually increase the block size, both evaluation metrics significantly improve. However, the block size should not be too large, as this can lead to a significant difference in the randomly cropped 128-sized inputs encountered by each epoch when fitting the same data block, making it difficult for the network to converge. In this baseline model, we found that the OIS and ODS scores were highest when the block size was 256. Therefore, we used this setting in all experiments, as shown in Table 2.

Table 4. Comparison of different data chunking sizes.

Model	Chunk Size	OIS	ODS
Swin-UNETR	128 × 128 × 128	0.830	0.827
	192 × 192 × 192	0.841	0.835
	256 × 256 × 256	0.844	0.840
	512 × 512 × 512	0.832	0.829

The proportion of positive samples in the fault labels is too low (Figure 5). Therefore, it is easy to consider increasing the weight of positive samples in the BCE loss to enhance the model’s ability to detect faults. However, adjusting the weight of positive samples only changes the balance between precision and recall and does not significantly affect the OIS and ODS, which are evaluated using the F1 score. This is also why we do not use the average precision (AP) as an evaluation metric. We can increase the precision of the model by adjusting the weight of positive samples in the BCE loss. However, this comes at the cost of reducing recall. The experimental results in Table 5 also confirm our observation. Therefore, we used the regular BCE loss without any special weighting for positive samples in the remaining experiments.

Table 5. The impact of BCE positive weight.

Model	Positive Weight	OIS	ODS
Swin-UNETR	1.0	0.844	0.840
	5.0	0.845	0.839
	10.0	0.843	0.841

5. Discussion

This section presents test cases for the 2D slices and 3D cubes of the Thebe dataset. The 2D showcase is shown in Figure 10, and the 3D showcase is shown in Figure 11.

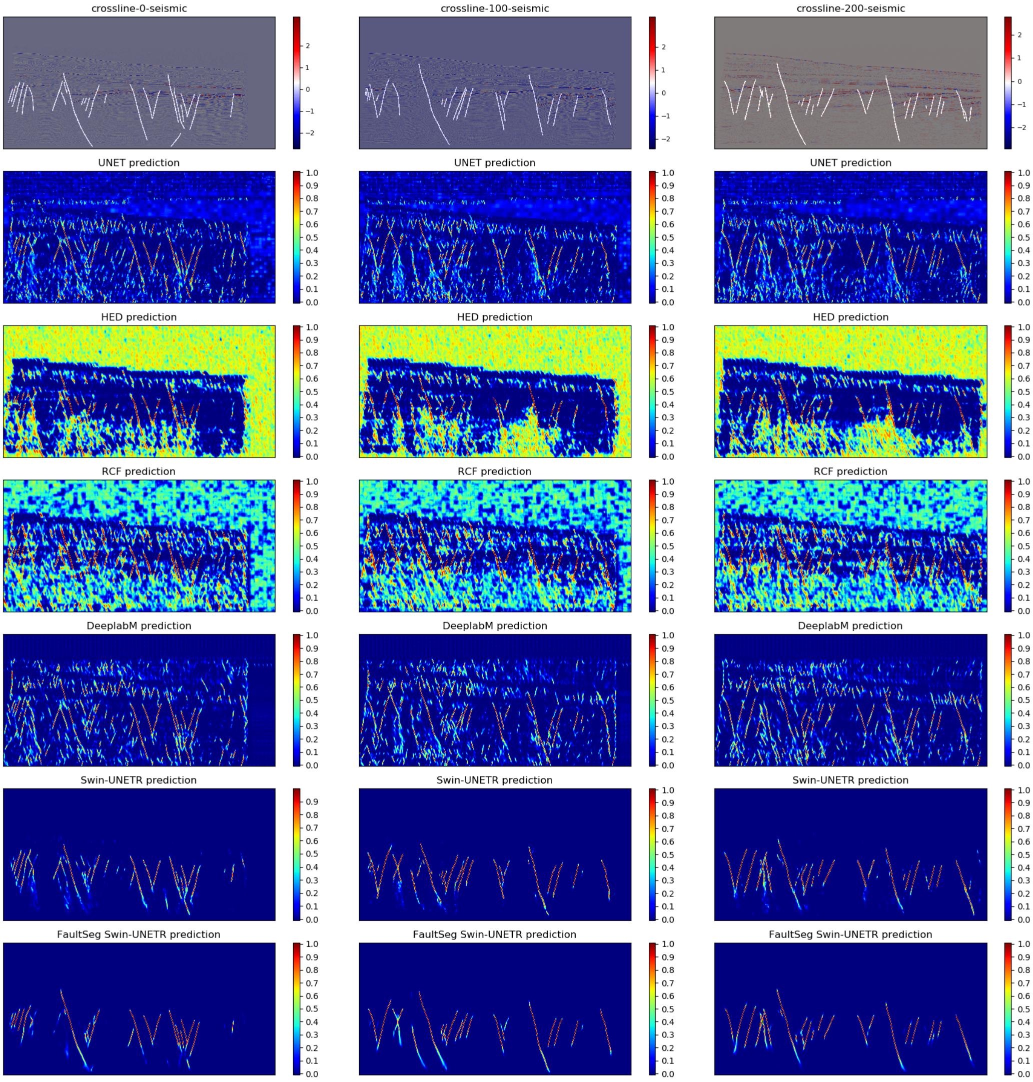
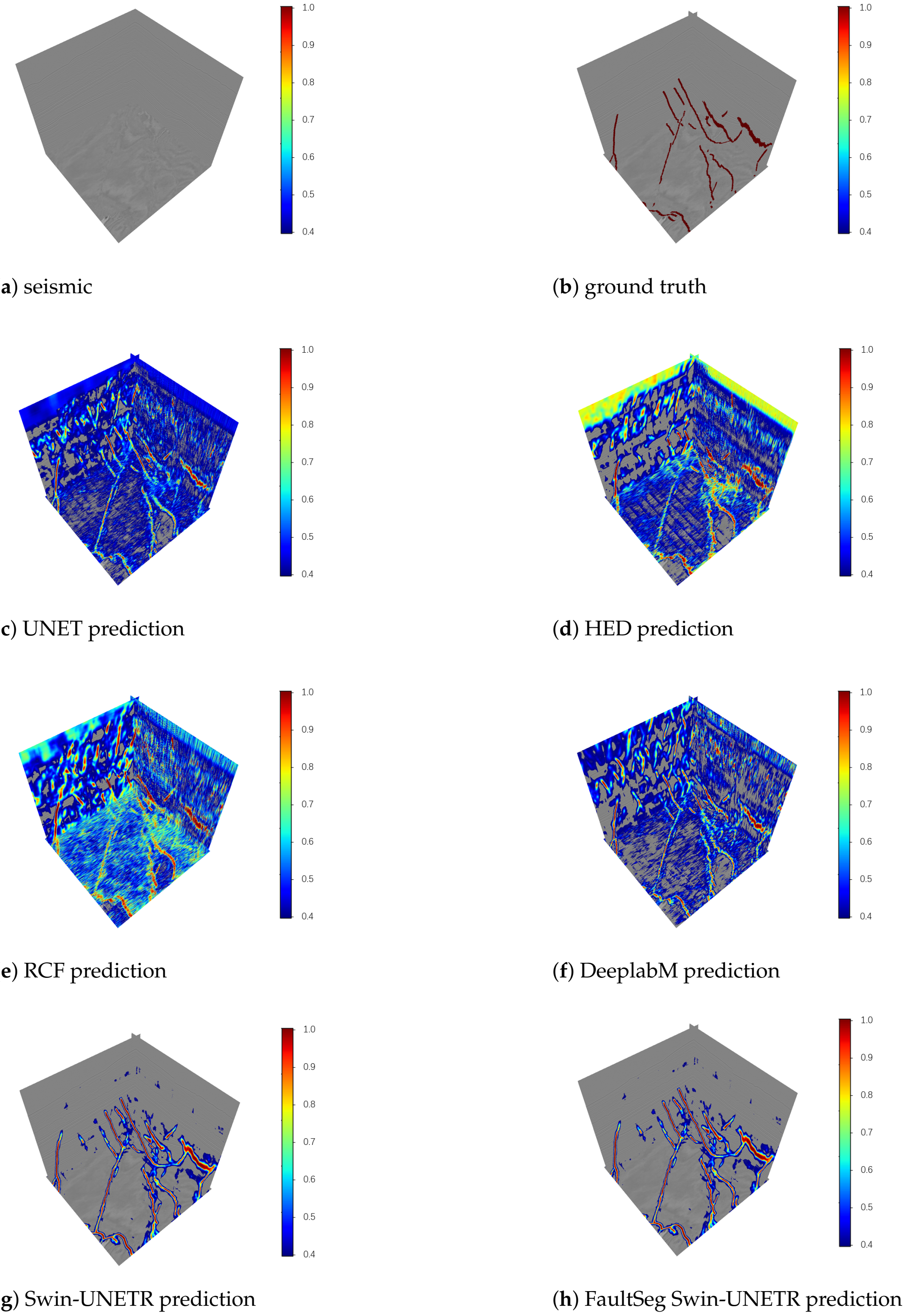


Figure 10. Comparison of different deep learning models on the Thebe test sets' 0th, 100th, and 200th crosslines. The first row represents expert annotations, followed by the second through fifth rows, displaying the inferences from the models proposed by An et al. [23] for comparison. The sixth and seventh rows present the visual probability maps predicted by Swin-UNETR and our FaultSeg Swin-UNETR, respectively.



(**a**) seismic

(**b**) ground truth

(**c**) UNET prediction

(**d**) HED prediction

(**e**) RCF prediction

(**f**) DeeplabM prediction

(**g**) Swin-UNETR prediction

(**h**) FaultSeg Swin-UNETR prediction

Figure 11. The 3D images are 640 × 640 × 640 cubes extracted from the testing subset of the Thebe dataset. We superimposed various deep learning models' predictive probabilities onto profiles in different directions, denoted as (**c**–**h**) for UNET, HED, RCF, DeeplabM, Swin-UNETR, and FaultSeg Swin-UNETR, respectively. Notably, FaultSeg Swin-UNETR accurately delineated the majority of fault lines.

5.1. Performance of 2D Slices

We tested the performance of several fault recognition methods on parts of the Thebe seismic data test set. Figure 10 displays the crossline of 0, 100, and 200 traces. The white lines in the first row represent the real faults of the strata on the seismic images as interpreted by experts for comparison with the performance of each automatic fault recognition model. We use fault likelihood to represent the prediction results of each model. The range of each pixel in the figure is zero to one. Values close to one (red) represent a higher probability of faults, and values close to zero (blue) represent a higher probability of non-faults. As expected, as shown in the quantitative evaluation results in Table 2, compared to the prediction results of the four models proposed by An et al. [23]—UNET (second row), HED (third row), RCF (fourth row), DeeplabM (fifth row)—Swin-UNTER achieves significant performance improvement, and our model achieves more accurate interpretations. Continuous fault prediction for the Thebe dataset means better results [23]. Specifically, the four models in Figure 10 (second through fifth rows) are susceptible to noise. In the probability map, many minor points and line segments appear in the non-fault part and are misjudged as faults, confirming the work [23], which said, "All four models struggle with the migration noise within the deeper layers, which the models do not see during the training phase". In comparison, UNet and DeeplabM are more accessible to identify non-faults than HED and RCF. At the same time, the HED and RCF models with weighted cross-entropy loss functions pay more attention to the fault part, so it is easier to misjudge non-fault pixels as faults. In particular, model RCF goes to the extreme of labeling non-fault pixels or noise as faults. Additionally, when calculating OIS and ODS values, these four models only extract the concentrated distribution of faults in the image. However, we lack prior knowledge of the concentrated fault areas in practical applications. The significant misjudgment in non-fault areas challenges the models when applied in real scenarios. Encouragingly, the Swin-UNETR segmentation model shows good prediction performance. Its prediction outcomes align more closely with expert interpretations than the previous four models, effectively identifying noise and non-fault areas, indicating this model's successful extraction of noise and non-fault features. The model structure and self-supervised pre-training tasks effectively improve the fault recognition capabilities. However, Swin-UNETR also has some redundant false detections and artifacts, and some faults are disconnected. This is due to the sparse and narrow earthquake faults, resulting in highly imbalanced data annotation. Therefore, the FaultSeg Swin-UNETR we proposed solves this problem to a certain extent through a multi-scale fusion decoder. As shown in the last row of Figure 10, our method is improved based on Swin-UNETR. Most fault points have a very high probability (close to one), the noise points are minimal, the predicted faults are complete, and they have more excellent continuity and refinement, almost comparable to expert explanations.

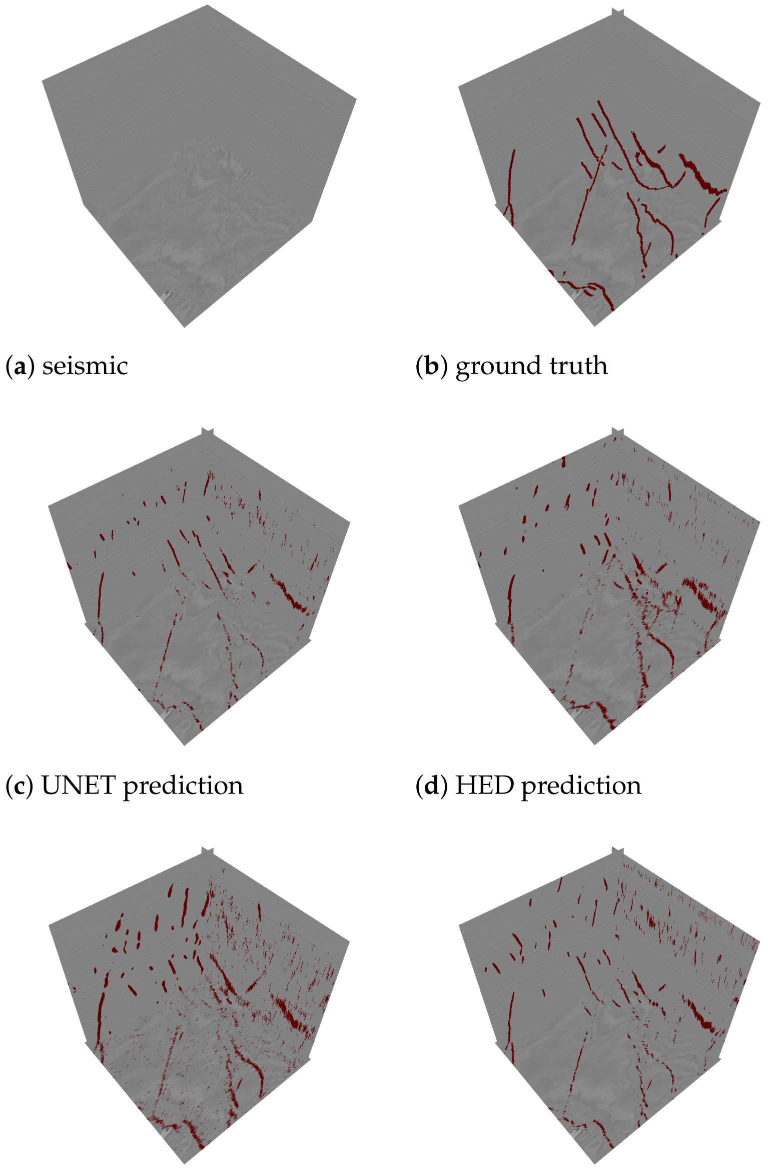
5.2. Performance of 3D Volumes

To further verify our model's fault recognition performance on 3D seismic volumes, we visualize the 3D inferences of Swin-UNETR, FaultSeg Swin-UNETR, and four other fault recognition methods on the Thebe test set in Figure 11. Figure 11c–f display six fault recognition probability maps computed by UNET, HED, RCF, DeeplabM, Swin-UNETR, and our FaultSeg Swin-UNETR methods. Figure 11a exhibits the extracted seismic volume, while the deep red annotations in Figure 11b represent seismic faults interpreted by experts.

Consistent with the 2D scenario, FaultSeg Swin-UNETR performs best in calculation accuracy, fault clarity, and completeness compared to the former four methods. It showcases more apparent fault characteristics and more continuous fault tracking in the 3D visualization. To observe multiple sets of fault features in different directions distinctly, we specifically visualize fault hard labels in Figure 12 using the following method:

$$\begin{aligned} \text{label}[Y_{pred} < 0.7] = 0 \\ \text{label}[Y_{pred} \geq 0.7] = 1 \end{aligned} \tag{9}$$

where Y_{pred} is the model output score, i.e., the probability of that pixel point being predicted as a fault. We regard voxels with a prediction probability greater than 0.7 as faults and the others as non-faults. The faults obtained by the former four methods are noisier, with numerous misidentified non-fault regions, discontinuities at connection points, and a lack of coherent structures within complex fault areas. However, our method provides clear and accurate fault segmentation results with good continuity at the intersection of three faces, aligning closely with real fault annotations. Dou et al. [43] explained that larger angles between faults and slices lead to poorer predictive outcomes. The skewed distribution of faults in the Thebe dataset poses substantial learning challenges for models and likely contributes to misjudgments in the former four models. Our model adeptly extracts 3D features, significantly enhancing predictive consistency. Compared to Swin-UNETR, FaultSeg Swin-UNETR demonstrates finer fault segmentation results with fewer misidentified fault points, indicating the effective enhancement of the model's ability to characterize faults through the multi-scale fusion module.



(**a**) seismic

(**b**) ground truth

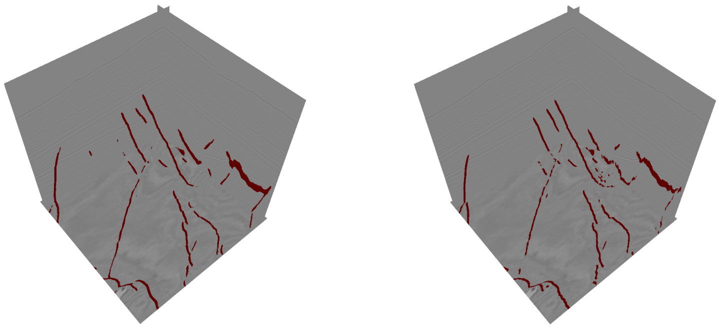
(**c**) UNET prediction

(**d**) HED prediction

(**e**) RCF prediction

(**f**) DeeplabM prediction

Figure 12. *Cont.*



(**g**) Swin-UNETR prediction (**h**) FaultSeg Swin-UNETR prediction

Figure 12. Visualizations of the hard scores predicted by the six models on the Thebe dataset are presented in a 3D cube. The deep red indicates that the probability of a pixel being predicted as a fault is equal to or greater than 70%; otherwise, it is considered a non-fault.

6. Conclusions

In our work, we explore the application of the most popular self-supervised pre-training in seismic fault recognition in recent years.

In summary, the contributions of this article are as follows:

- Utilizing the 3D Swin-Transformer backbone network, we investigated diverse pre-training methods with a substantial volume of field 3D seismic data. The integration of SimMIM's pre-training method with the enhanced Swin-UNETR model markedly improved performance. Consequently, we introduced FaultSeg Swin-UNETR, a method meticulously crafted for the unique characteristics of seismic data.
- We improved the Swin-UNETR model structure to adapt to the sparse distribution of seismic fault data and the narrow line profile characteristics in the inline or cross-line directions, promoting multi-scale decoding and fusion, thereby making fault detection easier.
- Furthermore, upon recognizing the significant imbalance in the number of parameters between the decoder and the backbone network in the Swin-UNETR model, we proposed a strategy for pre-training the complete segmentation model, which further improves fault detection accuracy.
- In the end, our proposed method achieves state-of-the-art performances on the Thebe dataset according to the standard metrics of the optimal image scale (OIS) and optimal dataset scale (ODS) metrics.
- Our research significantly advances the precision and efficiency of seismic fault recognition, overcoming the constraints associated with reliance on annotated datasets. This breakthrough paves the way for developing more robust and generalizable models capable of addressing the inherent complexities of field seismic data.

Our research robustly demonstrates the substantial potential of self-supervised learning in interpreting seismic data. Using self-supervised pre-training on extensive datasets, we find that models can generalize and significantly enhance identification accuracy, even when only a minimal number of labeled data are available in new work zones. This is particularly valuable considering the often complex and costly nature of annotating seismic datasets. The models can identify and segment faults, thus aiding geologists and engineers in precisely understanding subsurface structures, which have potential application values in earthquake risk assessment, research on seismic resilience of infrastructure and buildings [44,45], etc. However, our current focus has been mainly on fault recognition. Future work can include a broader range of seismic image-to-image tasks, such as stratigraphic layer segmentation. Implementing multi-objective training can improve the model's feature extraction and generalization capabilities.

Furthermore, our validation work has been confined to the two publicly available datasets we could access because of the scarcity of publicly available labeled datasets. Given the opportunity, we would conduct extensive testing across various geological conditions. In summary, seismic analysis needs a universally applicable pre-trained model akin to ImageNet [46] in computer vision or the GPT [47] series in natural language processing. In addition, there needs to be more high-quality, open-source, large-scale, and diverse seismic datasets within the field. Given these challenges, it is imperative to forge a data-driven, unified model akin to SAM [48]. A model fortified with semi-automated annotation capabilities will significantly enlarge our fault dataset, propelling us toward creating a universally applicable fault recognition AI.

Author Contributions: Conceptualization, J.M. and Z.Z.; methodology, Z.Z. and R.C.; software, Z.Z.; validation, Z.Z. and R.C.; formal analysis, R.C.; investigation, Z.Z. and R.C.; writing—original draft preparation, Z.Z. and R.C.; writing—review and editing, J.M. and R.C.; visualization, Z.Z. and R.C.; supervision, J.M.; project administration, J.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program of China under Grant 2022YFB2703002.

Data Availability Statement: In this study, the pre-training dataset cannot be disclosed because of privacy constraints. Nevertheless, to ensure transparency and reproducibility, we have shared the checkpoint of the pre-training network on private data. Experts can use this checkpoint to replicate our methods. This selection serves the research objectives while minimizing potential impacts on privacy. The Thebe datasets we used in this paper are available in Harvard Dataverse: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/YBYGBK (accessed on 6 May 2023).

Acknowledgments: Supported by the High-performance Computing Platform of Peking University. Thanks for the private unlabeled seismic data provided by BGP Inc., China National Petroleum Corporation.

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

Appendix A

Our study incorporates the Swin-Transformer encoder, a sophisticated feature extraction network tailored for analyzing 3D data. This network processes three-dimensional seismic data, denoted as $S \in \mathbb{R}^{H \times W \times D}$, where H, W, and D represent the dimensions of inline, crossline, and timeline, respectively. The initial step involves a patch partitioning of the input 3D seismic data into non-overlapping volumetric blocks, each of size $2 \times 2 \times 2$. This partitioning results in a total of $\frac{H}{2} \times \frac{W}{2} \times \frac{D}{2}$ blocks, which are subsequently flattened and mapped into a 48-dimensional embedding space through a linear projection layer, forming the inputs for the Swin-Transformer.

The architecture of the Swin-Transformer is divided into four distinct stages, each meticulously designed to extract information from the 3D data volumes in a hierarchical manner. Within each stage, a reduction of the spatial dimensions by half along each axis leads to a significant decrease in the number of tokens, specifically to one-eighth of their original count, while simultaneously doubling the channel count. This process is visually represented in the left half of Figure 7, which annotates the dimensions of the feature maps fed into each stage. Comprising Swin-Transformer blocks and a merging module, each stage is pivotal in transforming the feature maps, with the merging module solely responsible for all such deformations.

The detailed structure of a Swin-Transformer block is showcased in the right half of Figure 7, emphasizing the employment of two window-based self-attention mechanisms: window multi-head self-attention (W-MSA) and shifted-window multi-head self-attention (SW-MSA). These mechanisms are instrumental in calculating self-attention scores among the input tokens. Considering $s^{in} \in \mathbb{R}^{H' \times W' \times D' \times C'}$ as the input to a Swin-

Transformer block, where it is perceived as tokens of dimension C' arranged across a grid of $H' \times W' \times D'$, the forward pass through this module can be articulated through the following set of equations. It is noteworthy that the dimensions of the feature maps are preserved throughout this process:

$$\begin{aligned} s' &= \text{W-MSA}(\text{LN}(s^{in})) + s^{in}, \\ s'' &= \text{MLP}(\text{LN}(s')) + s', \\ s''' &= \text{SW-MSA}(\text{LN}(s'')) + s'', \\ s^{out} &= \text{MLP}(\text{LN}(s''')) + s'''. \end{aligned}$$

In this context, LN denotes the layer normalization layer, strategically placed before each sub-module within the network to address the internal covariate shift dilemma encountered during training phases. The MLP, or multi-layer perceptron, along with W-MSA and SW-MSA, the dual window-based self-attention mechanisms devised for the Swin-Transformer, play crucial roles. These mechanisms partition the input tokens into groups confined by a predetermined window size of $M \times M \times M$, leading to $\frac{H'}{M} \times \frac{W'}{M} \times \frac{D'}{M}$ groups, within which self-attention scores are computed as follows:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Here, $Q = W^q s$, $K = W^k s$, and $V = W^v s$ represent the query, key, and value components within the self-attention framework, respectively, with d_k reflecting their dimensionalities. The matrices W^q, W^k, and W^v facilitate the transformation of the input s into Q, K, and V, respectivelyx. In the W-MSA approach, tokens are partitioned in a manner that allocates equal window sizes across the board. Conversely, the SW-MSA strategy involves segmenting the input tokens by offsetting the windows along the three spatial dimensions by $\frac{M}{2}$ each. Figure A1 shows the visual results of these two window partitioning mechanisms. This partitioning scheme effectively balances the computational demands of the self-attention mechanism with the interactions among tokens across different window partitions, ensuring a harmonious integration of computational efficiency and inter-token dynamics.

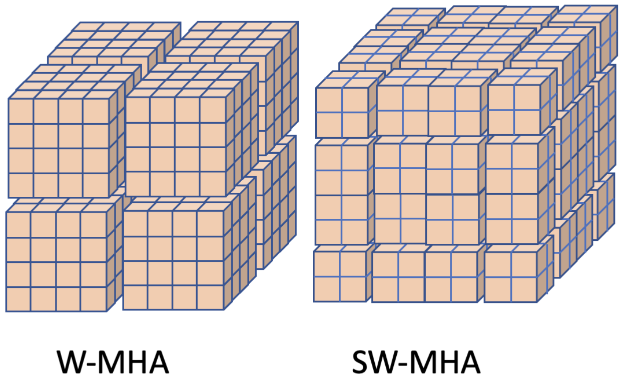


Figure A1. W-MHA and SW-MHA window partitioning mechanism.

References

1. Alcalde, J.; Bond, C.E.; Johnson, G.; Kloppenburg, A.; Ferrer, O.; Bell, R.; Ayarza, P. Fault interpretation in seismic reflection data: An experiment analysing the impact of conceptual model anchoring and vertical exaggeration. *Solid Earth* **2019**, *10*, 1651–1662. [CrossRef]
2. Bouvier, J.; Kaars-Sijpesteijn, C.; Kluesner, D.; Onyejekwe, C.; Van der Pal, R. Three-dimensional seismic interpretation and fault sealing investigations, Nun River Field, Nigeria. *AAPG Bull.* **1989**, *73*, 1397–1414.
3. Wu, X.; Geng, Z.; Shi, Y.; Pham, N.; Fomel, S.; Caumon, G. Building realistic structure models to train convolutional neural networks for seismic structural interpretation. *Geophysics* **2020**, *85*, WA27–WA39. [CrossRef]
4. Fossen, H. *Structural Geology*; Cambridge University Press: New York, NY, USA, 2010; pp. 205–207.

5. Posamentier, H.W.; Davies, R.J.; Cartwright, J.A.; Wood, L. *Seismic Geomorphology—An Overview*; Special Publications; Geological Society: London, UK, 2007.
6. Knipe, R.J.; Jones, G.; Fisher, Q. *Faulting, Fault Sealing and Fluid Flow in Hydrocarbon Reservoirs: An Introduction*; Special Publications; Geological Society: London, UK, 1998.
7. Ottesen Ellevset, S.; Knipe, R.; Svava Olsen, T.; Fisher, Q.; Jones, G. *Fault Controlled Communication in the Sleipner Vest Field, Norwegian Continental Shelf: Detailed, Quantitative Input for Reservoir Simulation and Well Planning*; Special Publications; Geological Society: London, UK, 1998; Volume 147. pp. 283–297.
8. Richards, F.L.; Richardson, N.J.; Bond, C.E.; Cowgill, M. *Interpretational Variability of Structural Traps: Implications for Exploration Risk and Volume Uncertainty*; Geological Society: London, UK, 2015; Volume 421, pp. 7–27.
9. Hale, D. Fault surfaces and fault throws from 3D seismic images. In Proceedings of the 2012 SEG Annual Meeting, Las Vegas, NV, USA, 4–9 November 2012.
10. Stark, T.J. Unwrapping instantaneous phase to generate a relative geologic time volume. In Proceedings of the 2003 SEG Annual Meeting, Dallas, TX, USA, 26–31 October 2003.
11. Wu, X.; Zhong, G. Generating a relative geologic time volume by 3D graph-cut phase unwrapping method with horizon and unconformity constraints. *Geophysics* **2012**, *77*, O21–O34. [CrossRef]
12. Silva, C.C.; Marcolino, C.S.; Lima, F.D. Automatic fault extraction using ant tracking algorithm in the Marlim South Field, Campos Basin. In Proceedings of the 2005 SEG Annual Meeting, Houston, TX, USA, 6–11 November 2005.
13. Pedersen, S.I.; Randen, T.; Sønneland, L.; Steen, Ø. Automatic fault extraction using artificial ants. In Proceedings of the 2002 SEG Annual Meeting, Salt Lake City, UT, USA, 6–9 October 2002.
14. Figueiredo, A.M.; Gattass, M.; Szenberg, F. Seismic horizon mapping across faults with growing neural gas. In Proceedings of the 10th International Congress of the Brazilian Geophysical Society, Rio de Janeiro, Brazil, 19–23 November 2007.
15. Zinck, G.; Donias, M.; Daniel, J.; Guillon, S.; Lavialle, O. Fast seismic horizon reconstruction based on local dip transformation. *J. Appl. Geophys.* **2013**, *96*, 11–18. [CrossRef]
16. Wang, Z.; AlRegib, G. Automatic fault surface detection by using 3D Hough transform. In Proceedings of the 2014 SEG Annual Meeting, Denver, CO, USA, 26–31 October 2014.
17. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006; Volume 4.
18. Kortström, J.; Uski, M.; Tiira, T. Automatic classification of seismic events within a regional seismograph network. *Comput. Geosci.* **2016**, *87*, 22–30. [CrossRef]
19. Guitton, A.; Wang, H.; Trainor-Guitton, W. Statistical imaging of faults in 3D seismic volumes using a machine learning approach. In *SEG Technical Program Expanded Abstracts 2017*; Society of Exploration Geophysicists: Houston, TX, USA, 2017; pp. 2045–2049.
20. Zhao, T.; Mukhopadhyay, P. A fault detection workflow using deep learning and image processing. In Proceedings of the 2018 SEG International Exposition and Annual Meeting, Anaheim, CA, USA, 14–19 October 2018.
21. Wu, X.; Shi, Y.; Fomel, S.; Liang, L. Convolutional neural networks for fault interpretation in seismic images. In Proceedings of the 2018 SEG International Exposition and Annual Meeting, Anaheim, CA, USA, 14–19 October 2018.
22. Wu, X.; Liang, L.; Shi, Y.; Fomel, S. FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation. *Geophysics* **2019**, *84*, IM35–IM45. [CrossRef]
23. An, Y.; Guo, J.; Ye, Q.; Childs, C.; Walsh, J.; Dong, R. Deep convolutional neural network for automatic fault recognition from 3D seismic datasets. *Comput. Geosci.* **2021**, *153*, 104776. [CrossRef]
24. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; Volume 30.
25. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
26. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
27. He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; Girshick, R. Masked autoencoders are scalable vision learners. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 16000–16009.
28. Xie, Z.; Zhang, Z.; Cao, Y.; Lin, Y.; Bao, J.; Yao, Z.; Dai, Q.; Hu, H. Simmim: A simple framework for masked image modeling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 9653–9663.
29. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. *PMLR* **2020**, *119*, 1597–1607.
30. Liu, B.; Jiang, P.; Wang, Q.; Ren, Y.; Yang, S.; Cohn, A.G. Physics-driven self-supervised learning system for seismic velocity inversion. *Geophysics* **2023**, *88*, R145–R161. [CrossRef]
31. Monteiro, B.A.; Oliveira, H.; dos Santos, J.A. Self-Supervised Learning for Seismic Image Segmentation From Few-Labeled Samples. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 8028805. [CrossRef]
32. Xu, Z.; Luo, Y.; Wu, B.; Meng, D.; Chen, Y. Deep Nonlocal Regularizer: A Self-Supervised Learning Method for 3D Seismic Denoising. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5921517. [CrossRef]

33. Yang, J.; Wu, X.; Bi, Z.; Geng, Z. A multi-task learning method for relative geologic time, horizons, and faults with prior information and transformer. *IEEE Trans. Geosci. Remote. Sens.* **2023**, *61*, 5907720. [CrossRef]
34. Tang, Z.; Wu, B.; Wu, W.; Ma, D. Fault Detection via 2.5 D Transformer U-Net with Seismic Data Pre-Processing. *Remote Sens.* **2023**, *15*, 1039. [CrossRef]
35. Silva, R.M.; Baroni, L.; Ferreira, R.S.; Civitarese, D.; Szwarcman, D.; Brazil, E.V. Netherlands dataset: A new public dataset for machine learning in seismic interpretation. *arXiv* **2019**, arXiv:1904.00770.
36. Wang, Z.; You, J.; Liu, W.; Wang, X. Transformer assisted dual U-net for seismic fault detection. *Front. Earth Sci.* **2023**, *11*, 1047626. [CrossRef]
37. Dou, Y.; Dong, M.; Li, K.; Xiao, Y. FaultSSL: Seismic Fault Detection via Semi-supervised learning. *arXiv* **2023**, arXiv:2309.02930.
38. Tang, Y.; Yang, D.; Li, W.; Roth, H.R.; Landman, B.; Xu, D.; Nath, V.; Hatamizadeh, A. Self-supervised pre-training of swin transformers for 3d medical image analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 20730–20740.
39. An, Y.; Du, H.; Ma, S.; Niu, Y.; Liu, D.; Wang, J.; Du, Y.; Childs, C.; Walsh, J.; Dong, R. Current state and future directions for deep learning based automatic seismic fault interpretation: A systematic review. *Earth-Sci. Rev.* **2023**, *243*, 104509. [CrossRef]
40. An, Y.; Guo, J.; Ye, Q.; Childs, C.; Walsh, J.; Dong, R. A gigabyte interpreted seismic dataset for automatic fault recognition. *Data Brief* **2021**, *37*, 107219. [CrossRef]
41. Hatamizadeh, A.; Tang, Y.; Nath, V.; Yang, D.; Myronenko, A.; Landman, B.; Roth, H.R.; Xu, D. Unetr: Transformers for 3d medical image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 574–584.
42. An, Y.; Dong, R. Understanding the Effect of Different Prior Knowledge on CNN Fault Interpreter. *IEEE Access* **2023**, *11*, 15058–15068. [CrossRef]
43. Dou, Y.; Li, K.; Zhu, J.; Li, T.; Tan, S.; Huang, Z. MD loss: Efficient training of 3-D seismic fault segmentation network under sparse labels by weakening anomaly annotation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5919014. [CrossRef]
44. Kourehpaz, P.; Molina Hutt, C. Machine learning for enhanced regional seismic risk assessments. *J. Struct. Eng.* **2022**, *148*, 04022126. [CrossRef]
45. Forcellini, D. An expeditious framework for assessing the seismic resilience (SR) of structural configurations. *Structures* **2023**, *56*, 105015. [CrossRef]
46. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
47. Floridi, L.; Chiriatti, M. GPT-3: Its nature, scope, limits, and consequences. *Minds Mach.* **2020**, *30*, 681–694. [CrossRef]
48. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment anything. *arXiv* **2023**, arXiv:2304.02643.

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.