CoSeDif: A Conditional Diffusion Approach to Mitigate Synthetic-Real Disparities in Seismic Fault Detection

A Preprint

Isack Farady
Department of Electrical Engineering
Yuan Ze University Taiwan
isackf@saturn.yzu.edu.tw

Soufiene Sellami
Department of Electrical Engineering
National Engineering School of Carthage Tunisia
ssoufiene21@gmail.com

Chia-Chen Kuo
National Center for High-performance Computing
National Applied Research Laboratories Taiwan
cckuo@narlabs.org

Chih-Yang Lin
Department of Mechanical Engineering
National Central University Taiwan
andrewlin@ncu.edu.tw

Abstract

While manual seismic interpretation requires significant resources, recent advancements in automation tools have emerged to assist experts in analyzing the data more efficiently. Deep learning models, in particular, have become invaluable in automating the process of fault detection. However, the availability of labeled seismic fault data remains extremely limited, which pushes researchers to rely on synthetic images. Despite their usefulness, the heavy reliance on synthetic data often hampers the generalization capabilities of deep learning models. In this work, we introduce CoSeDif (Conditional Seismic Diffusion) network, a novel conditional denoising diffusion model designed to generate high-quality synthetic seismic data for fault detection. Unlike traditional methods that rely on simplistic and conventional fault generation approaches, CoSeDif generates realistic 2D seismic images by conditioning on fault layer attributes and edge maps, providing a smoother representation of geological features. A key finding of our model is the introduction of a seismic conditional encoder, which incorporates seismic layer information extracted from canny edge detection algorithm, thereby enhancing the model's ability to capture complex geological layers and structures. This conditional encoder enriches the generation process by ensuring that the synthetic images reflect the nuances of original seismic layers, improving both the fidelity and realism of the generated data. CoSeDif leverages a dual-path encoder-decoder architecture, combining ResNet blocks with transformer-based attention mechanisms to capture both local and global seismic features. We evaluate CoSeDif using several metrics, including MSE, PSNR, DSSIM, and FID, demonstrating that it outperforms state-of-the-art generative models like Pix2Pix and ControlNet in terms of fidelity and structural quality. Furthermore, we show that models trained on synthetic data generated by CoSeDif achieve performance comparable to those trained on real seismic data, effectively bridging the gap between synthetic and real-world applications.

Keywords Seismic Fault Detection · Diffusion · Synthetic · Deep Learning · Seismic Generative Model

1 Introduction

Seismic faults are fractures in the Earth's crust where movement has occurred, making their accurate identification crucial for reservoir characterization, hydrocarbon exploration, and geohazard assessments. Faults play a significant role in controlling fluid flow within reservoirs, which directly impacts resource extraction and the management of subsurface risks. Proper interpretation of faults is essential not only for maximizing the efficiency of hydrocarbon recovery but also for ensuring the safety and sustainability of operations. However, the task of fault interpretation is complex and requires a high level of expertise, as seismic faults are often subtle and difficult to detect in seismic images.

In order to develop automated methods for fault analysis, many approaches have been designed to automatically detect faults using advanced models, including deep learning-based models CNN model for generating synthetic images. However, Most seismic data lack labels, and in the case of fault detection, fault instances are vastly outnumbered by normal data, further complicating model training [1]. The current state-of-the-art approach to address this issue is by introducing synthetic data, which can help models learn fault characteristics more effectively. However, existing methods for generating synthetic seismic images often overlook the detailed stratigraphic facies and layering present in real seismic data. These synthetic methods incorporate various assumptions and generalized variations of fault structures, which may result in unrealistic stratigraphic sequences and poorly represent actual geological formations.

Advancements in generative models using deep learning, such as Generative Adversarial Networks (GANs) and diffusion models, have shown promise in various seismic applications. Generative-based networks have been applied to address challenges in resolution enhancement [2], reconstruction [3], and inversion [4, 5, 6, 7]. However, earlier generative models were limited by the quality of input sketches and did not explore the potential of generated data for specific seismic interpretation tasks. More recent generative models, particularly diffusion models, have outperformed GANs in terms of image quality and training efficiency due to their simpler denoising process [8, 9]. Yet, the potential of diffusion models for generating synthetic seismic images remains underexplored. Leveraging diffusion models offers a unique opportunity to produce high-fidelity synthetic data, bridging the gap between synthetic and real seismic datasets.

To address the issue of seismic data scarcity and improve the robustness of fault detection models, we propose the use of modified generative AI models, particularly diffusion models [10], to generate synthetic seismic images that closely mimic real seismic patterns. Unlike conventional generative approaches, our diffusion model is designed to add a small additional encoder to captures the key characteristics of faults which is the facies layers by performing Canny edge to get the seismic layer. This conditional encoder process the edge layer combined with fault map as conditions for generating synthetic fault images. During training, The diffusion model learns to generate the seismic layer that considering the condition given by additional encoder. This approach ensures that the generated images more accurately reflect the complexities and subtleties of actual seismic faults. By generating high-quality synthetic seismic images using this pretrained diffusion model, we aim to provide a large and diverse dataset that can be used to train fault detection models more effectively. In summary, our contributions can be described as follows:

- We introduce a novel conditional denoising diffusion model designed to generate synthetic seismic data. The model is conditioned on fault segmentation maps and edge features, extracted using Canny edge detection, to enhance its ability to capture complex seismic characteristics.
- We demonstrate that our approach outperforms state-of-the-art generative models in key quality metrics for synthetic data. Furthermore, we validate the fidelity of the generated seismic images by comparing them to corresponding fault masks through segmentation, showing that models trained on our synthetic data perform comparably to those trained on real data.
- To the best of our knowledge, this is the first application of diffusion models to seismic data synthesis, achieving high quality and fidelity. We hope this work will encourage further research and help overcome the limitations of synthetic data in seismic interpretation.

This paper is organized as follows. Section 2 reviews related work and discusses previous research relevant to our proposed approach. In Section 3, we present our proposed method in detail, along with the underlying rationale. Section 4 presents the experimental results and a thorough discussion of the findings. Finally, Section 5 concludes the paper by summarizing our contributions and outlining potential future directions.

2 Related Work

The detection of faults in seismic data has long been a challenge in geophysics. Traditional approaches typically relied on a two-stage process: first, computing seismic attributes to enhance discontinuities, such as coherence [11], dips and azimuth [12], and curvature [13]; and then applying specific algorithms to extract the faults. While these methods provided useful insights, they often suffered from limitations, such as the inability to capture subtle fault structures and a heavy reliance on manually tuned seismic attributes. These approaches lacked the flexibility to adapt to varying fault geometries. To fully exploit the spatial and contextual information inherent in seismic data, the Hough transform was initially introduced for seismic interpretation by [14] and later extended to 3D by [15]. Additionally, [16] proposed the use of ant tracking algorithms, while other studies focused on filtering techniques [17, 18]. However, these methods are highly sensitive to noise, and their parameters are often chosen through trial and error.

With the rise of deep learning, various models have been employed to solve seismic problems such as facies classification [19], layer segmentation [20], denoising [21], and seismic reconstruction [22]. In seismic fault detection, several CNN models have also been proposed, including [23], Transformer-based models [24, 25, 26], and SAM [27]. However, the lack of annotated data remains a significant challenge [28] to improve prediction performance. To address this data limitation, Wu et al. [29] introduced FaultSeg3D, a 3D U-Net model trained on synthetic data. Their method involves generating a random horizontal reflectivity model, incorporating folds and faults, convolving the model with a Ricker wavelet, and adding noise to produce synthetic seismic images. Other studies [30, 31, 32] have attempted to align synthetic data with real data by modifying the simulation process. However, these approaches still face limitations, such as inadequate representation of complex fault shapes and imperfect noise modeling.

Although these synthetic models are efficient and memory-conserving due to their reliance on simulations, the generated images still need to be interpreted or analyzed by experts to resemble real seismic fault images. In fault segmentation work by [33], their model, which was trained on synthetic data, underperformed when applied to field seismic data. According to a review by Lei et al. [34], more than half of the fault detection studies relied on synthetic data, underscoring the need to develop more realistic synthetic datasets to improve model performance. For instance, FaultSeg3D [29], trained solely on synthetic data, struggled with the Thebe dataset [35] due to differences in fault characteristics and annotation thickness, resulting in high false-positive rates and reduced performance.

Since the introduction of the attention mechanism in deep learning [36], many models have adopted the concept of focusing more on significant elements in an image. In seismic applications, understanding and emphasizing seismic layer information is essential. One approach is to control the generated image using seismic layer information. Unlike unconditional image synthesis [29], which generates images from random noise mimicking the training data distribution, conditional methods allow for greater control over the output, producing images that adhere to given specifications such as text prompts[37], segmentation maps[38], or image references[39].

Among these conditional approaches, using image references as input has given rise to a particularly important sub-field known as image-to-image translation. This sub-field has evolved alongside general generative AI, progressing from early approaches using conditional GANs [40], with Pix2Pix[41] being one of the most influential frameworks, to more recent advancements like ControlNet [42]. Controlnet adding spatial conditioning controls to large, pre-trained text-to-image diffusion models. However these methods primarily focus on natural images and are rarely adapted to the noisier conditions of seismic data and our work aims to bridge this gap by introducing a diffusion model tailored for seismic imaging.

3 Proposed Method

In this section, we provide a detailed discussion of our diffusion model, including the key design choices behind it. this section will cover the process of condition creation, the model architecture, and the training methodology, explaining how each component contributes to the overall performance of the model.

3.1 Diffusion Model

The seismic diffusion is developed based on the foundational principles of diffusion probabilistic models introduced by [10]. As illustrated in Figure 1, our proposed model learns to generate synthetic seismic images by applying a denoising approach structured as two Markov chain processes (forward and reverse). In forward process, random noise is added to the original seismic image $\mathbf{x}_0$ over a series of T steps (with $T = 1000$ in our case). At each step, a small amount of noise is added according to a fixed schedule, in this process we adopt a cosine scheduler to gradually transform the image into pure noise $\mathbf{x}_t$. This process is defined by:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\,\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \tag{1}$$

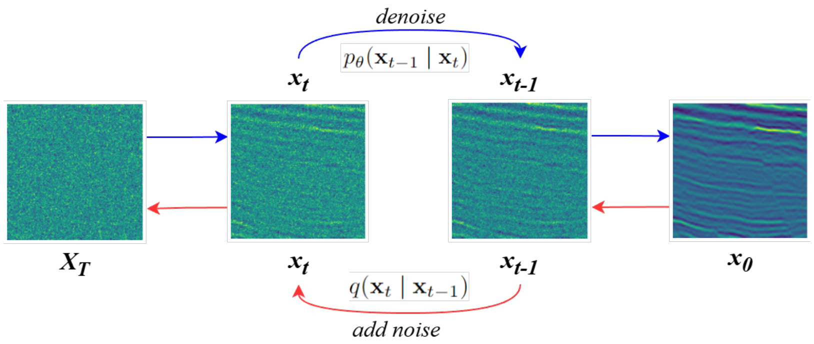
denoise
$p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$
x_t
x_{t-1}
X_T
x_t
x_{t-1}
x_0
$q(\mathbf{x}_t \mid \mathbf{x}_{t-1})$
add noise

Figure 1: The red arrows represent the forward process, where noise is progressively added to the seismic image from X_0 to X_T, causing the image to become increasingly noisy. The blue arrows represent the reverse (denoising) process, where each step moves from the noisy image X_T back towards the original clean image X_0, removing noise at each step.

where $q(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ represents the Gaussian distribution with mean $\sqrt{1-\beta_t}\,\mathbf{x}_{t-1}$ and variance $\beta_t\mathbf{I}$. This process can be reformulated to enable direct sampling of $\mathbf{x}_t$ from $\mathbf{x}_0$ with a single draw of Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon \tag{2}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$.

In denoising or reverse process, the neural network is trained to estimate the noise added at each diffusion step as formulated in Eq. 3. Given a noisy image $\mathbf{x}_t$ at step t, the objective is to minimize the difference between the predicted noise $\epsilon_\theta(\mathbf{x}_t, t)$ and the actual noise ϵ. This allows the diffusion model to effectively learn how to denoise the seismic image.

$$p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t) = \mathcal{N}\left(\mathbf{x}_{t-1}; \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)\right), \sigma_t^2\mathbf{I}\right) \tag{3}$$

- $\epsilon_\theta(\mathbf{x}_t, t)$ estimate function to predict noise ϵ.
- σ_t^2 is the variance term .

The loss function used for this purpose is derived from the mean squared error (Eq. 4). By minimizing this loss, the our diffusion model refines its ability to predict noise accurately.

$$L = \mathbb{E}_{\mathbf{x}_0, t, \epsilon}\left\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\right\|^2 \tag{4}$$

3.2 Conditional Diffusion

The generation of realistic synthetic seismic images remains a key challenge in fault detection. To address this, we propose a method that conditions the diffusion model on both fault and edge maps, enhancing the model's ability to accurately synthesize seismic data. The central idea behind our approach is to leverage existing fault segmentation data from public datasets to control the image generation process.

The process begins by preprocessing both the seismic image and the segmentation label. For the seismic image, noise is reduced using a Gaussian filter, and its amplitude is normalized to enhance contrast. The segmentation label is then aligned with the seismic image and converted into a binary format, where 1 represents fault/horizon pixels and 0 represents background. Next, the Canny edge detection algorithm is applied to both the seismic image and the segmentation label. This involves computing the gradient magnitude and direction, applying non-maximum suppression to thin the edges, and using double thresholding to identify strong and weak edges. The edges extracted from both the seismic image and segmentation label are then combined, with the fault locations and boundaries from both sources overlaid onto the original seismic image. This integrated approach

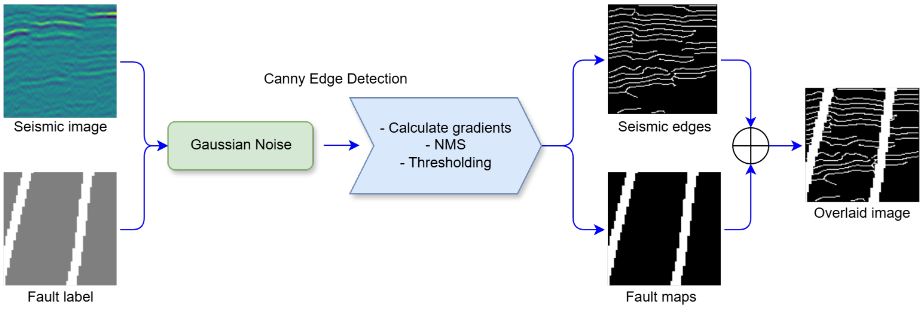

Figure 2: Canny edge detection for seismic images and segmentation labels.

provides a more comprehensive and accurate representation of geological features, enabling the model to better understand and generate realistic seismic data.

As shown in Figure 2, while fault maps provide essential information about seismic discontinuities, they often lack the detailed contextual information that geological boundaries and finer-scale features offer. Seismic layer edges, however, capture these subtle details, including layer boundaries and variations within the seismic data. To incorporate this critical structural information, we use the Canny edge detection algorithm to extract both fault locations and layer edges from the seismic data.

As hown in Figure 2, The fault locations and edges, detected by Canny, are fed as conditional inputs to the encoder, guiding the diffusion model to focus on these important features. Once the condition is prepared, it passes through the conditional encoder. The output of this encoder, $E_c(\mathbf{c}_t)$, is then combined with the output from the original image encoder, $E_x(\mathbf{x}_t)$, creating a unified representation that captures both the original seismic features and the additional conditional information.This fusion does not alter the forward process but modifies the estimate function as follows:

$$\epsilon_\theta(\mathbf{x}_t, \mathbf{c}_t, t) = D\left((E_c(\mathbf{c}_t) + E_x(\mathbf{x}_t))\, t\right) \tag{5}$$

- $E_c(\mathbf{c}_t)$ represents the embedding of the conditional input $\mathbf{c}_t$.
- $E_x(\mathbf{x}_t)$ denotes the embedding of the original image $\mathbf{x}_t$.
- The decoder D generates the predicted noise ϵ_θ based on both the image and the condition.

3.3 Model Architecture

The CoSeDif architecture consists of three main components: the encoder (*E*), conditional encoder (*Ec*), and decoder (*D*). Both encoder and decoder utilize ResNet as the CNN backbone, integrated with a transformer block in the middle. As illustrated in Figure 3. the both encoders *E* contain four hierarchical downsampling stages. Each stage comprises two ResNet blocks, followed by a linear attention block to capture long-range dependencies. Each ResNet block includes two convolutional layers, each featuring a 3x3 convolution, group normalization, and SiLU activation for efficient feature extraction and stable gradient flow. After the attention block, a downsampling operation is applied, either by pixel rearrangement or strided convolution, to reduce the spatial resolution while increasing feature dimensionality. The conditional encoder *Ec* mirrors the structure of *E*, processing auxiliary inputs (masks and edge maps) in a parallel path. The features extracted from *E* and *Ec* are then summed element-wise at the bottleneck.

The encoded features from both paths are processed by a middle vision transformer (VIT) block, which aggregates global context through self-attention and further refines the representations. These features are then passed to the mirrored decoder, which reconstructs the output using four upsampling stages. Each decoder stage contains two ResNet blocks and an optional attention block, followed by an upsampling operation using interpolation and convolution. Skip connections between the encoder and decoder ensure that spatial details are preserved for reconstruction quality. During training, CoSeDif relies on the MSE loss method from Eq.(4). With this loss, the CoSeDif learns to predict the noise, which is a critical step for effectively generating the synthetic image during the reverse diffusion process. The loss is computed for every training batch and backpropagated to update the model weights. The detailed architecture is illustrated in Figure 3.

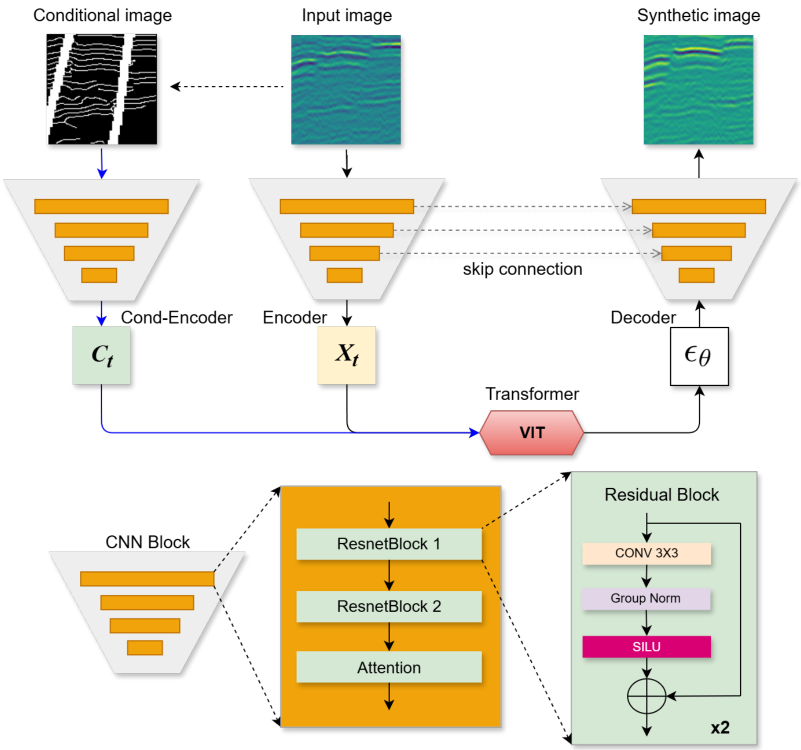

Figure 3: The overall network structure of CoSeDif.

To evaluate the performance of CoSeDif and compare it with other models, we used three metrics: Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Dissimilarity Index (DSSIM). These metrics assess the fidelity of generated images in terms of pixel-level accuracy, perceptual quality, and structural similarity.

Mean Squared Error (MSE): MSE measures the average squared difference between the pixel values of the generated image $\hat{y}$ and the target image y. It is calculated as:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{6}$$

where N is the total number of pixels. A lower MSE indicates a closer match to the target image.

Peak Signal-to-Noise Ratio (PSNR): PSNR quantifies the perceptual quality of the generated image by comparing its maximum possible pixel value to the reconstruction error. It is defined as:

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{\text{Max}^2}{\text{MSE}}\right) \tag{7}$$

where Max is the maximum pixel value in the image. Higher PSNR values indicate better quality.

Structural Dissimilarity Index (DSSIM): DSSIM evaluates the structural dissimilarity between the generated image $\hat{y}$ and the target image y, where a lower value indicates greater structural similarity. DSSIM is derived from the Structural Similarity Index (SSIM), which measures similarity in terms of luminance, contrast, and structure. Unlike simpler metrics like MSE, which only look at pixel-level differences, SSIM tries to mimic how humans perceive image quality. SSIM is defined as:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{8}$$

where μ_x, μ_y are the mean intensities of x and y. σ_x^2, σ_y^2 are the variances of x and y. σ_{xy} is the covariance between x and y. $C_1 = (K_1L)^2$ and $C_2 = (K_2L)^2$ are stabilization constants, where L is the dynamic range of pixel values, and $K_1 = 0.01$, $K_2 = 0.03$. DSSIM is then calculated as: DSSIM = 1 - SSIM. This value reflects how structurally dissimilar the images are, with lower DSSIM indicating higher similarity.

4 Experiments and Results

4.1 Training Preparation

In this experiment, we use the Thebe dataset [35], a public dataset from the Thebe gas field in the Exmouth Plateau of the Carnarvon Basin, located on the northwest shelf of Australia. The dataset has been manually labeled by expert interpreters from the Fault Analysis Group at University College Dublin. Following the original dataset structure, we split the data into training, validation, and test sets with proportions of 5, 1, and 4, respectively. The dataset dimensions are 1803 [crossline] $\times$ 3174 [inline] $\times$ 1537 [sample], and the data is saved in NumPy array format.

For training the diffusion model, we adopt a slicing window approach with a patch size of 128×128 and a step size of 64. Fault labels are defined as binary values, where 1 indicates the presence of a fault and 0 indicates the absence of faults. To ensure high-quality samples during preprocessing stage, patches containing less than 3% faults are excluded. Specifically, we extract 128×128 patches from every 10th crossline in the training set, resulting in 12133 images and their corresponding conditions. This approach leverages approximately 10% of the training set, significantly reducing computational load and training time while demonstrating that a smaller subset of data is sufficient to generate high-quality synthetic data effectively. For testing, we extract patches from the testing set at every crossline, yielding 143560 images.

To create the conditional images for our diffusion model, the Canny parameters are set with σ=3, a low threshold of 0.25, and a high threshold of 0.75 on the original image. These settings are chosen to capture prominent edges without introducing excessive noise or over-complicating the condition. The edge map obtained is then combined with the fault mask as mentioned in 3.2.

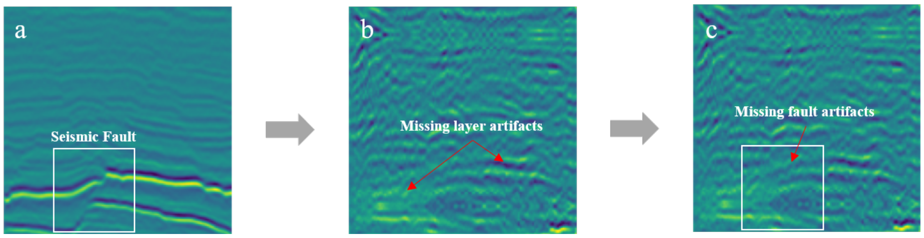

Figure 4: The problem with diffusion models that directly use real seismic data is that they cause important seismic layer artifacts to be lost in the resulting results. (a) Original seismic with fault, (b) synthetic with missing seismic layers and (c) synthetic without seismic fault missing.

Through our observations, when we trained a model using only fault maps as input (see Figure 4), the model's generalization capability was limited, resulting in sub-optimal performance. But when we condition the model on both fault and edge maps significantly enhances its ability to understand spatial relationships and geological patterns. This richer context allows the diffusion model to generalize more effectively. In fact, the addition of edge maps addressed this limitation, allowing the model to generate more realistic images that better reflect diverse geological features.

4.2 Implementation

To evaluate the effectiveness of CoSeDif, we employed two evaluation perspectives: (1) Generative evaluation, where we assessed the generative fidelity of CoSeDif by comparing it against state-of-the-art models, Pix2Pix [41] and ControlNet [42]; and (2) Segmentation evaluation, where we measured the impact of CoSeDif-generated data on fault detection accuracy, comparing it to real data and other synthetic datasets. To ensure a fair comparison, all models were trained on the same dataset splits, with identical hardware configurations and carefully tuned hyperparameters. All experiments were implemented using PyTorch, with training and testing conducted on an NVIDIA A6000 GPU equipped with 16 GB of RAM.

CoSeDif was trained with carefully selected hyperparameters to balance computational efficiency and performance. The learning rate was set to 0.0005, and the Adam optimizer was configured with $\beta_1 = 0.95$, $\beta_2 = 0.999$, a weight decay of 1×10^{-6}, and $\epsilon = 1 \times 10^{-8}$. To enhance computational efficiency, mixed precision with BF16 was employed. The batch size was set to 64, and the model converged after 200 epochs using $1{,}000$ diffusion timesteps during training. For sampling, we used the DDIM algorithm with 100 timesteps. In contrast, Pix2Pix was trained using the default settings provided in its original implementation [41]. A batch size of 64 was maintained, and the model was trained for 200 epochs. No modifications were made to its default architecture or loss functions to preserve consistency with its standard usage.

ControlNet, on the other hand, required additional adaptation to the seismic dataset due to its significant differences from the large-scale natural image datasets used in pre-training. First, the variational autoencoder (VAE) was fine-tuned on the seismic dataset for 200 epochs. Next, the Stable Diffusion (SD) model was fine-tuned for 400 epochs. Finally, ControlNet itself was fine-tuned on the seismic images and their corresponding conditions for an additional 400 epochs. During this process, empty prompts were used for text inputs, and Stable Diffusion 2 served as the base diffusion model. For all stages of training, the batch size was set to 64, and the learning rate was 1×10^{-5}. These fine-tuning steps were essential to adapt ControlNet effectively to the seismic domain, ensuring a meaningful and fair comparison with CoSeDif and Pix2Pix.

4.2.1 Generative Evaluation

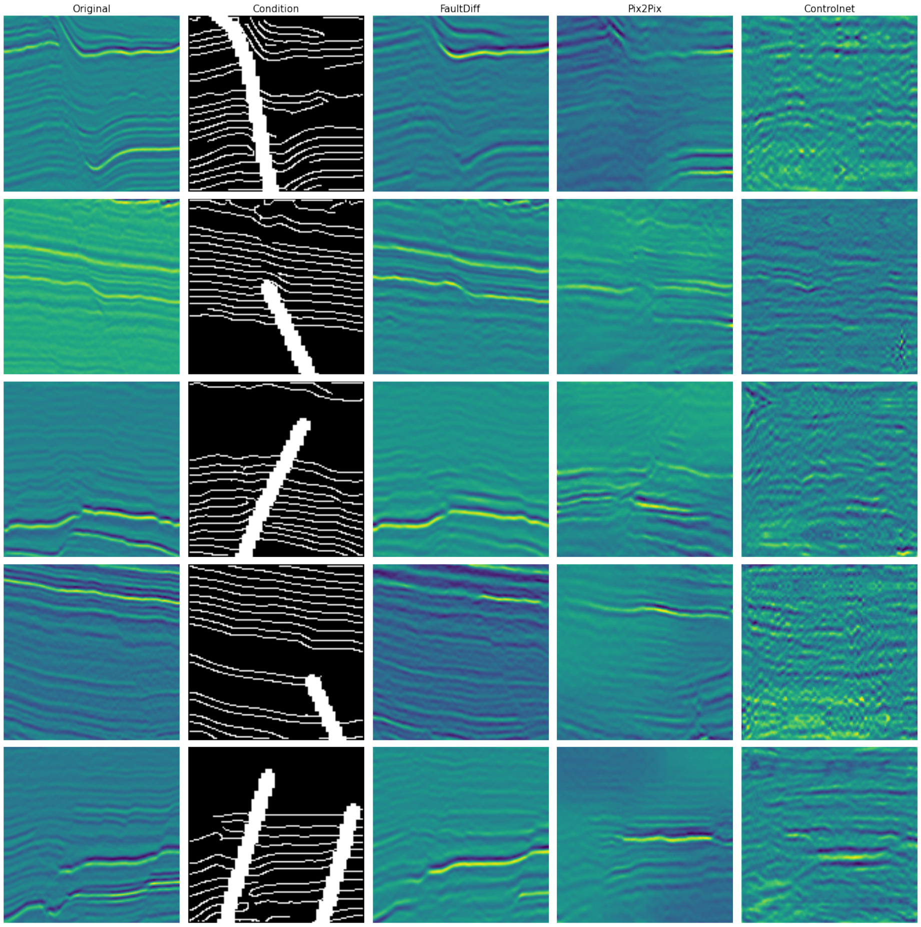

Figure 5: Visual comparison of seismic image synthesis across different models. The first column represents the original seismic image with faults (a). In (b), the fault label is shown in the corresponding image. The synthetic images from CoSeDif (c), Pix2Pix (d), and ControlNet (e) are presented, respectively.

Figure 5 presents a visual comparison of the generated images from CoSeDif, Pix2Pix, and ControlNet, highlighting their performance under identical conditions. Quantitative and qualitative analyses reveal that CoSeDif consistently outperforms the other models in generating seismic images with higher fidelity to the original data. Specifically, CoSeDif achieves the lowest MSE score of 0.0028, which is approximately 47% lower than ControlNet and 47% lower than Pix2Pix, indicating superior pixel-level accuracy in reconstruction. Additionally, the PSNR of CoSeDif is 26.74 dB, significantly higher than Pix2Pix (23.02 dB) and ControlNet (23.54 dB), reflecting its ability to preserve both the overall structure and subtle details of seismic patterns.

The DSSIM score further emphasizes CoSeDif's advantage, with a score of 0.1905, which is 11% lower than Pix2Pix and 14% lower than ControlNet. These quantitative results correlate strongly with the visual observations where the synthetic images from CoSeDif exhibit smoother transitions in seismic reflections, sharper fault structures, and fewer artifacts compared to the other models. This is largely due to the combination of ResNet-based encoders and a transformer block in CoSeDif, which effectively captures both local spatial features and global dependencies.

Pix2Pix, while able to produce outputs that align with the given conditions, struggles to maintain fault sharpness and introduces noticeable distortions in regions of high complexity, as seen in Figure 5. The U-Net backbone and adversarial loss function used in Pix2Pix appear less effective in preserving structural details, which is reflected in its higher MSE and DSSIM values.

ControlNet exhibits the most significant challenges in this comparison. Its reliance on pre-training with natural image datasets limits its ability to adapt to the unique characteristics of seismic data. This results in pronounced artifacts and noise, particularly in regions with abrupt geological transitions, which are critical in seismic imaging. Furthermore, its text-based conditioning mechanism, designed for descriptive prompts, is not fully utilized in this task, as the seismic domain lacks appropriate textual counterparts. These shortcomings are reflected in its highest MSE (0.0066) and DSSIM (0.2218) scores, along with its relatively low PSNR (23.54 dB). Table 1 illustrates quantitative results in terms of generation metrics. CoSeDif outperforms the other models by up to 25% in MSE, achieving the lowest error rate 3.72 db in PSNR and 2%in less in DSSIM.

Table 1: Generation results for CoSeDif, Pix2Pix, and ControlNet.

Model	**MSE**	**PSNR**	**DSSIM**
CoSeDif	**0.0028**	**26.74**	**0.1905**
Pix2Pix	0.0053	23.02	0.2143
ControlNet	0.0066	23.54	0.2218

4.3 Generative Fidelity

To evaluate the generative fidelity of our synthetic seismic images, we assess how closely these images match the distribution of field seismic data. We use the Fréchet Inception Distance (FID)[43] to quantify this resemblance. We calculate the FID between features extracted from the last average pooling layer of the Inception-v3 model. we use the Thebe dataset [35] and the F3 dataset [44] as our real data. The F3 dataset serves as an unbiased benchmark, as it was not involved in the training of our diffusion model. This allows us to evaluate the performance of our synthetic images more impartially. We compare our synthetic data with those generated by the method introduced by Wu et al. [29].

Table 2: FID scores between synthetic datasets and real datasets.

Synthetic Dataset	**Real Dataset**	**FID Score**
FaultDiff	Thebe	6.47562
	F3	6.47568
FaultSeg3d	Thebe	130.12766
	F3	130.12789

Table 2 presents the FID scorer where our approach achieves notably low FID scores for both the Thebe and F3 datasets, with values of 6.47562 and 6.47568, respectively. These results indicate that our synthetic images closely resemble real field data in terms of feature distribution, reflecting its high generative fidelity. In contrast, the FaultSeg3D method [29] yields significantly higher FID scores, indicating that its synthetic images diverge considerably from the real data distributions. This high FID score is likely due to its simulation-based approach, explaining the poor generalization in models trained on such data.

4.4 Segmentation Perspective

To evaluate the effectiveness of our synthetic data in fault detection, we measure the ability of segmentation models trained solely on synthetic data to generalize to real field data. We compare the performance of segmentation models, based on a 2D U-Net architecture, trained on real data (Thebe) and synthetic data (CoSeDif and FaultSeg3D).

For the Thebe dataset, we used the slicing window approach described in 4.1 to extract 120,976 images of size 128x128 from every crossline in the training set and generate an equivalent number of 128x128 images from the respective conditions. Additionally, we experiment with training on a reduced dataset of synthetic images by generating images conditioned on patches from every 5th crossline, resulting in 24338 images less than 5x the original trainset size.

For FaultSeg3D [29], we convert every crossline and inline profile into 2D images, resulting in 51200 images of size 128x128. The models are tested on the Thebe test set consisting of 143560 images. We trained each model for 100 epochs with a batch size of 16 and a learning rate of 10^{-3}. Evaluation is performed using standard metrics, including accuracy, precision, recall, average precision (AP), and area under the curve (AUC).

Table 3: Segmentation performance metrics for real and synthetic data.

Training Set	Strategy	Accuracy	Precision	Recall	AUC	AP
Thebe	Real	0.9316	0.6699	**0.5438**	**0.9291**	**0.6640**
CoSeDif	Synthetic	**0.9320**	**0.6873**	0.5142	0.9232	0.6530
CoSeDif	Synthetic (reduced set)	0.9315	0.6689	0.5122	0.9212	0.6330
FaultSeg3D	Synthetic	0.9043	0.4115	0.0322	0.6593	0.1933

The results presented in Table 3 demonstrate the efficacy of the CoSeDif synthetic data in training segmentation models. The model trained on CoSeDif-synthesized images exhibits performance metrics that are not only comparable but marginally superior to those of models trained on real data. Notably, the CoSeDif-trained model achieved an accuracy of 0.9322 and precision of 0.6879, slightly outperforming the model trained on original data (accuracy: 0.9316, precision: 0.6699). Besides, the segmentation model trained on a reduced subset of CoSeDif synthetic data, utilizing fewer crosslines, maintained comparable performance metrics. This reduced dataset model achieved an accuracy of 0.9315 and precision of 0.6689, closely mirroring the performance of models trained on both the full original and synthetic datasets. The consistency in recall measurements (0.5142 for the reduced set vs. 0.5244 for the full set) is particularly noteworthy. These results suggest that a small set of our synthetic data can yield performance comparable to larger datasets, while simultaneously offering advantages in computational efficiency during the training phase.

5 Conclusion

In this work, we introduced CoSeDif, a conditional denoising diffusion model designed to synthesize seismic images for fault detection. The core of our method is a conditional denoising process that incorporates fault segmentation and edge maps extracted from field data. Our evaluation confirms that CoSeDif generates synthetic seismic images that closely match real data distributions. Furthermore, segmentation models trained on CoSeDif data demonstrate competitive performance in fault detection. Looking ahead, we plan to refine and expand CoSeDif by incorporating additional seismic attributes and exploring multi-modal generation. Future work will also focus on extending CoSeDif to 3D. This study lays a foundation for advancing synthetic seismic data generation and its integration into practical fault detection and analysis workflows.

References

[1] S Mostafa Mousavi and Gregory C Beroza. Deep-learning seismology. *Science*, 377(6607):4470, 2022.

[2] Adrian Bulat, Jing Yang, and Georgios Tzimiropoulos. To learn image super-resolution, use a gan to learn how to do image degradation first. In *Proceedings of the European conference on computer vision (ECCV)*, pages 185–200, 2018.

[3] Yuzhen Niu, Jianbin Wu, Wenxi Liu, Wenzhong Guo, and Rynson WH Lau. Hdr-gan: Hdr image reconstruction from multi-exposed ldr images with large motions. *IEEE Transactions on Image Processing*, 30:3885–3896, 2021.

[4] Dario A. B. Oliveira, Rodrigo S. Ferreira, Reinaldo Silva, and Emilio Vital Brazil. Improving seismic data resolution with deep generative networks. *IEEE Geoscience and Remote Sensing Letters*, 16(12):1929–1933, 2019.

[5] Haihang Zhang, Guangzhi Zhang, Jianhu Gao, Shengjun Li, Jinmiao Zhang, and Zhenyu Zhu. Seismic impedance inversion based on geophysical-guided cycle-consistent generative adversarial networks. *Journal of Petroleum Science and Engineering*, 218:111003, 2022.

[6] Hao-Ran Zhang, Yang Liu, Yu-Hang Sun, and Gui Chen. Seisresodiff: Seismic resolution enhancement based on a diffusion model. *Petroleum Science*, 2024.

[7] Paul Goyes-Peñafiel, León Suárez-Rodríguez, Claudia V. Correa, and Henry Arguello. Gan supervised seismic data reconstruction: An enhanced learning for improved generalization. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–10, 2024.

[8] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, 2105.05233, 2021.

[9] Hoang Thanh-Tung and Truyen Tran. Catastrophic forgetting and mode collapse in gans. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10, 2020.

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[11] Mike Bahorich and Steve Farmer. 3-d seismic discontinuity for faults and stratigraphic features: The coherence cube. *The Leading Edge*, 14(10):1053–1058, 1995.

[12] Arthur E. Barnes. Weighted average seismic attributes. *Geophysics*, 65(1):275–285, 2000.

[13] Andy Roberts. Curvature attributes and their application to 3d interpreted horizons. *First Break*, 19:85–100, 02 2001.

[14] Nasher M AlBinHassan and Kurt Marfurt. Fault detection using hough transforms. In *SEG International Exposition and Annual Meeting*, pages 1719–1721. SEG, 2003.

[15] Zhen Wang and Ghasssan AlRegib. Automatic fault surface detection by using 3d hough transform. In *SEG International Exposition and Annual Meeting*, pages 1439–1444. SEG, 2014.

[16] Stein Inge Pedersen, Trygve Randen, Lars Sønneland, and Øyvind Steen. Automatic fault extraction using artificial ants. In *SEG International Exposition and Annual Meeting*, pages 512–515. SEG, 2002.

[17] Dave Hale. Structure-oriented smoothing and semblance. *CWP report*, 635(635), 2009.

[18] Ahmed Adnan Aqrawi and Trond Hellem Boe. Improved fault segmentation using a dip guided and modified 3d sobel filter. In *SEG Technical Program Expanded Abstracts 2011*, pages 999–1003. Society of Exploration Geophysicists, 2011.

[19] Mehran Rahimi and Mohammad Ali Riahi. Reservoir facies classification based on random forest and geostatistics methods in an offshore oilfield. *Journal of Applied Geophysics*, 201:104640, 2022.

[20] Isack Farady, Chia-Chen Kuo, Sang Le, Chih-Wei Wang, Hui-Fuang Ng, Chih-Yang Lin, and Ming-Jen Wang. Seismic layer segmentation models with channel attention block in carbon storage study. In *Proceedings of the 2nd Workshop on Advances in Environmental Sensing Systems for Smart Cities*, pages 1–6, 2024.

[21] Qiankun Feng and Yue Li. Denoising deep learning network based on singular spectrum analysis—das seismic data denoising with multichannel svddcnn. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2021.

[22] Ming Cheng, Jun Lin, Shaoping Lu, Shiqi Dong, and Xintong Dong. Seismic data reconstruction based on multiscale attention deep learning. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[23] Tao Zhao and Pradip Mukhopadhyay. A fault detection workflow using deep learning and image processing. In *SEG International Exposition and Annual Meeting*, pages 1966–1970. SEG, 2018.

[24] Zhanxin Tang, Bangyu Wu, Weihua Wu, and Debo Ma. Fault detection via 2.5 d transformer u-net with seismic data pre-processing. *Remote Sensing*, 15(4):1039, 2023.

[25] Zhiwei Wang, Jiachun You, Wei Liu, and Xingjian Wang. Transformer assisted dual u-net for seismic fault detection. *Frontiers in Earth Science*, 11:1047626, 2023.

[26] Zeren Zhang, Ran Chen, and Jinwen Ma. Improving seismic fault recognition with self-supervised pre-training: a study of 3d transformer-based with multi-scale decoding and fusion. *Remote Sensing*, 16(5):922, 2024.

[27] Ran Chen, Zeren Zhang, and Jinwen Ma. Seismic fault sam: Adapting sam with lightweight modules and 2.5d strategy for fault detection, 2024.

[28] Yu An, Haiwen Du, Siteng Ma, Yingjie Niu, Dairui Liu, Jing Wang, Yuhan Du, Conrad Childs, John Walsh, and Ruihai Dong. Current state and future directions for deep learning based automatic seismic fault interpretation: A systematic review. *Earth-Science Reviews*, 243:104509, 2023.

[29] Xinming Wu, Luming Liang, Yunzhi Shi, and Sergey Fomel. Faultseg3d: Using synthetic data sets to train an end-to-end convolutional neural network for 3d seismic fault segmentation. *Geophysics*, 84(3):35–45, 2019.

[30] Shenghou Wang, Xu Si, Zhongxian Cai, and Yatong Cui. Structural augmentation in seismic data for fault prediction. *Applied Sciences*, 12(19), 2022.

[31] Jiankun Jing, Zhe Yan, Zheng Zhang, Hanming Gu, and Bingkai Han. Fault detection using a convolutional neural network trained with point-spread function-convolution-based samples. *Geophysics*, 88(1):IM1–IM14, 2023.

[32] Xinming Wu, Zhicheng Geng, Yunzhi Shi, Nam Pham, Sergey Fomel, and Guillaume Caumon. Building realistic structure models to train convolutional neural networks for seismic structural interpretation. *GEOPHYSICS*, 85(4):WA27–WA39, 2020.

[33] Yimin Dou, Kewen Li, Jianbing Zhu, Xiao Li, and Yingjie Xi. Attention-based 3-d seismic fault segmentation training by a few 2-d slice labels. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.

[34] Lei Lin, Zhi Zhong, Chenglong Li, Andrew Gorman, Hao Wei, Yanbin Kuang, Shiqi Wen, Zhongxian Cai, and Fang Hao. Machine learning for subsurface geological feature identification from seismic data: Methods, datasets, challenges, and opportunities. *Earth-Science Reviews*, 257:104887, 2024.

[35] Yu An, Jiulin Guo, Qing Ye, Conrad Childs, John Walsh, and Ruihai Dong. Deep convolutional neural network for automatic fault recognition from 3d seismic datasets. *ComputersGeosciences*, 153:104776, 2021.

[36] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[37] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models, 2023.

[38] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors, 2022.

[39] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.

[40] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[41] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[42] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

[43] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[44] Yazeed Alaudah, Patrycja Michałowicz, Motaz Alfarraj, and Ghassan AlRegib. A machine-learning benchmark for facies classification. *Interpretation*, 7(3):SE175–SE187, 2019.