Research Square

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

Time-aligned fusion of temporal attention and convolutional features for seismic velocity inversion with limited training data

Md Mahabube Alahi Atik
North South University
Abdul Ahad
ahadraj@ieee.org

North South University https://orcid.org/0009-0005-1253-5314

Research Article

Keywords: Full-waveform inversion, Seismic velocity modeling, Deep learning, Attention, OpenFWI, Uncertainty quantification

Posted Date: November 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11861251/v1

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

Additional Declarations: The authors declare no competing interests.

Time-aligned fusion of temporal attention and convolutional features for seismic velocity inversion with limited training data

Md Mahabube Alahi Atik[1], Abdul Ahad[1*]

1 Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh
**Corresponding author: Abdul Ahad | ahad.abdul01@northsouth.edu*

Abstract

Data-driven seismic full-waveform inversion estimates subsurface velocity from seismic recordings, but comparisons between architectures can be confounded by training protocols. We investigate DSUNet, a convolutional encoder–decoder that uses a bidirectional LSTM and multi-head self-attention to encode source gathers and fuse temporal features along the bottleneck time axis. On a synthetic, file-disjoint subset of ten OpenFWI-derived families (4,250 training and 5,000 test samples), DSUNet achieved a mean absolute error (MAE) of 214.6 ± 0.3 m/s over three seeds. The official InversionNet, retrained on the same split under four protocols, achieved 235.1–252.7 m/s. DSUNet reduced MAE by 8.7% relative to the strongest protocol, with 11.7 times fewer parameters, and outperformed it on 70–74% of test samples. Removing the temporal branch increased MAE by 20.5%; replacing time-aligned fusion with band-wise fusion increased it by 8.4%. Under additive Gaussian noise (20–5 dB), DSUNet achieved 25–35% lower MAE than an input-normalization-matched InversionNet. A three-member ensemble reduced MAE to 204.4 m/s; ensemble spread tracked errors within families (r = 0.47–0.76) and identified the worst 10% of predictions (AUROC 0.93). Removing two physics-inspired penalties jointly did not measurably affect accuracy. DSUNet requires at least 12.5 times more operations and 4.1 times longer inference than InversionNet. These findings establish a limited-data synthetic benchmark result, not field-data performance or superiority to other inversion methods.

Keywords: Full-waveform inversion; Seismic velocity modeling; Deep learning; Attention; OpenFWI; Uncertainty quantification

1 Introduction

Seismic full-waveform inversion (FWI) estimates subsurface velocity by matching observed and simulated waveforms (Tarantola 1984; Virieux and Operto 2009). It can resolve fine structure, but it is nonlinear, sensitive to the starting model and computationally expensive (Virieux and Operto 2009). Data-driven inversion replaces the iterative solver with a network trained to map seismic recordings directly to velocity models (Araya-Polo et al. 2018; Yang and Ma 2019; Wu and Lin 2020), and the OpenFWI benchmark (Deng et al. 2022) has made such models comparable across ten structural families.

Three practical questions remain open. First, published architectures are usually trained on tens of thousands of samples per family (Deng et al. 2022; Jin et al. 2024), whereas field projects rarely have that much labelled data; how architectures rank when training data are limited is less well documented. Second, new architectures are often compared with baselines trained under their own default recipes, so the effect of the architecture and the effect of the training recipe are confounded. Third, claims of robustness to noise and of benefits from physics-inspired loss terms are rarely tested against controls that isolate their cause.

This study investigates whether preserving the temporal correspondence between sequence features and convolutional feature maps improves supervised seismic velocity estimation under limited labelled training data. DSUNet combines a convolutional encoder–decoder with a bidirectional LSTM (Hochreiter and Schmidhuber 1997) and multi-head self-attention (Vaswani et al. 2017). We compare it with the official InversionNet implementation (Wu and Lin 2020; Los Alamos National Laboratory 2022) on an identical file-disjoint split, using four training protocols and three seeds per protocol. The principal contributions are:

- **Time-aligned fusion.** A fusion scheme that keeps the temporal features aligned with the image time axis. Ablations show that removing the temporal branch raises the test error by 20.5% and that the earlier band-wise fusion raises it by 8.4%.
- Controlled baseline comparison. DSUNet's MAE is 8.7% lower than the strongest of four InversionNet protocols, with 11.7 times fewer parameters but greater computation. Changing InversionNet's recipe lowers its error by 7.0%, demonstrating the influence of training choices.
- Noise robustness with a preprocessing control. Under additive Gaussian noise, DSUNet has 25–35% lower error than InversionNet with matched input normalization; other training differences remain.
- Ensemble uncertainty. A three-seed ensemble lowers MAE to 204.4 m/s. Its spread correlates with error within each family (r = 0.47–0.76) and flags the worst predictions (AUROC 0.93).
- **A data check of a physics prior.** About half of the depth transitions in two families are velocity decreases, violating a monotonic depth prior, and removing the two physics-inspired penalties jointly does not change accuracy.
- Reproducibility. The reported experiments use deterministic settings, explicit file-disjoint splits and logged runs with fixed seeds.

Section 2 reviews related work. Section 3 describes the model and objective, Section 4 the data and protocol, and Section 5 the results. Section 6 discusses the findings and their limitations, and Section 7 concludes.

2 Related Work

This section positions DSUNet relative to classical waveform inversion, supervised and unsupervised neural inversion, and physics-guided methods. Only InversionNet was retrained on the present split; reported results from other studies are context rather than head-to-head comparisons.

2.1 Classical Full-Waveform Inversion and Its Limitations

FWI estimates subsurface parameters by iteratively minimizing the misfit between observed and simulated waveforms; the modern least-squares, adjoint-based formulation was introduced by Tarantola (1984):

$$v^{*} = arg\ min_{v}\ J(v) = \tfrac{1}{2}\ \Sigma_{j,k}\ \| F[v](s_j, r_k) - d_{jk}^{obs} \|_2^2 \quad (1)$$

where $\mathscr{F}$ is the forward operator (a wave-equation solver). Virieux and Operto (2009) review the method for exploration geophysics and identify the lack of low frequencies and the need for an accurate starting model as central obstacles to convergence, because both govern cycle skipping: when simulated and observed events are more than half a period apart, local optimization converges to a wrong minimum. Multiscale strategies that proceed from low to high frequencies partly mitigate this problem (Bunks et al. 1995).

Computation time. In the OpenFWI comparison, multi-scale FWI required about 364 s, 370 s and 222 s per sample on the Vel, Fault and Style families and 48 h per sample for the three-dimensional Kimberlina-V1 model, whereas InversionNet inference took 0.3–0.33 s per two-dimensional sample (Deng et al. 2022). These figures, obtained on GPU hardware for a 500-sample subset, indicate the order of magnitude of the cost gap rather than a controlled benchmark.

2.2 Data-Driven FWI: End-to-End CNN Architectures

Early work showed that deep networks can map seismic data directly to velocity models (Araya-Polo et al. 2018; Yang and Ma 2019). InversionNet (Wu and Lin 2020) uses a CNN encoder–decoder to map multi-source seismic data to velocity maps; the original paper adds a conditional random field for structured prediction, whereas the OpenFWI benchmark implementation is purely convolutional (Deng et al. 2022). VelocityGAN (Zhang et al. 2019; Zhang and Lin 2020) adds a CNN discriminator to an InversionNet-

style generator; in the OpenFWI implementation it is trained with a Wasserstein loss with gradient penalty plus pixel-wise L1 and L2 losses (Deng et al. 2022).

OpenFWI (Deng et al. 2022) provides 12 datasets (2.1 TB in total, 2K–67K samples per dataset; eleven two-dimensional and one three-dimensional) covering flat, curved, faulted, style-transferred and CO2-reservoir structures. Its two-dimensional benchmarks include InversionNet, VelocityGAN and the unsupervised UPFWI (Jin et al. 2022), alongside analyses of complexity, generalization and uncertainty.

Reported accuracy. On the two-dimensional OpenFWI datasets, InversionNet's MAE ranges from 0.011 (FlatVel-A) to 0.167 (CurveFault-B) and VelocityGAN's from 0.012 to 0.158, with accuracy degrading on the harder "B" versions (Deng et al. 2022); these values are computed on velocity normalized to [−1, 1]. Jin et al. (2024) showed that training on 470K OpenFWI pairs improves MAE by 13.03% on average over single-dataset training and by 28.60% in leave-one-out generalization, and that larger data require larger models. Generalization to out-of-distribution data remains a key weakness of data-driven FWI (Deng et al. 2022; Deng and Lin 2021). Physics-guided data-driven FWI has been reviewed by Lin et al. (2023).

2.3 Attention Mechanisms in Seismic Velocity Inversion

Attention has been added to U-shaped inversion networks to capture long-range dependencies. The CLR-VMB framework (Zhu et al. 2025) embeds dual attention modules in a U-shaped ResNet (DA-ResNet) to extract cross-gather relationships and ensembles network snapshots obtained during a cyclic learning-rate schedule.

AER-UNet (Liu et al. 2026) uses self-attention skip connections and is evaluated on slices of the SEG/EAGE Overthrust model. FAR-UNet (Wei et al. 2026) uses multi-head attention with relative position biases, acting as adaptive spectral filters, and reports a PSNR of 27.20 dB and an SSIM of 68.30% on its simulated dataset (3.20 dB and 8.87% above Swin-Unet) at 19.54 GFLOPs. Outside geophysics, Islam et al. (2025) combine attention-enhanced bidirectional LSTM with stacking for temporal classification; Akter et al. (2024) compare convolutional and recurrent architectures in biomedical prediction. Fahmid et al. (2025) use convolutional features and probabilistic pooling for image localization. These studies illustrate distinct uses of sequence modelling and spatial feature aggregation; none evaluates seismic inversion. DSUNet instead preserves temporal correspondence when fusing the two feature streams for velocity regression.

2.4 Transformer-Based Approaches for FWI

Transformer architectures (Vaswani et al. 2017) have also been applied to velocity-model generation. VelocityGPT (Harsuko et al. 2025) uses Transformer decoders trained autoregressively to generate velocity models from shallow to deep, conditioned on well or migration-image information.

In reparameterized FWI, a network generates the velocity model, which is then updated through a wave-equation solver. Geng et al. (2026) combine a CNN with a Vision Transformer using spatial-reduction attention and exploit the untrained network as an implicit regularizer.

2.5 Physics-Guided and Physics-Informed Deep Learning

Physics-informed neural networks have been applied to the acoustic wave equation for wave propagation and FWI (Rasht-Behesht et al. 2022). Physics-informed recurrent networks embed finite-difference wave propagation in the recurrent structure; Lu et al. (2025) add first-arrival constraints to make vertical-seismic-profiling FWI insensitive to the initial model. PGFRNN (Chen et al. 2026) embeds Fourier-transformed seismic data in a latent space and updates the velocity with a softplus-approximated log-cosh (SALC) loss and a physics-guided optimizer, for both unsupervised and simultaneous-source FWI. FWIGAN (Yang and Ma 2023) provides an example of an adversarial physics-informed approach.

Methods of this kind must balance data misfit against physical constraints, and embedding a wave-equation solver in training or inference adds cost per iteration. The regularizers used in the present work

are much weaker: they are computed directly on the predicted velocity map and involve no wave simulation (Section 3.3).

2.6 Unsupervised and Self-Supervised Approaches

Jia et al. (2025) propose a variational-autoencoder-based probabilistic FWI that quantifies uncertainty and report that it mitigates the tendency of dropout-based methods to underestimate uncertainty in deeper layers. Wang et al. (2026) perform unsupervised inversion in the frequency domain with a Wasserstein GAN and gradient penalty. These approaches differ from DSUNet, which uses labelled velocity maps for training.

2.7 Positioning of this work

Hybrid convolutional and attention-based inversion networks already exist (Zhu et al. 2025; Liu et al. 2026; Wei et al. 2026; Geng et al. 2026). The contribution here is a specific, experimentally tested time-aligned fusion design with deep supervision, not a new class of architecture. DSUNet estimates velocity directly from seismic gathers without a wave-equation solver. Its two auxiliary penalties are physics-inspired rather than wave-equation constraints. The following table summarizes the relevant method families and the scope of the comparison; published accuracy values cannot be ranked against the present results because training data, preprocessing and evaluation protocols differ.

Table 1. Methodological comparison with existing seismic inversion approaches (literature context; not a numerical benchmark).

Method / reference	Inversion strategy	Comparison in this study
Classical multiscale FWI (Bunks et al. 1995; Virieux and Operto 2009)	Iterative wave-equation misfit minimization; requires starting model.	Literature only; no solver-based baseline.
InversionNet (Wu and Lin 2020)	Supervised CNN encoder–decoder; direct velocity regression.	Retrained; four protocols, three seeds.
VelocityGAN (Zhang et al. 2019; Zhang and Lin 2020)	Adversarial supervised velocity inversion.	Literature only; not retrained.
UPFWI (Jin et al. 2022)	Unsupervised inversion using a CNN and wave-equation solver.	Literature only; not retrained.
CLR-VMB (Zhu et al. 2025)	Dual-attention U-shaped ResNet and snapshot ensemble.	Literature only; not retrained.
DSUNet (this study)	Time-aligned BiLSTM/self-attention and CNN feature fusion.	Synthetic benchmark; no field-data evaluation.

3 Method

3.1 Problem formulation

Each sample consists of seismic data $d \in \mathbb{R}^{S\times T\times R}$, with S = 5 sources, T = 1000 time samples and R = 70 receivers, and a velocity map $v \in \mathbb{R}^{70\times 70}$ indexed by depth z and lateral position x. The network f_θ predicts $\hat{v} = f_\theta(d)$, an estimate of the normalized velocity $\tilde{v}$; velocities are min–max normalized with the training range (v_{min} = 1500 m/s, v_{max} = 4500 m/s), so one normalized unit corresponds to 3000 m/s. Each source gather is standardized to zero mean and unit variance before entering the network.

3.2 DSUNet architecture

Figure 1 summarizes the convolutional encoder–decoder, the temporal branch and their bottleneck fusion.

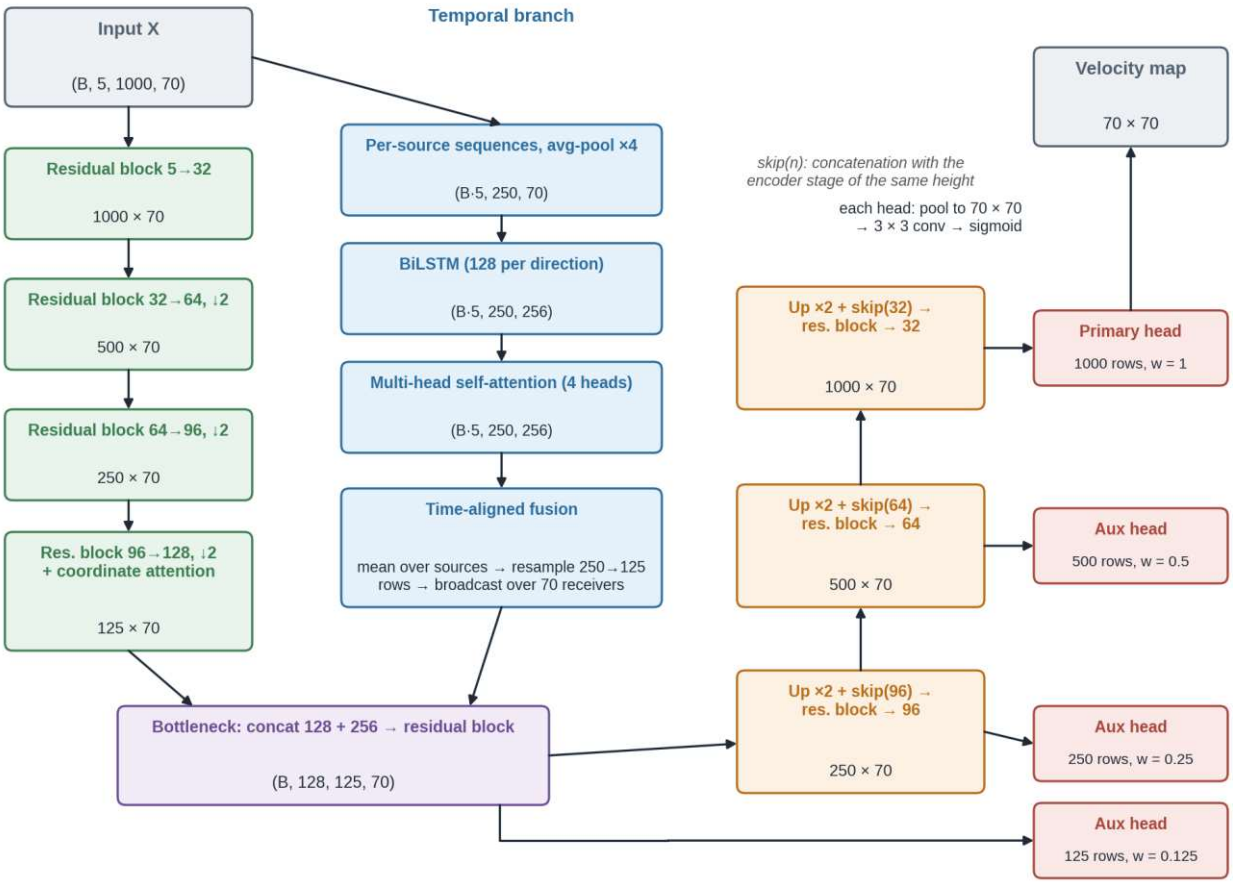


Fig. 1 DSUNet. The encoder (left) halves the time axis three times while keeping the 70 receivers; the temporal branch (centre) encodes each source gather as a sequence and is fused into the bottleneck along the time axis; the decoder (right) restores the 1000-row resolution through skip connections. Four heads, each pooled to 70 × 70, provide deep supervision with the weights shown. Shapes are given as (batch, channels or sequence, time, receivers)

Encoder. Four residual blocks (He et al. 2016), each with two 3 × 3 convolutions, batch normalization and ReLU, map the five source gathers (treated as input channels) to 32, 64, 96 and 128 channels. Blocks two to four use stride 2 along time only, reducing the time axis from 1000 to 500, 250 and 125 rows while preserving the 70 receiver columns. A simplified coordinate-attention block (Hou et al. 2021) reweights the deepest features with separate sigmoid gates along the time and receiver axes.

Temporal branch. Each source gather is treated as a sequence of time steps whose 70 receiver amplitudes form the feature vector. After average pooling with stride 4 (250 steps), a bidirectional LSTM with 128 units per direction and a four-head self-attention layer (model width 256) produce a sequence of 256-dimensional features for each source.

Time-aligned fusion. The attended features are averaged over the five sources, linearly resampled along time from 250 steps to the bottleneck height H = 125, and broadcast across the 70 receiver columns:

$$F_t = Resample_H((1/S)\ \Sigma_s\ MHSA(BiLSTM(Pool_4(d_s)))) \otimes 1_R \qquad (2)$$

so that row h of the temporal feature map summarizes the same time window as row h of the convolutional feature map. F_t (256 channels) is concatenated with the encoder output (128 channels) and passed through a residual block. An earlier variant averaged each source's sequence over time and stretched the five resulting vectors over five bands of rows (band fusion); the ablation in Section 5.4 compares the two.

Decoder and heads. Three decoder stages upsample the time axis by two with bilinear interpolation, concatenate the encoder features of the same height (U-Net skip connections (Ronneberger et al. 2015)) and apply a residual block, giving 96, 64 and 32 channels at 250, 500 and 1000 rows. Every head applies adaptive average pooling to 70 × 70, a 3 × 3 convolution and a sigmoid. The primary head reads the 1000-row decoder output; auxiliary heads read the 500- and 250-row decoder outputs and the bottleneck. The

model has 2,078,868 trainable parameters; removing the temporal branch, together with the extra bottleneck input channels, removes 812,032 of them. Pooling the time axis to 70 depth rows implicitly assumes that later arrivals carry information about deeper structure; the network learns the actual mapping.

3.3 Training objective

The objective combines a deep-supervised reconstruction term with two auxiliary penalties:

$$L = L_{rec} + \lambda_{CS} L_{CS} + \lambda_{dir} L_{dir}, \quad \lambda_{CS} = 0.10, \ \lambda_{dir} = 0.02 \tag{3}$$

$$L_{rec} = \Sigma_k w_k \| \hat{v}_k - \tilde{v} \|_1 / \Sigma_k w_k, \quad w = (1, 0.5, 0.25, 0.125) \tag{4}$$

where $\hat{v}_k$ are the predictions of the primary and auxiliary heads (Lee et al. 2015) and $\tilde{v}$ is the normalized target; every head is compared with the full 70 × 70 target. L_{CS} compares cumulative slowness $C(z, x) = \Sigma_{z' \le z} 1/v(z', x)$ of prediction and target on the odd columns, normalized by the target's mean total slowness, as a crude travel-time proxy. L_{dir} = mean $\mathrm{ReLU}(\hat{v}(z, x) - \hat{v}(z + 1, x))$ penalizes predicted velocity decreasing with depth. Neither term evaluates a wave equation; Section 5.4 shows that removing both together does not change the accuracy, and Section 5.7 explains why.

4 Experimental Setup

4.1 Data and split

We use the training samples of the Kaggle Yale/UNC-CH – Geophysical Waveform Inversion competition (Kaggle 2025), which are drawn from OpenFWI (Deng et al. 2022): ten families (FlatVel, CurveVel, FlatFault, CurveFault and Style, each in versions A and B), with two files of 500 samples per family. The split is file-disjoint for testing: the second file of every family forms the test set (500 samples per family, 5,000 in total), and the first file is divided at random into 425 training and 75 validation samples (Table 2). Test samples therefore come from files never seen in training. The validation draw depends on the seed; the test set is identical for every run, which enables paired comparisons. All input records and reference velocity maps in this experiment are synthetic. The test split shares the same family definitions and acquisition layout as training; it is not a test of transfer to field data or unseen acquisition geometries.

Table 2 Data split (fold 0). Each family contributes one training/validation file and one test file of 500 samples

Family	Training/validation file	Test file	Training	Validation	Test
FlatVel_A	data1	data2	425	75	500
FlatVel_B	data1	data2	425	75	500
CurveVel_A	data1	data2	425	75	500
CurveVel_B	data1	data2	425	75	500
FlatFault_A	seis2_1_0	seis4_1_0	425	75	500
FlatFault_B	seis6_1_0	seis8_1_0	425	75	500
CurveFault_A	seis2_1_0	seis4_1_0	425	75	500
CurveFault_B	seis6_1_0	seis8_1_0	425	75	500
Style_A	data1	data2	425	75	500
Style_B	data1	data2	425	75	500
Total	**10 files**	**10 files**	**4,250**	**750**	**5,000**

4.2 DSUNet training

DSUNet was trained with AdamW (Loshchilov and Hutter 2019) (learning rate 3×10^{-4}, weight decay 5×10^{-4}, batch size 8) for up to 50 epochs, with a three-epoch linear warm-up (Goyal et al. 2017) followed by cosine decay (Loshchilov and Hutter 2017). Early stopping (Prechelt 1998) monitored the validation MAE

with a patience of eight epochs, and the best validation epoch was restored before testing. Training used mixed precision (Micikevicius et al. 2018), gradient-norm clipping at 1.0 after loss-scale removal (Pascanu et al. 2013), data parallelism over two NVIDIA T4 GPUs and deterministic algorithms. Three seeds (13, 42, 73) were trained; each run took about 3.6 h (about 255 s per epoch). All three runs reached their best validation error at epoch 48.

4.3 Baseline: official InversionNet under four protocols

The baseline is InversionNet (Wu and Lin 2020) as implemented in the official OpenFWI repository (Los Alamos National Laboratory 2022) (24,409,123 parameters), used without architectural change; its tanh output in [−1, 1] is mapped to the [0, 1] target scale. Because a baseline's accuracy depends on how it is trained, we ran four protocols (Table 3), each with the same three seeds and the identical split. The input-matched protocol is a control: it differs from the standard protocol only in the input normalization, which tests the contribution of preprocessing within InversionNet. No protocol simultaneously matches DSUNet's input normalization, loss, batch size and optimizer schedule; residual performance differences therefore cannot be attributed solely to architecture.

Table 3 Training protocols for InversionNet. All use the official architecture and the split of Table 2

Protocol	Input	Loss	Optimizer and schedule	Batch	Epochs (stopping)
Standard (OpenFWI defaults)	log + min–max to [−1, 1]	L1 + L2 on [−1, 1]	AdamW 1×10^{-4}, constant	64	≤ 50 (patience 8)
120-epoch	log + min–max	L1 + L2	AdamW 1×10^{-4}, constant	64	120 (best epoch kept)
Matched recipe	log + min–max	L1 + L2	DSUNet's: AdamW 3×10^{-4}, warm-up + cosine	8	≤ 50 (patience 8)
Input-matched (control)	per-gather standardization	L1 + L2	AdamW 1×10^{-4}, constant	64	≤ 50 (patience 8)

With batch size 8 and two GPUs, the final batch of two samples per epoch was dropped so that each GPU's batch normalization received at least two samples.

4.4 Ablations

Three variants were trained with seed 42 and DSUNet's protocol: without the temporal branch (1,266,836 parameters), with band fusion instead of time-aligned fusion, and without the two auxiliary penalties ($\lambda_{CS} = \lambda_{dir} = 0$). The variant without the temporal branch stopped early at epoch 14 under the patience rule because its validation error fluctuated strongly at the peak learning rate; we therefore retrained it for the full 50 epochs without early stopping, as the full model effectively was, and report that run.

4.5 Evaluation

All metrics are computed on the 5,000 test samples: MAE and root-mean-square error (RMSE) in m/s, the structural similarity index (SSIM) (Wang et al. 2004) on [0, 1]-scaled maps, the peak signal-to-noise ratio and the coefficient of determination R^2. We also report the MAE on odd columns, which is the competition's scoring rule, and the MAE on the [−1, 1] scale used by OpenFWI, for context only.

Statistics. For each comparison, the per-sample MAE of DSUNet and of the comparator with the same seed were compared with the two-sided Wilcoxon signed-rank test (Wilcoxon 1945). The Holm correction (Holm 1979) was applied to the family of all 16 comparisons computed in this study: the 12 comparisons with InversionNet (four protocols × three seeds) and four ablation comparisons (the three variants of Section 4.4 and the early-stopped run without the temporal branch). We also report the fraction of test samples on which DSUNet has the lower error.

Noise robustness. White Gaussian noise was added to the raw seismic data before normalization at signal-to-noise ratios of 20, 10 and 5 dB, with a fixed random stream per sample, and the trained models were evaluated without retraining.

Uncertainty. The three DSUNet seeds form a deep ensemble (Lakshminarayanan et al. 2017). We report the ensemble-mean accuracy, the pixel-wise standard deviation across members, and the Pearson correlation between each sample's ensemble MAE and its mean standard deviation, both pooled and within each family (to exclude differences between families), and the area under the ROC curve (AUROC) for identifying the 10% of samples with the largest error from the spread alone.

Depth-prior check. On 200 training maps per family, we counted the fraction of vertically adjacent cell pairs whose velocity decreases, or increases, by more than 1 m/s with depth. We call such a change a depth transition.

Efficiency. We report parameters, inference latency (batch size 1, one T4 GPU, mean of 50 runs after warm-up), peak GPU memory and floating-point operations counted with PyTorch's FlopCounterMode, which may not count LSTM kernels; DSUNet's count is therefore a lower bound.

5 Results

5.1 Comparison with InversionNet

Table 4 compares DSUNet with InversionNet under the four protocols. DSUNet reached a test MAE of 214.6 ± 0.3 m/s (mean ± standard deviation over three seeds) and an SSIM of 0.596. The strongest baseline was InversionNet trained with DSUNet's recipe (235.1 ± 1.7 m/s); DSUNet's MAE was 8.7% lower, its RMSE 2.6% lower, and it used 11.7 times fewer parameters. Against the standard OpenFWI protocol, the difference was 15.1%. The seed-to-seed spread of DSUNet (0.3 m/s) was an order of magnitude smaller than the differences between models.

Table 4 Test results on 5,000 file-disjoint samples (mean ± standard deviation over seeds 13, 42 and 73)

Model	Parameters	MAE (m/s)	RMSE (m/s)	SSIM	MAE, [−1, 1] scale	DSUNet lower by
InversionNet, standard	24.4 M	252.7 ± 2.1	384.4	0.564 ± 0.006	0.169	15.1%
InversionNet, 120-epoch	24.4 M	249.8 ± 1.6	390.3	0.570 ± 0.002	0.167	14.1%
InversionNet, input-matched	24.4 M	243.6 ± 2.9	376.8	0.571 ± 0.003	0.162	11.9%
InversionNet, matched recipe	24.4 M	235.1 ± 1.7	358.1	0.585 ± 0.002	0.157	8.7%
DSUNet (ours)	**2.08 M**	**214.6 ± 0.3**	**348.9**	**0.596 ± 0.001**	**0.143**	—

DSUNet's MAE on odd columns (competition rule) was 214.8 m/s; $R^2 \approx 0.82$; PSNR = 21.1 dB.

The difference is consistent at the level of individual samples (Table 5). For every seed and protocol, DSUNet had the lower error on most test samples: 70–74% against the matched recipe and 79–83% against the standard protocol. All 12 comparisons remained significant after Holm correction (largest adjusted $p = 4.2 \times 10^{-191}$).

Table 5 Paired comparison per seed: fraction of the 5,000 test samples on which DSUNet has the lower MAE (Holm-corrected Wilcoxon p in parentheses)

Comparator	Seed 42	Seed 13	Seed 73
InversionNet, standard	83.2% ($< 10^{-300}$)	78.9% ($< 10^{-300}$)	79.4% ($< 10^{-300}$)
InversionNet, 120-epoch	79.2% ($< 10^{-300}$)	75.0% ($< 10^{-300}$)	76.6% ($< 10^{-300}$)
InversionNet, input-matched	73.6% (2.8×10^{-284})	72.6% (4.8×10^{-258})	76.0% ($< 10^{-300}$)
InversionNet, matched recipe	70.4% (4.2×10^{-191})	74.0% (6.5×10^{-252})	71.0% (1.5×10^{-206})

p-values that underflow double precision are reported as $< 10^{-300}$.

5.2 Training behaviour

Fig. 2 shows the validation error during training. InversionNet under the 120-epoch protocol stopped improving at about 240 m/s after roughly 30 epochs, so the longer schedule did not close the gap; the

matched recipe reached a lower plateau, and DSUNet continued to improve until the end of its cosine schedule. All models showed large validation fluctuations while the learning rate was high

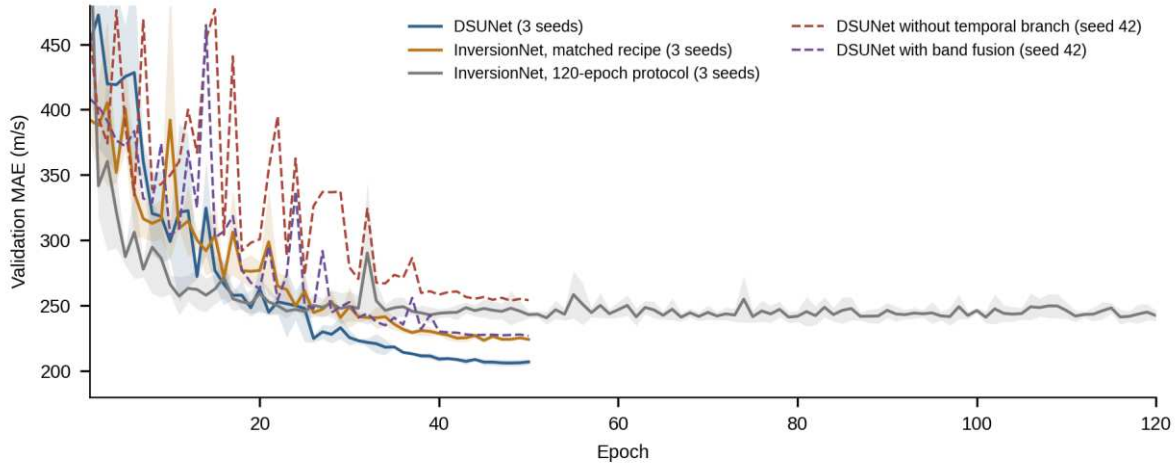


Fig. 2 Validation MAE during training. Solid lines: mean over three seeds (shaded: ± one standard deviation). Dashed lines: single-seed ablations. Early-stopped runs end before epoch 50

5.3 Accuracy by family and examples

Accuracy varied strongly between families (Table 6). Flat layered models (FlatVel_A, 83 m/s) and simple faults (FlatFault_A, 135 m/s) were reconstructed best; the B versions of the curved families were hardest (CurveFault_B 332 m/s, CurveVel_B 413 m/s), mirroring the difficulty ordering reported for OpenFWI (Deng et al. 2022). Fig. 3 shows the test sample of median error for six families. DSUNet recovers the overall velocity trend and the position of major interfaces, but its predictions are smoother than the targets: sharp interfaces, thin layers and fault offsets are blurred, and most of the error concentrates along them.

Table 6 DSUNet test results per family (mean ± standard deviation over three seeds; 500 test samples per family)

Family	MAE (m/s)	RMSE (m/s)	SSIM	R^2
FlatVel_A	83.4 ± 0.8	133.9	0.794	0.971
FlatFault_A	134.9 ± 1.9	217.9	0.769	0.934
Style_B	153.4 ± 0.3	195.8	0.532	0.900
CurveFault_A	175.7 ± 1.2	274.3	0.673	0.897
CurveVel_A	179.0 ± 1.1	282.9	0.623	0.874
Style_A	211.7 ± 0.5	291.2	0.599	0.798
FlatVel_B	212.7 ± 4.1	374.6	0.627	0.814
FlatFault_B	250.1 ± 1.3	348.4	0.540	0.834
CurveFault_B	332.3 ± 0.7	463.0	0.416	0.728
CurveVel_B	413.0 ± 2.4	630.4	0.388	0.479
All families	**214.6 ± 0.3**	**348.9**	**0.596**	**0.817**

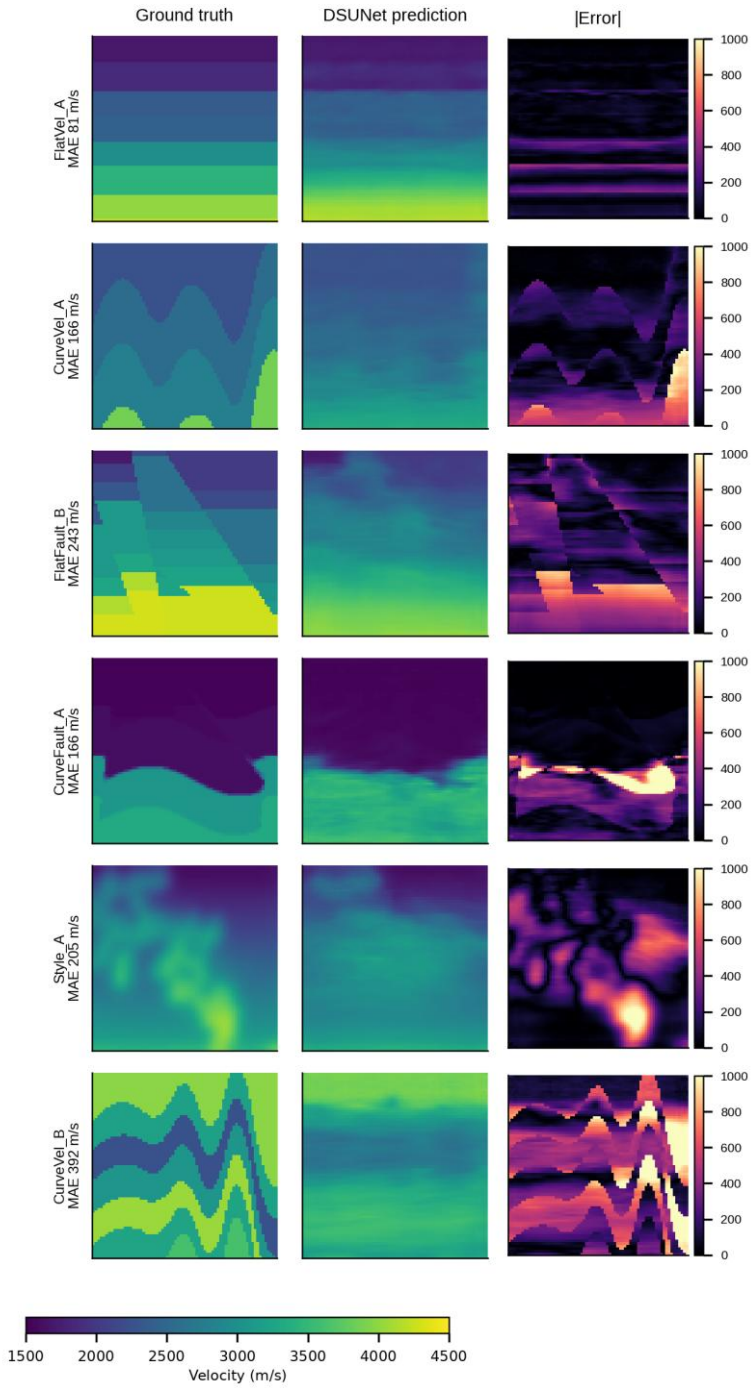


Fig. 3 Test samples of median error (seed 42) for six families: ground truth, DSUNet prediction and absolute error. Rows are labelled with the family and the sample's MAE. Velocity maps share the colour scale at the bottom

5.4 Ablations

Table 7 isolates the contribution of each component (seed 42). Removing the temporal branch increased the MAE by 20.5% (258.8 m/s), which is worse than all four InversionNet protocols; the full model was better on 83.3% of test samples. Replacing time-aligned fusion with band fusion, at an identical parameter count, increased the MAE by 8.4% (232.9 m/s; better on 70.5% of samples). Removing both auxiliary penalties together had no measurable effect (214.4 versus 214.8 m/s; $p = 0.35$). Under the standard early-stopping rule, the variant without the temporal branch stopped at epoch 14 with 330.7 m/s; the full-schedule run is reported to reduce the effect of premature stopping.

Table 7 Ablations (seed 42; full model 214.8 m/s). Paired tests against the full model with the same seed

Variant	Parameters	MAE (m/s)	Change	SSIM	Full model better on	Holm-corrected p
Full DSUNet	2.08 M	214.8	—	0.597	—	—
Without auxiliary penalties	2.08 M	214.4	−0.2%	0.595	49.6%	0.35
Band fusion instead of time-aligned	2.08 M	232.9	+8.4%	0.567	70.5%	7.2×10^{-205}
Without temporal branch	1.27 M	258.8	+20.5%	0.527	83.3%	$< 10^{-300}$

5.5 Robustness to noise

Noise degraded all models (Table 8, Fig. 4), but DSUNet least. At 5 dB, its MAE rose to 468 m/s, compared with 951 m/s for InversionNet under the standard protocol. Part of this advantage comes from the input normalization: InversionNet trained on per-gather standardized inputs degraded much less than with its default preprocessing (645 m/s at 5 dB). Against this input-matched control, DSUNet's error was still 35%, 25% and 27% lower at 20, 10 and 5 dB, which is consistent with a benefit beyond input normalization, although differences in the training objective and schedule remain. InversionNet trained with the matched recipe was unstable under noise, with seed-to-seed standard deviations of 140–200 m/s.

Table 8 Test MAE (m/s) under additive Gaussian noise (mean ± standard deviation over three seeds)

Model	Clean	20 dB	10 dB	5 dB
InversionNet, standard	252.7 ± 2.1	518.1 ± 60.0	692.5 ± 95.4	951.2 ± 47.2
InversionNet, 120-epoch	249.8 ± 1.6	528.6 ± 56.4	659.1 ± 26.5	882.6 ± 73.8
InversionNet, matched recipe	235.1 ± 1.7	584.2 ± 183.7	591.5 ± 199.7	587.8 ± 142.2
InversionNet, input-matched	243.6 ± 2.9	424.4 ± 59.4	541.3 ± 49.8	645.0 ± 65.5
DSUNet	**214.6 ± 0.3**	**275.4 ± 44.6**	**407.8 ± 33.7**	**468.1 ± 40.1**

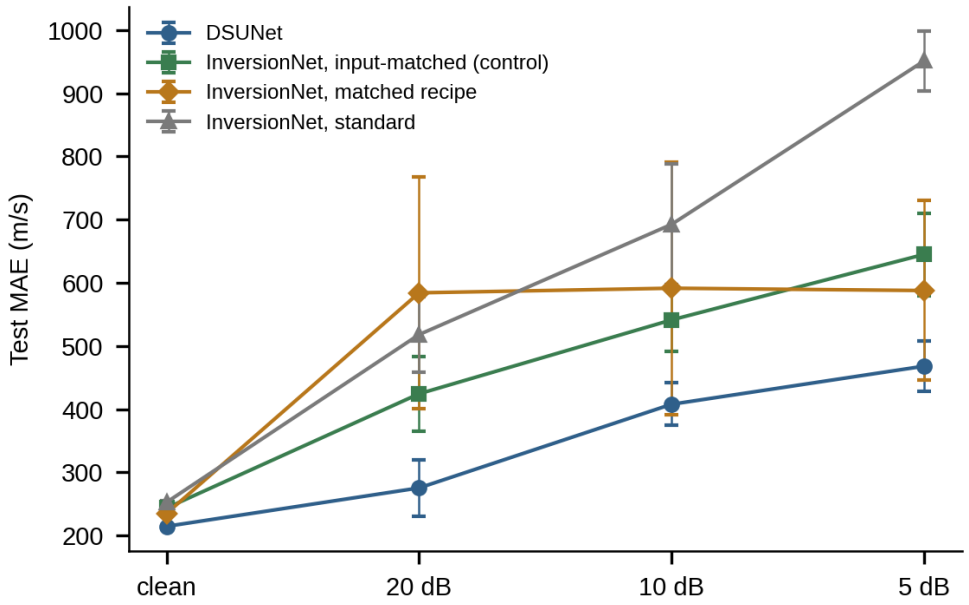


Fig. 4 Test MAE under additive Gaussian noise (mean ± standard deviation over three seeds)

5.6 Ensemble uncertainty

Averaging the three DSUNet seeds reduced the MAE from 214.6 to 204.4 m/s (4.8%) and raised the SSIM from 0.596 to 0.620. The pixel-wise spread of the members was informative. Across all test samples, the ensemble MAE correlated with the mean predictive standard deviation (Pearson r = 0.80; Fig. 5a), but part of this reflects differences between families: within individual families the correlation was 0.47–0.76

(median 0.57), and 0.68 after removing family means. The spread alone identified the 10% of samples with the largest error with an AUROC of 0.93, and within a sample it was largest along the interfaces where the error was largest (Fig. 5b, c; pixel-level r = 0.45). This uncertainty estimate comes at no extra training cost when several seeds are trained anyway, although three members are too few for calibrated intervals.

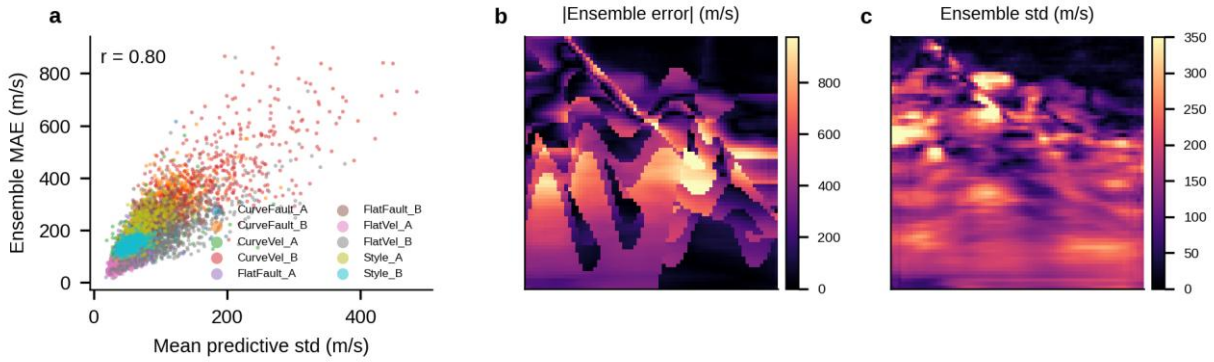


Fig. 5 Ensemble uncertainty. a Per-sample ensemble MAE against mean predictive standard deviation for the 5,000 test samples, coloured by family. b, c Absolute error and standard deviation for the CurveFault_B test sample of median error

5.7 Data check of the monotonic depth prior

The directional penalty assumes that velocity does not decrease with depth. Table 9 shows how often the training targets violate this assumption. Because the Vel and Fault maps are piecewise constant, only 6–9% of vertically adjacent pairs are depth transitions (changes larger than 1 m/s), whereas almost every pair changes in the smooth Style maps; the last column therefore reports decreases as a share of transitions. The assumption holds exactly only for FlatVel_A and CurveVel_A. In the four Fault families, 10–20% of the transitions are decreases, in the Style families 18–23%, and in FlatVel_B and CurveVel_B about half. A prior that is wrong for a large part of the data cannot be expected to help, and the ablation confirms that it did not.

Table 9 Velocity transitions between vertically adjacent cells in 200 training maps per family (changes larger than 1 m/s)

Family	Decreasing (% of all pairs)	Increasing (% of all pairs)	Decreases (% of transitions)
CurveFault_A	1.5	7.5	16%
CurveFault_B	1.0	7.3	12%
CurveVel_A	0.0	6.3	0%
CurveVel_B	3.3	3.0	52%
FlatFault_A	1.6	6.4	20%
FlatFault_B	0.8	7.5	10%
FlatVel_A	0.0	6.5	0%
FlatVel_B	3.1	3.2	49%
Style_A	23.0	75.9	23%
Style_B	18.1	81.3	18%

5.8 Computational cost

DSUNet trades parameters for computation (Table 10): it is parameter-efficient but not compute-efficient. It has 11.7 times fewer parameters and needs half the peak memory of InversionNet, but it performs at least 12.5 times more floating-point operations and is 4.1 times slower at inference, because its encoder and decoder operate at full time resolution (up to 1000 × 70) and the LSTM processes long sequences.

The measured latency is about 24 ms per sample for a trained network; comparison with iterative FWI runtimes reported elsewhere is not a controlled end-to-end benchmark.

Table 10 Cost per sample (batch size 1, one NVIDIA T4 GPU)

Model	Parameters	GFLOPs	Latency (ms)	Peak memory (MB)
InversionNet	24,409,123	4.19	5.7	485
DSUNet	2,078,868	≥ 52.3	23.8	244

DSUNet's GFLOPs are a lower bound because LSTM kernels may not be counted.

6 Discussion

6.1 Why time alignment helps

The temporal branch is an important contributor to DSUNet's performance: without it, DSUNet falls behind the best-trained InversionNet. The fusion strategy also matters. Band fusion compresses each source's sequence into one vector and places the five vectors on fixed bands of rows, so the position of a feature in the image no longer corresponds to the arrival time it came from. Time-aligned fusion keeps that correspondence, and because arrival time relates to depth through the velocity structure, it gives the decoder information that is both sequence-aware and spatially consistent. We offer this as an interpretation; the ablation establishes the effect, not the mechanism.

6.2 Training recipes confound baseline comparisons

InversionNet's error fell from 252.7 to 235.1 m/s (7.0%) when it was trained with DSUNet's recipe, whereas a longer schedule with its default recipe barely helped (249.8 m/s). Without the matched protocol, DSUNet's advantage would have been reported as 15.1% instead of 8.7%. In the limited-data regime studied here, the batch size and learning-rate schedule materially affect the observed difference, and comparisons should match them or report several protocols.

6.3 Physics-inspired penalties

Removing the two auxiliary penalties together did not change accuracy; we did not test them individually. The cumulative-slowness term carries little weight relative to the reconstruction loss, and the directional penalty encodes an assumption that the data violate at up to half of the depth transitions (Table 9). We keep both terms in the reported configuration only because removing them changed nothing, and we do not claim that DSUNet is physics-guided. More generally, the prevalence of such priors in the data can be checked before they are imposed; a family-adaptive or learned prior is a possible extension.

6.4 Geophysical interpretation and field-validation requirements

Arrival time carries information about propagation paths, but its relationship to depth depends on the velocity model, source–receiver geometry and wave mode. Time-aligned fusion is therefore a learned representation, not a physical time-to-depth conversion. The blurred interfaces and fault offsets in Fig. 3 are important limitations for structural interpretation. A future validation should include a realistic synthetic model with unseen structures and measured field gathers with documented acquisition geometry, source signature and preprocessing. Field velocity estimates should be assessed against available well logs or independent velocity models, and by forward-modelled waveform or travel-time residuals where feasible. None of these field experiments is reported here.

6.5 Limitations

- Data scale and comparability. The study uses 10,000 samples from the competition subset of OpenFWI. Published results obtained on full splits cannot be directly compared with these results. Only one independent architecture (InversionNet) was retrained; VelocityGAN, UPFWI, modern attention-based inversion models and iterative FWI remain necessary comparators for broader claims.

- **One test fold.** Each family contributes one test file. Seed variability is small, but variability across splits was not measured; the second fold (files swapped) is the obvious next experiment.
- Single-seed ablations and incomplete controls. Ablations were conducted with one seed; coordinate attention and the two penalties were not ablated individually. A fully matched architecture comparison and repeated ablations are needed to isolate individual design effects.
- Synthetic data only. No measured field data or realistic synthetic benchmark models such as Marmousi were tested. Additive white Gaussian noise does not represent all field effects, including coherent noise, unknown source signatures, acquisition variations, attenuation and modelling error. The current results do not establish field-data applicability.
- **Smooth predictions.** Sharp interfaces and fault offsets are blurred (Fig. 3); losses or decoders that preserve discontinuities may help.
- **Cost.** DSUNet needs at least 12.5 times more operations and is 4.1 times slower than InversionNet at inference.
- **Uncertainty.** Three ensemble members indicate where errors are large but do not give calibrated intervals.

7 Conclusion

DSUNet combines BiLSTM and self-attention features with a convolutional encoder–decoder through time-aligned bottleneck fusion. In the limited-data synthetic setting studied here, it achieved a test MAE of 214.6 ± 0.3 m/s, 8.7% lower than the strongest of four official InversionNet training protocols on the same split. Single-seed ablations support the temporal branch and time-aligned fusion, while the noise experiments and ensemble provide evidence of robustness to additive Gaussian noise and error-ranking ability. Joint removal of the physics-inspired penalties did not measurably affect MAE. Although DSUNet uses fewer parameters, it incurs substantially more operations and longer inference than InversionNet. These conclusions are restricted to the tested synthetic families and one file-disjoint test fold. A fully matched comparison with additional inversion methods, evaluation on unseen geological models and acquisition geometries, and validation on measured field data are needed before claims of broader geophysical applicability can be made.

Statements and Declarations

Data Availability

The training samples of the Kaggle Yale/UNC-CH – Geophysical Waveform Inversion competition (Kaggle 2025) are available to registered participants under the competition's data-use rules; they are derived from OpenFWI (Deng et al. 2022).

Code Availability

The DSUNet training and analysis code, including the experimental notebook, is available from the corresponding author upon reasonable request. The InversionNet implementation used in this study is publicly available from the official OpenFWI repository (https://github.com/lanl/OpenFWI).

Funding

No funding was received for this research.

Competing Interests

The authors declare no competing interests.

Author Contributions

Md Mahabube Alahi Atik: Methodology, software, investigation, data curation, formal analysis, visualization, and writing – original draft. Abdul Ahad: Supervision, conceptualization, methodological guidance, research oversight, critical review, and writing – review and editing. Abdul Ahad is the corresponding author.

Ethics Approval and Consent to Participate

Not applicable. This study uses synthetic seismic datasets and does not involve human participants or animals.

Consent for Publication

Not applicable; no individual participant data are included.

Use of AI-assisted Tools

Large language model tools were used to assist with code review, the engineering of the experiment pipeline, analysis and figure scripts, and language editing. Figure 1 is a text-based diagram; Figs. 2–5 plot recorded experimental outputs. The authors designed and approved the study, verified all content and take full responsibility for the manuscript.

References

Akter KT, Ahad A, Tasnia R, Fahim A (2024) Predicting antibiotic resistance in gonorrhea: a comparative analysis of machine learning, deep learning, and explainable AI. In: 2024 27th International Conference on Computer and Information Technology (ICCIT), pp 1696–1701. https://doi.org/10.1109/ICCIT64611.2024.11022432

Araya-Polo M, Jennings J, Adler A, Dahlke T (2018) Deep-learning tomography. Lead Edge 37(1):58–66. https://doi.org/10.1190/tle37010058.1

Bunks C, Saleck FM, Zaleski S, Chavent G (1995) Multiscale seismic waveform inversion. Geophysics 60(5):1457–1473. https://doi.org/10.1190/1.1443880

Chen G, Liu Y, Zhang H, Zhang M (2026) Seismic full waveform inversion via a physics-guided Fourier representation neural network. arXiv:2606.30126. https://doi.org/10.48550/arXiv.2606.30126

Deng C, Feng S, Wang H, Zhang X, Jin P, Feng Y, Zeng Q, Chen Y, Lin Y (2022) OpenFWI: large-scale multi-structural benchmark datasets for full waveform inversion. In: Advances in Neural Information Processing Systems 35 (Datasets and Benchmarks Track), pp 6007–6020. arXiv:2111.02926

Deng C, Lin Y (2021) On the robustness and generalization of deep learning driven full waveform inversion. arXiv:2111.14220

Fahmid A, Ahad A, Rahman S, Ferdausi S, Aashish KC, Pranto SMSH, Akter KT, Islam DA (2025) Multi-label thoracic disease detection and weakly supervised localization in chest X-rays using DenseNet121 and probabilistic CAM pooling. In: 2025 IEEE 4th International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON), pp 450–455. https://doi.org/10.1109/RAAICON69033.2025.11502273

Geng B, Wang N, Shi Y (2026) Robust physics-informed reparameterized full-waveform inversion via CNN-enhanced vision transformer. J Geophys Eng 23(5):1513–1527. https://doi.org/10.1093/jge/gxag061

Goyal P, Dollár P, Girshick R, Noordhuis P, Wesolowski L, Kyrola A, Tulloch A, Jia Y, He K (2017) Accurate, large minibatch SGD: training ImageNet in 1 hour. arXiv:1706.02677

Harsuko R, Cheng S, Alkhalifah T (2025) Propagating the prior from shallow to deep with a pre-trained velocity-model generative Transformer network. J Geophys Res Mach Learn Comput 2:e2024JH000408. https://doi.org/10.1029/2024JH000408

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 770–778. https://doi.org/10.1109/CVPR.2016.90

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Holm S (1979) A simple sequentially rejective multiple test procedure. Scand J Stat 6(2):65–70

Hou Q, Zhou D, Feng J (2021) Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 13713–13722

Islam DA, Ahad A, Islam MM, Atik MMA, Fahmid A (2025) Anomaly detection for Sunburst attack identification using hybrid AI: a stacking ensemble and attention based LSTM approach. In: 2025 International Conference on Computer Systems and Technologies (CompSysTech). https://doi.org/10.1109/CompSysTech65493.2025.11136883

Jia A, Sun J, Du B, Lin Y (2025) Seismic full waveform inversion with uncertainty analysis using unsupervised variational deep learning. IEEE Trans Geosci Remote Sens 63:1–16. https://doi.org/10.1109/TGRS.2025.3564647

Jin P, Feng Y, Feng S, Wang H, Chen Y, Consolvo B, Liu Z, Lin Y (2024) An empirical study of large-scale data-driven full waveform inversion. Sci Rep 14:20034. https://doi.org/10.1038/s41598-024-68573-7

Jin P, Zhang X, Chen Y, Huang SX, Liu Z, Lin Y (2022) Unsupervised learning of full-waveform inversion: connecting CNN and partial differential equation in a loop. In: International Conference on Learning Representations (ICLR)

Kaggle (2025) Yale/UNC-CH – Geophysical Waveform Inversion [competition]. https://www.kaggle.com/competitions/waveform-inversion. Accessed 30 September 2026

Lakshminarayanan B, Pritzel A, Blundell C (2017) Simple and scalable predictive uncertainty estimation using deep ensembles. In: Advances in Neural Information Processing Systems 30, pp 6402–6413

Lee C-Y, Xie S, Gallagher P, Zhang Z, Tu Z (2015) Deeply-supervised nets. In: Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS), PMLR 38, pp 562–570

Lin Y, Theiler J, Wohlberg B (2023) Physics-guided data-driven seismic inversion: recent progress and future opportunities in full-waveform inversion. IEEE Signal Process Mag 40(1):115–133

Liu W, Xie Y, Du Z, Li H, Xiong P (2026) Seismic velocity modeling based on attention-enhanced UNet. J Geophys Prospect 2(1):1–11. https://doi.org/10.1016/j.jgp.2026.01.002

Los Alamos National Laboratory (2022) OpenFWI code repository. https://github.com/lanl/OpenFWI. Accessed 30 September 2026

Loshchilov I, Hutter F (2017) SGDR: stochastic gradient descent with warm restarts. In: International Conference on Learning Representations (ICLR). arXiv:1608.03983

Loshchilov I, Hutter F (2019) Decoupled weight decay regularization. In: International Conference on Learning Representations (ICLR). arXiv:1711.05101

Lu C, Liu J, Qu L, Gao J, Cai H, Liang J (2025) First-arrival constrained physics-informed recurrent neural networks for initial model-insensitive full waveform inversion in vertical seismic profiling. Appl Sci 15(10):5757. https://doi.org/10.3390/app15105757

Micikevicius P, Narang S, Alben J, Diamos G, Elsen E, Garcia D, Ginsburg B, Houston M, Kuchaiev O, Venkatesh G, Wu H (2018) Mixed precision training. In: International Conference on Learning Representations (ICLR). arXiv:1710.03740

Pascanu R, Mikolov T, Bengio Y (2013) On the difficulty of training recurrent neural networks. In: Proceedings of the 30th International Conference on Machine Learning (ICML), PMLR 28(3), pp 1310–1318

Prechelt L (1998) Early stopping—but when? In: Orr GB, Müller K-R (eds) Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science, vol 1524. Springer, Berlin, pp 55–69. https://doi.org/10.1007/3-540-49430-8_3

Rasht-Behesht M, Huber C, Shukla K, Karniadakis GE (2022) Physics-informed neural networks (PINNs) for wave propagation and full waveform inversions. J Geophys Res Solid Earth 127(5):e2021JB023120. https://doi.org/10.1029/2021JB023120

Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer, Cham, pp 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

Tarantola A (1984) Inversion of seismic reflection data in the acoustic approximation. Geophysics 49(8):1259–1266. https://doi.org/10.1190/1.1441754

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in Neural Information Processing Systems 30, pp 5998–6008

Virieux J, Operto S (2009) An overview of full-waveform inversion in exploration geophysics. Geophysics 74(6):WCC1–WCC26. https://doi.org/10.1190/1.3238367

Wang C, Huang X, Mousavi SM, Li Y (2026) An unsupervised inversion framework in the frequency domain using a Wasserstein generative adversarial network. Geophys J Int 246(2):ggag192. https://doi.org/10.1093/gji/ggag192

Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612. https://doi.org/10.1109/TIP.2003.819861

Wei Y, Peng K, Feng C, Chen T, Zhang G, Sun J, Cui L (2026) Frequency-aware seismic velocity inversion via long-range dependency learning. Open Geosci 18(1):20250970. https://doi.org/10.1515/geo-2025-0970

Wilcoxon F (1945) Individual comparisons by ranking methods. Biom Bull 1(6):80–83. https://doi.org/10.2307/3001968

Wu Y, Lin Y (2020) InversionNet: an efficient and accurate data-driven full waveform inversion. IEEE Trans Comput Imaging 6:419–433. https://doi.org/10.1109/TCI.2019.2956866

Yang F, Ma J (2019) Deep-learning inversion: a next-generation seismic velocity model building method. Geophysics 84(4):R583–R599. https://doi.org/10.1190/geo2018-0249.1

Yang F, Ma J (2023) FWIGAN: full-waveform inversion via a physics-informed generative adversarial network. J Geophys Res Solid Earth 128(4):e2022JB025493. https://doi.org/10.1029/2022JB025493

Zhang Z, Lin Y (2020) Data-driven seismic waveform inversion: a study on the robustness and generalization. IEEE Trans Geosci Remote Sens 58(10):6900–6913

Zhang Z, Wu Y, Zhou Z, Lin Y (2019) VelocityGAN: subsurface velocity image estimation using conditional adversarial networks. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, Waikoloa

Zhu C, Wang Z, Li F, Zhang H, Gao J (2025) Cyclic learning rate U-shaped ResNet embedded with dual attentions for velocity model building. IEEE J Sel Top Appl Earth Obs Remote Sens 18:1054–1069. https://doi.org/10.1109/JSTARS.2024.3501736