

## APPENDICES

## A THEORETICAL JUSTIFICATIONS FOR LATENT SPACE TRANSLATION ASSUMPTION

In this Section, we provide two Lemmas to theoretically support our Latent Space Translation assumption for the Generalized-Forward Inverse (GFI) Framework.

**Lemma A.1** (Forward Latent Space Translation Assumption). *Let  $f : \mathcal{V} \rightarrow \mathcal{P}$  be an arbitrary forward operator mapping velocity maps  $\mathcal{V}$  to seismic waveforms  $\mathcal{P}$ . Let  $E_v : \mathcal{V} \rightarrow \tilde{\mathcal{V}}$  and  $D_v : \tilde{\mathcal{V}} \rightarrow \mathcal{V}$  denote the encoder and decoder for the velocity space  $\mathcal{V}$ , respectively. Similarly, let  $E_p : \mathcal{P} \rightarrow \tilde{\mathcal{P}}$  and  $D_p : \tilde{\mathcal{P}} \rightarrow \mathcal{P}$  denote the encoder and decoder for the seismic waveform space  $\mathcal{P}$ . Here,  $\tilde{v} \in \tilde{\mathcal{V}}$  and  $\tilde{p} \in \tilde{\mathcal{P}}$  represent the latent space encodings. If we assume that the auto-encoder for the velocity are optimal, i.e.,  $D_v \circ E_v(v) = \hat{v} \approx v$ , then there exists a functional mapping in the latent space  $L_{\tilde{v} \rightarrow \tilde{p}} : \tilde{\mathcal{V}} \rightarrow \tilde{\mathcal{P}}$ .*

*Proof.* Given the forward operator  $f : \mathcal{V} \rightarrow \mathcal{P}$ , by definition, for any  $v \in \mathcal{V}$ , there exists  $p \in \mathcal{P}$  such that  $p = f(v)$ .

Let the latent space representations  $\tilde{v} \in \tilde{\mathcal{V}}$  and  $\tilde{p} \in \tilde{\mathcal{P}}$  be defined by the auto-encoders as follows:

$$\tilde{v} = E_v(v), \quad \hat{v} = D_v(\tilde{v}) \quad (10)$$

$$\tilde{p} = E_p(p), \quad \hat{p} = D_p(\tilde{p}) \quad (11)$$

To construct the latent space mapping  $\tilde{p} = L_{\tilde{v} \rightarrow \tilde{p}}(\tilde{v})$ , consider the sequence of compositions involving the encoders, decoders, and the forward operator  $f$ :

$$\begin{aligned} \tilde{p} &= E_p(p) \\ &= E_p(f(v)) \quad (\text{since } p = f(v)) \\ &= E_p(f(\hat{v})) \quad (\text{assuming reconstruction: } \hat{v} \approx v) \\ &= E_p(f(D_v(\tilde{v}))) \quad (\text{since } \hat{v} = D_v(\tilde{v})) \end{aligned} \quad (12)$$

Thus, by definition of the composition of functions, the latent space mapping can be expressed as:

$$L_{\tilde{v} \rightarrow \tilde{p}} = E_p \circ f \circ D_v \quad (13)$$

□

**Lemma A.2** (Inverse Latent Space Translation Assumption). *Let  $f^{-1} : \mathcal{P} \rightarrow \mathcal{V}$  be an arbitrary inverse operator mapping seismic waveforms  $\mathcal{P}$  to velocity maps  $\mathcal{V}$  that is unique. Let  $E_p : \mathcal{P} \rightarrow \tilde{\mathcal{P}}$  and  $D_p : \tilde{\mathcal{P}} \rightarrow \mathcal{P}$  denote the encoder and decoder for the seismic waveform space  $\mathcal{P}$ , respectively. Similarly, let  $E_v : \mathcal{V} \rightarrow \tilde{\mathcal{V}}$  and  $D_v : \tilde{\mathcal{V}} \rightarrow \mathcal{V}$  denote the encoder and decoder for the velocity space  $\mathcal{V}$ . Here,  $\tilde{p} \in \tilde{\mathcal{P}}$  and  $\tilde{v} \in \tilde{\mathcal{V}}$  represent the latent space encodings. If we assume that the auto-encoder for the seismic waveform space is optimal, i.e.,  $D_p \circ E_p(p) = \hat{p} \approx p$ , then there exists a functional mapping in the latent space  $L_{\tilde{p} \rightarrow \tilde{v}} : \tilde{\mathcal{P}} \rightarrow \tilde{\mathcal{V}}$ .*

*Proof.* Given the inverse operator  $f^{-1} : \mathcal{P} \rightarrow \mathcal{V}$ , by definition, for any  $p \in \mathcal{P}$ , there exists  $v \in \mathcal{V}$  such that  $v = f^{-1}(p)$ .

Let the latent space representations  $\tilde{p} \in \tilde{\mathcal{P}}$  and  $\tilde{v} \in \tilde{\mathcal{V}}$  be defined by the auto-encoders as follows:

$$\tilde{p} = E_p(p), \quad \hat{p} = D_p(\tilde{p}) \quad (14)$$

$$\tilde{v} = E_v(v), \quad \hat{v} = D_v(\tilde{v}) \quad (15)$$

To construct the latent space mapping  $\tilde{v} = L_{\tilde{p} \rightarrow \tilde{v}}(\tilde{p})$ , consider the sequence of compositions involving the encoders, decoders, and the inverse operator  $f^{-1}$ :

$$\begin{aligned}
 \tilde{v} &= E_v(v) \\
 &= E_v(f^{-1}(p)) \quad (\text{since } v = f^{-1}(p)) \\
 &= E_v(f^{-1}(\hat{p})) \quad (\text{assuming reconstruction: } \hat{p} \approx p) \\
 &= E_v(f^{-1}(D_p(\tilde{p}))) \quad (\text{since } \hat{p} = D_p(\tilde{p}))
 \end{aligned} \tag{16}$$

Thus, by definition of the composition of functions, the latent space mapping can be expressed as:

$$L_{\tilde{p} \rightarrow \tilde{v}} = E_v \circ f^{-1} \circ D_p \tag{17}$$

□

Dataset	Examples	Velocity shape	Waveform shape
FlatVel-A	30,000	(1, 70, 70)	(5, 1000, 70)
FlatVel-B	30,000	(1, 70, 70)	(5, 1000, 70)
CurveVel-A	30,000	(1, 70, 70)	(5, 1000, 70)
CurveVel-B	30,000	(1, 70, 70)	(5, 1000, 70)
FlatFault-A	60,000	(1, 70, 70)	(5, 1000, 70)
FlatFault-B	60,000	(1, 70, 70)	(5, 1000, 70)
CurveFault-A	60,000	(1, 70, 70)	(5, 1000, 70)
CurveFault-B	60,000	(1, 70, 70)	(5, 1000, 70)
Style-A	67,000	(1, 70, 70)	(5, 1000, 70)
Style-B	67,000	(1, 70, 70)	(5, 1000, 70)

Table 2: Statistics on the number of samples, the size of the velocity and waveforms for each dataset in OpenFWI [Deng et al. \(2022\)](#).

## B DATASET DESCRIPTION

The OpenFWI comprises multi-structural benchmark datasets of significant size that can be used for solving full waveform inversion (FWI) using machine learning techniques ([Deng et al., 2022](#)). In particular, the repository contains 3 major groups of data: (1) Vel Family, (2) Fault Family, and (3) Style Family. These groups represent simple to complex sub-surface geological settings with seismic velocity and waveforms information. The Vel family is the simplest geological patterns including four datasets - (1) FlatVel-A (FVA), (2) FlatVel-B (FVB), (3) CurveVel-A (CVA), and (4) CurveVel-B (CVB). The difference between FlatVel and CurveVel is that the former represents low-energy geological environments where the rock layers are deposited horizontally and the latter consists of curved layers which are formed due to structural deformation of flat layers. The Fault family also has four datasets - (1) FlatFault-A (FFA), (2) FlatFault-B (FFB), (3) CurveFault-A (CFA), and (4) CurveFault-B (CFB). Unlike Vel datasets, the Fault family contains fault-like deformations, which is fracturing of rocks under certain pressure conditions. Due to the presence of faults, the Fault family becomes more complicated and challenging to model. The Style family has two datasets - (1) Style-A (STA), and (2) Style-B (STB). This dataset is generated using the style transfer method where the COCO dataset ([Lin et al., 2014](#)) is set as the content images and the Marmousi dataset is set as the style image. This is the most complex OpenFWI dataset as it represents highly complex geological settings where the velocity is changing rapidly and abruptly. In summary, the details about the Vel and Fault datasets are described in [table 2](#).

The waveform data is represented as (# source  $\times$  recording time  $\times$  receiver length) whereas the velocity follow (1  $\times$  depth  $\times$  receiver length) shape. In seismic surveys, the wave arrival time is recorded at the surface and thus, the waveform data for a single source is represented as a function of time and receiver length. In total, there are 5 seismic sources uniformly spaced along the surface, and

wavefields are recorded by 70 receivers (uniformly spaced) along the surface for 1000 milliseconds. Therefore, the seismic wavefields are of the shape  $(5 \times 1000 \times 70)$ . On the other hand, velocity maps are represented as functions of spatial dimensions, depth and horizontal coverage, and thus have the shape  $(1 \times 70 \times 70)$ .

## C PRIOR WORKS IN DL4SI AS SPECIAL CASES OF GFI

Figure 9 provides additional schematic illustrations of prior works in DL4SI such as InversionNet, WaveformNet, and AutoLinear as special cases of our proposed GFI framework.

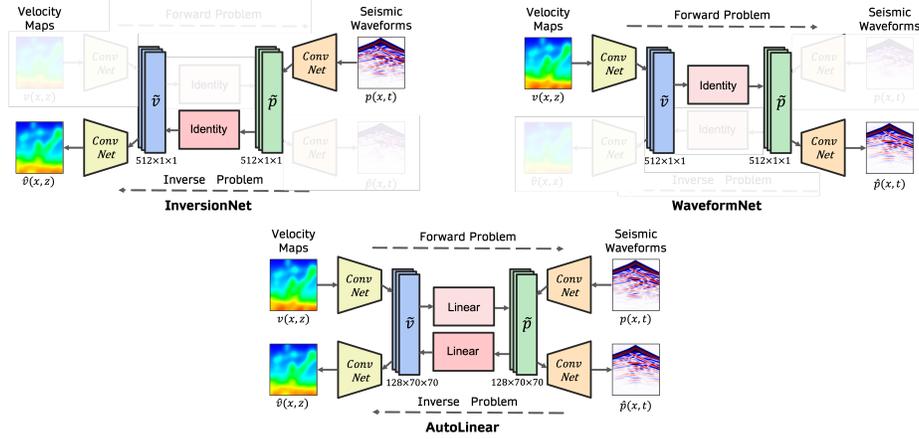


Figure 9: Prior Works expressed as GFI

## D ADDITIONAL EXPERIMENTAL DETAILS

### D.1 CYCLE LOSS FOR INVERTIBLE X-NET

Given the velocity maps  $v$  and the waveforms  $p$ , the predictions for the waveform and velocity can be obtained using the Invertible X-Net as follows:

$$\hat{p} = f_{v \rightarrow p}(v) = D_p \circ L_{\tilde{v} \rightarrow \tilde{p}} \circ E_v(v) \quad (18)$$

$$\hat{v} = f_{p \rightarrow v}(p) = D_v \circ L_{\tilde{p} \rightarrow \tilde{v}} \circ E_p(p) \quad (19)$$

Now, the Invertible X-Net architecture can be further applied on  $\hat{p}$  and  $\hat{v}$  to create the following transformations:

$$\hat{\hat{p}} = f_{v \rightarrow p}(\hat{v}) = D_p \circ L_{\tilde{v} \rightarrow \tilde{p}} \circ E_v(\hat{v}) \quad (20)$$

$$\hat{\hat{v}} = f_{p \rightarrow v}(\hat{p}) = D_v \circ L_{\tilde{p} \rightarrow \tilde{v}} \circ E_p(\hat{p}) \quad (21)$$

The cycle-loss for Invertible X-Net can be mathematically defined as follows:

$$\mathcal{L}_{\text{cycle}} = \mathcal{L}(p, \hat{\hat{p}}) + \mathcal{L}(v, \hat{\hat{v}}) \quad (22)$$

where  $\mathcal{L}$  is the loss function that can be MSE, MAE or Elastic Loss. Note that the formulation of Cycle-Loss does not rely on paired examples, and can be applied on un-paired data as well. The combined loss function including cycle-loss can be given as:

$$\mathcal{L}_{\text{X-Net (Cycle)}} = \mathcal{L}_{\text{forward}} + \mathcal{L}_{\text{inverse}} + \mathcal{L}_{\text{cycle}} \quad (23)$$

For training Invertible X-Net (Cycle) model, we use loss function shown in Equation 23.

## D.2 ADDITIONAL TRAINING DETAILS

For training, we normalize the velocity using min-max normalization and seismic waveform using standard normalization to rescale the data to mean 0 and standard deviation as 1. Table 5 shows other hyperparameter details for training Latent U-Net and Invertible X-Net models on OpenFWI datasets.

Since the baseline models have different normalization schemes than our models, we compare model predictions during evaluation by unnormalizing the predictions to original domains. For example, AutoLinear uses min-max normalization for velocity and a combination of log-normalization  $x_{\text{norm}} = (\log_e(1 + |x|)) * \text{sign}(x)$  followed by min-max in the log-normalized domain for waveform. We unnormalized both the velocity and waveform predictions so that we can measure and visualize errors in the predictions in the original space, allowing easy comparison across models.

## D.3 MODEL ARCHITECTURE

We summarize details related to model architecture, layers, and number of parameters related to seismic and velocity encoder-decoder architecture in Table 3, and regarding the latent models used for translation in Table 4. Further, we compare our model parameters with baseline models in Table 6.

### D.3.1 LATENT U-NET

Latent U-Net architectures are specifically adapted to subsurface imaging by employing U-Net-based encoder-decoder pairs designed to handle domain-specific input and output dimensions, such as velocity maps and seismic waveforms, while enabling latent dimension processing. Both the Large and Small variants utilize identical encoder and decoder designs, featuring 5 layers in the encoder and 6 layers in the decoder. These layers operate on an embedding dimension of 128x70x70, with channel sizes ranging from 8 to 128 in the encoder and 128 to 1 (velocity) or 128 to 5 (waveform) in the decoder.

The Large Latent U-Net incorporates 2 depth levels and 4 convolutional blocks per depth level in its latent space translation model, resulting in a total parameter size of 34.96M. The Small variant, designed for reduced complexity, includes 1 convolutional block per depth level in its latent translation model, reducing the total parameter size to 18.13M.

### D.3.2 INVERTIBLE X-NET

Invertible X-Net leverages an iUNet-based architecture to achieve bidirectional mappings, enabling consistent forward (velocity-to-waveform) and inverse (waveform-to-velocity) transformations within a shared framework. iUNet introduces invertible coupling blocks, ensuring bijectivity for the latent space translation model. Max pooling is replaced with orthogonal convolutional filters for downsampling, while upsampling is performed using orthogonal deconvolution filters to maintain invertibility.

While the latent space translation model in Invertible X-Net is fully invertible due to its iUNet structure, the encoder-decoder pairs are not invertible. Therefore, Invertible X-Net as a whole is not fully invertible. However, by employing two separate encoder-decoder pairs for velocity and waveform domains, Invertible X-Net achieves an architecture that enables bidirectional mappings between the two domains while preserving consistency.

Invertible X-Net features 4 depth levels, with 4 invertible coupling blocks per level, and adopts an encoder-decoder structure similar to Latent U-Net, with 5 encoder layers and 6 decoder layers, an embedding dimension of 128x70x70, and channel sizes ranging from 8 to 128. The latent translation model - iUNet consists of 25.78M parameters, contributing to a total model size of 26.06M.

Table 3: Architecture Details of Seismic Waveform and Velocity Encoder-Decoder models.

Model	#Layers	#Embedding Dim	Channels	#Params
Velocity Encoder	5	128x70x70	[8, 16, 32, 64, 128]	11632
Velocity Decoder	6	128x70x70	[128, 64, 32, 16, 1, 1]	27877
Seismic Waveform Encoder	5	128x70x70	[8, 16, 32, 64, 128]	55680
Seismic Waveform Decoder	6	128x70x70	[128, 64, 32, 16, 5, 5]	186497

Table 4: Architecture details of Latent U-Net and IU-Net latent space translation models

Model	#Depths	#Conv blocks/ Coupling Blocks	#Params
Latent U-Net	2	4	34.68M
IU-Net	4	4	25.78M

Table 5: Hyperparameter details for training Latent U-Net and Invertible X-Net models.

Model	#Epochs	Optimizer	LR	LR Scheduler
Latent U-Net	450	Adam	2e-3	StepLR
Invertible X-Net	450	Adam	2e-3	StepLR

Table 6: Comparison of encoder, decoder, and latent model parameters for our model (Latent U-Net and Invertible X-Net) with other baseline models. The parameters for Latent U-Nets and Invertible X-Nets are calculated for Latent dimension 70.

Model	#Vel Encoder Params	#Vel Decoder Params	#Amp Encoder Params	#Amp Decoder Params	#Translation Params	#Total Params
FNO	-	-	-	-	-	7.38M
InversionNet	-	9.34M	35.76M	-	Identity	24.41M
VelocityGAN	-	9.34M	35.76M	-	Identity	24.41M
Autolinear	12.98M	9.98M	2.29M	10.18M	16.5K	35.45M
Latent U-Net(Small)	11.6K	27.88K	55.68K	186.5K	17.86M	18.13M
Latent U-Net(Large)	11.6K	27.88K	55.68K	186.5K	34.68M	34.96M
Invertible X-Net	11.6K	27.88K	55.68K	186.5K	25.78M	26.06M

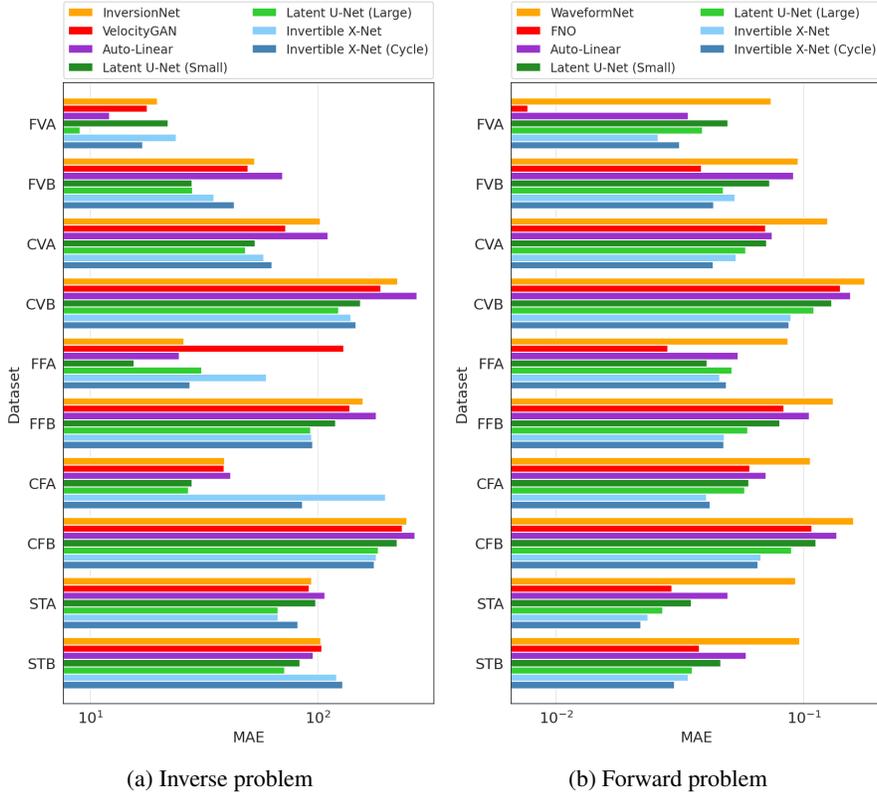


Figure 10: Comparison of Latent U-Nets (Small and Large), Invertible X-Net, Invertible X-Net (Cycle) with different baseline methods across different OpenFWI datasets.

## E ADDITIONAL RESULTS

### E.1 ADDITIONAL INVERSE MODELING RESULTS

We provide more detailed comparison of our models with other baseline models in Table 7. Our proposed models consistently outperform baseline models on multiple datasets indicating superior generalizability on in-distributions examples.

Additionally, we also show zero shot generalization of our models on the Marmousi and Overthrust dataset in Tables 8. Our model Invertible X-Net shows generalizability in SSIM indicating that overall prediction has better geological understanding than other baseline models.

Table 7: Comparison of our models (Latent U-Net (Large) and Invertible X-Net) with other baseline models for the inverse problem across 10 OpenFWI datasets. The bold highlights the best performing model on that dataset.

Metric	Model	FVA	FVB	CVA	CVB	FFA	FFB	CFA	CFB	STA	STB	Average Rank
MAE ↓	InversionNet	19.67	52.77	102.77	224.57	25.80	158.31	38.90	246.94	93.96	103.37	2.8
	Auto-Linear	12.16	70.05	110.72	273.02	<b>24.55</b>	181.39	41.38	268.47	107.85	95.63	3.5
	Latent U-Net	<b>9.01</b>	<b>28.11</b>	<b>48.05</b>	<b>123.56</b>	30.91	<b>92.91</b>	<b>27.05</b>	185.22	<b>67.00</b>	<b>71.41</b>	<b>1.3</b>
	Invertible X-Net	23.80	34.94	57.86	139.81	59.45	94.11	198.56	<b>181.03</b>	67.09	121.42	2.7
MSE ↓	InversionNet	1002.74	17271.62	36438.33	188044.46	4081.10	68214.92	9490.73	136327.84	23626.76	58622.35	2.8
	Auto-Linear	1053.03	33907.57	42391.39	236457.48	5952.25	81789.80	13715.42	154713.12	31274.60	21819.14	3.5
	Latent U-Net	<b>216.92</b>	<b>6464.06</b>	<b>12609.65</b>	<b>91784.50</b>	<b>2348.22</b>	33935.12	<b>3771.38</b>	93249.92	14353.55	<b>14564.94</b>	<b>1.3</b>
	Invertible X-Net	1245.52	6969.42	14659.46	96121.90	7719.55	<b>32559.95</b>	77105.41	<b>86512.36</b>	<b>13106.96</b>	31550.22	2.4
SSIM ↑	InversionNet	0.9894	0.9461	0.8073	0.6726	0.9765	0.7208	0.9566	0.6136	0.8858	0.6314	3
	Auto-Linear	0.9887	0.9044	0.8056	0.6169	0.97	0.6865	0.9424	0.5695	0.8422	0.7274	3.8
	Latent U-Net	<b>0.9967</b>	<b>0.9809</b>	<b>0.9273</b>	<b>0.8156</b>	<b>0.991</b>	0.8515	<b>0.98</b>	0.6930	0.9298	<b>0.8064</b>	<b>1.3</b>
	Invertible X-Net	0.9917	0.9769	0.9135	0.8076	0.9826	<b>0.8532</b>	0.9316	<b>0.7116</b>	<b>0.9360</b>	0.7913	1.9

Table 8: Comparison of our models (Latent U-Net (Large) and Invertible X-Net) on real world like datasets - Marmousi, Overthrust, Marmousi Smooth, Overthrust Smooth for the Inverse Problem. The bold highlights the best performing model on that dataset.

Metric	Model	Marmousi	Overthrust	Marmousi Smooth	Overthrust Smooth
MAE ↓	InversionNet	<b>282.39</b>	273.80	<b>179.34</b>	<b>148.34</b>
	Auto-Linear	285.38	298.89	206.38	207.41
	Latnet U-Net	322.13	<b>264.40</b>	242.71	198.53
	Invertible X-Net	298.39	308.50	245.86	258.26
MSE ↓	InversionNet	<b>160084.5</b>	135988.73	<b>67978.40</b>	<b>37804.58</b>
	Auto-Linear	159517.41	157434.56	78207.07	71128.92
	Latnet U-Net	217508.17	<b>122398.64</b>	112685.39	69122.58
	Invertible X-Net	180250.47	179093.10	107344.46	124700.75
SSIM ↑	InversionNet	0.46	0.4519	0.6044	<b>0.7217</b>
	Auto-Linear	0.438	0.4097	0.5423	0.6447
	Latnet U-Net	0.438	0.4807	0.6371	0.7031
	Invertible X-Net	<b>0.504</b>	<b>0.4827</b>	<b>0.6633</b>	0.6952

## E.2 ADDITIONAL FORWARD MODELING RESULTS

Similar to the inverse problem, we provide detailed comparison of our models with other baseline models in Table 9 and 10. Our proposed models consistently outperform baseline models on multiple datasets indicating superior in-distributions generalizability.

Table 9: Comparison of our models (Latent U-Net (Large) and Invertible X-Net) with other baseline models for the forward problem across 10 OpenFWI datasets. The bold highlights the best performing model on that dataset.

Metric	Model	FVA	FVB	CVA	CVB	FFA	FFB	CFA	CFB	STA	STB	Average Rank
MAE ↓	FNO	<b>0.0077</b>	<b>0.0385</b>	0.0700	0.1405	<b>0.0282</b>	0.0829	0.0605	0.1075	0.0292	0.0379	2.4
	Auto-Linear	0.0340	0.0906	0.0744	0.1537	0.0541	0.1048	0.0703	0.1356	0.0492	0.0584	3.9
	Latent U-Net	0.0389	0.0473	0.0581	0.1098	0.0512	0.059	0.0576	0.0891	0.0269	0.0354	2.3
	Invertible X-Net	0.0257	0.0527	<b>0.0532</b>	<b>0.0887</b>	0.0457	<b>0.0477</b>	<b>0.0404</b>	<b>0.0671</b>	<b>0.0235</b>	<b>0.0341</b>	<b>1.4</b>
MSE ↓	FNO	<b>0.0004</b>	<b>0.0066</b>	0.0259	0.0825	0.0059	0.0358	0.0251	0.0471	0.0041	<b>0.0053</b>	1.9
	Auto-Linear	0.0084	0.0427	0.0303	0.1045	0.0217	0.0625	0.033	0.0866	0.0174	0.0173	3.7
	Latent U-Net	0.0173	0.0127	<b>0.0160</b>	0.0489	0.0344	0.0175	0.0442	0.0297	0.0053	0.0082	2.6
	Invertible X-Net	0.0062	0.0176	0.0175	<b>0.0305</b>	<b>0.0185</b>	<b>0.0117</b>	<b>0.0126</b>	<b>0.0163</b>	<b>0.004</b>	0.0087	<b>1.8</b>
SSIM ↑	FNO	<b>0.9967</b>	0.9781	0.8881	0.8354	0.9667	0.8702	0.9166	0.8160	0.9655	0.9417	3.1
	Auto-Linear	0.9694	0.9289	0.9038	0.8567	0.9470	0.8736	0.9253	0.8188	0.9569	0.9300	3.5
	Latent U-Net	0.9764	0.9779	0.9237	0.89	0.9659	0.9283	0.9571	0.8636	0.9819	0.9702	2.3
	Invertible X-Net	0.9887	<b>0.9782</b>	<b>0.9559</b>	<b>0.9221</b>	<b>0.9744</b>	<b>0.954</b>	<b>0.9757</b>	<b>0.9156</b>	<b>0.989</b>	<b>0.9805</b>	<b>1.1</b>

## E.3 IMPORTANCE OF COMBINED LOSS FUNCTION FOR INVERTIBLE-XNET

In this section, we focus on the training of the Invertible X-Net model using a combined loss function (incorporating both the forward and inverse problems), as opposed to training it solely with a forward loss function. Figure 11 shows that when model is trained using only forward loss, then its performance falls short compared to the Latent U-Net (Large) model. This discrepancy can be attributed to the fact that the Latent U-Net has a higher model complexity than Invertible X-Net, despite their similar sizes. Nonetheless, when the Invertible X-Net model is trained with combined loss (forward and inverse), the model is able to outperform Latent U-Net model with a good mar-

Table 10: Comparison of our models (Latent U-Net (Large) and Invertible X-Net) on real world like datasets - Marmousi, Overthrust, Marmousi Smooth, Overthrust Smooth for the Forward Problem. The bold highlights the best performing model on that dataset.

Metric	Model	Marmousi	Overthrust	Marmousi Smooth	Overthrust Smooth
MAE ↓	FNO	0.1484	0.2726	0.1077	0.1892
	Auto-Linear	0.2818	0.3018	0.2821	0.2730
	Latnet U-Net	<b>0.1338</b>	0.2502	<b>0.1013</b>	<b>0.1875</b>
	Invertible X-Net	0.1425	<b>0.2311</b>	0.1056	0.2062
MSE ↓	FNO	<b>0.1110</b>	0.580	<b>0.0811</b>	<b>0.3495</b>
	Auto-Linear	0.4227	0.6203	0.5325	0.5593
	Latnet U-Net	0.1116	0.5494	0.0927	0.4133
	Invertible X-Net	0.1168	<b>0.4026</b>	0.08130	0.5049
SSIM ↑	FNO	0.8148	0.7404	0.9021	0.8447
	Auto-Linear	0.6344	0.672	0.6670	0.7192
	Latnet U-Net	<b>0.8343</b>	0.7700	<b>0.9143</b>	<b>0.8626</b>
	Invertible X-Net	0.827	<b>0.7863</b>	0.9112	0.8500

gin. This highlights the value of joint training, demonstrating that simultaneously learning both the forward and inverse problems can lead to better results than learning the two models separately.

In Figures [12](#) and [13](#), we illustrate asymmetry in learning the translation for the forward and inverse problems using CVB and CFA datasets. From the figures, we observe that the model learns the inverse mapping in the initial epochs and gradually starts to learn the solution to the forward problem in later epochs. Since the network optimizes the combined loss function on both velocity and waveform together, the gradients from the combined loss help the model to achieve better forward solution. This corroborates with our hypothesis that the model trained on combined loss is able to learn the connection between forward and inverse problem.

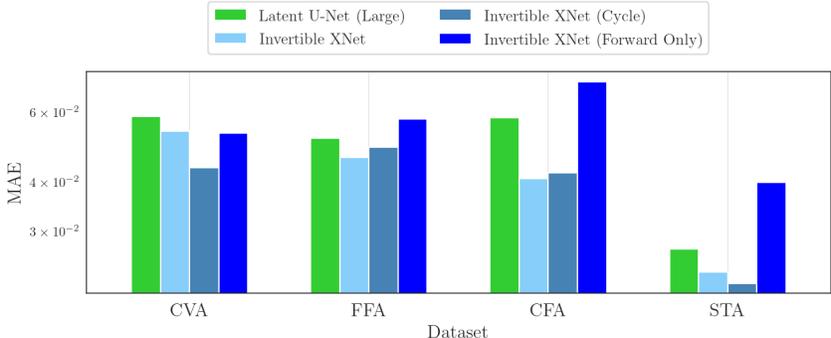


Figure 11: Comparison of Latent U-Net (Large), Invertible X-Net, Invertible X-Net (Forward Only), and Invertible X-Net (Cycle) on the forward problem across various OpenFWI datasets.

#### E.4 ZERO-SHOT PERFORMANCE

In this part, we provide detailed insights into zero shot performance of our models (Latent U-Net (Large) and Invertible X-Net) with other baselines across all datasets. This investigation helps us understand overall out-of-distribution generalization of a model and underscore the importance of learning the translation problem in the latent space.

## E.5 INVERSE PROBLEM

Figure 14 and 15 show generalization performance of Invertible X-Net and Latent U-Net (Large) models respectively on the inverse problem using MAE, MSE, and SSIM metrics. As described in the main paper, we show models trained across all dataset as rows and evaluated across all test datasets as columns. For comparison, we evaluate metrics such as MAE, MSE, and SSIM and calculate its difference between our models and other baseline models. For MAE and MSE, when the color intensity is blue, our model show better generalizability and vice-versa whereas, for SSIM, when the color intensity is red indicates better generalization of our model and vice-versa.

From Figure 14, we observe that the Invertible X-Net shows superior generalization over the baseline models (AutoLinear, InversionNet, and VelocityGAN) across all metrics except on the FVB dataset. Figure 14(d) compare Invertible X-Net with Latent U-Net (Large) model where we see that Invertible X-Net shows better generalization on complex datasets such as CFB, STA, and STB datasets, while Latent U-Net is better on relatively simpler datasets such as CVA, CFA, and more. Similarly, Figure 15 shows the comparison of Latent U-Net (Large) model with other baseline models. As expected, we observe that our model is able to generalize much better than all the baseline models.

## E.6 FORWARD PROBLEM

Similar to the inverse problem, we analyze out-of-distribution generalization of our models against baseline models across all evaluation metrics.

Figures 16 and 17 compares the performance of our models Invertible X-Net and Latent U-Net (Large) against baselines. In Figure 16, we observe strong generalization of Invertible X-Net over all baselines - AutoLinear, FNO, and WaveformNet. Figure 16(d) shows the comparative performance of Invertible X-Net against Latent U-Net (Large) where we see that Invertible X-Net dominates overall across all metrics. In Figure 17, we compare the Latent U-Net (Large) model against same baselines as above. The figure indicates Latent U-Net (Large) model has much stronger generalizability than baseline models consistently.

## F ABLATION STUDIES

### F.1 EFFECT OF VARYING LATENT SPACE SIZES

Figure 18a(a) shows how the performance of Latent U-Net model is affected with change in the latent space size. As the latent space size is reduced, the MAE and MSE metrics for mapping seismic waveform to velocity is increasing, while the SSIM is decreasing. The impact is more pronounced on complex datasets such as B group of datasets. These datasets represent geologically complex datasets and therefore may require larger latent space to encode geological heterogeneity. In Figure 18b, we compare the large Latent U-Net with the small Latent U-Net model, while latent space size is also reduced. We observe that the performance gap between the two models reduces as the latent space size is also reduced. This underscore the importance of latent space size for encoding the geological features for an effective translation.

Further, we provide visualizations of the first two primary PCA and t-SNE components of the velocity latent space in Figure 19 and 20 respectively. We take the encoder trained on a dataset and get the latent space encoding on several datasets. This visualization shows the in-distribution and out-of-distribution generality of our encoders and highlights the importance of the manifold for latent space translation. Overall, we observe that the larger latent space  $70 \times 70$  have better structure than  $8 \times 8$ , indicating that the ideal size of latent space should be decided based on the complexity of dataset and problem being solved.

### F.2 EFFECT OF SKIP CONNECTIONS FOR LATENT U-NET

In this section, we study the impact of skip connections of Latent U-Net model for the inverse problem. Figure 21 shows how the MAE and MSE are increasing and SSIM is decreasing when the latent space size is decreasing from  $70 \times 70$  to  $8 \times 8$ . We observe that the impact of skip connections is more pronounced at smaller latent space size as opposed to larger space.

### F.3 EFFECT OF MANIFOLD LEARNING

Figure 22 for Latent U-Net Small further elaborates on our finding that direct translation learning consistently outperforms the two-stage approach, where translation follows reconstruction.

## G NOISE EXPERIMENTS

Table 11: Quantitative results on CurveFault-B with Gaussian noise of varying variance  $\sigma^2$  added during testing for the inverse problem.

Model	$\sigma^2 = 0$			$\sigma^2 = 1e-5$ PSNR=84.07dB			$\sigma^2 = 5e-5$ PSNR=77.08dB			$\sigma^2 = 1e-4$ PSNR=74.07dB			$\sigma^2 = 5e-4$ PSNR=67.08dB		
	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM
Latnet U-Net	185.22	93248.64	0.6930	185.54	93538.78	0.6926	187.03	94674.546	0.6917	188.56	95777.95	0.6906	195.94	101444.7	0.6844
Degradation (%)	(-)	(-)	(-)	0.17%	0.31%	0.05%	0.97%	1.52%	0.20%	1.80%	2.71%	0.35%	5.78%	8.78%	1.24%
Invertible X-Net	180.97	86465.25	0.7117	181.32	86673.38	0.7115	182.83	87674.14	0.7106	184.17	88585.86	0.7097	192.22	94728.94	0.7036
Degradation (%)	(-)	(-)	(-)	0.19%	0.24%	0.02%	1.02%	1.39%	0.15%	1.76%	2.45%	0.27%	6.21%	9.55%	1.13%
Auto-Linear	268.47	154713.10	0.5695	270.29	156201.81	0.5682	277.05	162156.50	0.5632	285.34	169892.26	0.5572	325.02	211616.29	0.5190
Degradation (%)	(-)	(-)	(-)	0.67%	0.96%	0.21%	3.19%	4.81%	1.10%	6.28%	9.81%	2.15%	21.06%	36.77%	8.86%

Table 12: Quantitative results on FlatFault-B with Gaussian noise of varying variance  $\sigma^2$  added during testing for the inverse problem.

Model	$\sigma^2 = 0$			$\sigma^2 = 1e-5$ PSNR=84.07dB			$\sigma^2 = 5e-5$ PSNR=77.08dB			$\sigma^2 = 1e-4$ PSNR=74.07dB			$\sigma^2 = 5e-4$ PSNR=67.08dB		
	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM
Latnet U-Net	92.93	33925.46	0.8515	93.47	34137.49	0.8511	95.43	34907.52	0.8499	98.19	36002.58	0.8482	108.83	40770.50	0.8417
Degradation (%)	(-)	(-)	(-)	0.58%	0.62%	0.05%	2.68%	2.89%	0.18%	5.66%	6.12%	0.38%	17.10%	20.17%	1.15%
Invertible X-Net	94.05	32548.49	0.8532	96.23	33283.08	0.8529	104.75	36522.60	0.85	110.30	39016.04	0.8474	122.66	45509.88	0.8382
Degradation (%)	(-)	(-)	(-)	2.31%	2.25%	0.03%	11.37%	12.20%	0.37%	17.27%	19.87%	0.67%	30.41%	39.82%	1.76%
Auto-Linear	181.39	81789.80	0.6865	187.40	85327.79	0.6821	206.58	98616.76	0.6664	221.79	110396.50	0.6511	283.50	167482.73	0.5796
Degradation (%)	(-)	(-)	(-)	3.31%	4.32%	0.64%	13.88%	20.57%	2.93%	22.26%	34.97%	5.16%	56.28%	104.77%	15.57%

Table 13: Quantitative results on CurveFault-B with Gaussian noise of varying variance  $\sigma^2$  added during testing for the forward problem.

Model	$\sigma^2 = 0$			$\sigma^2 = 1e-5$ PSNR=56.02dB			$\sigma^2 = 5e-5$ PSNR=49.03dB			$\sigma^2 = 1e-4$ PSNR=46.02dB			$\sigma^2 = 5e-4$ PSNR=39.03dB		
	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM
Latnet U-Net	0.0891	0.0297	0.8636	0.0891	0.0297	0.8635	0.0892	0.0297	0.8634	0.0897	0.0301	0.8630	0.0966	0.0396	0.8565
Degradation (%)	(-)	(-)	(-)	0%	0%	0.01%	0.13%	0.30%	0.01%	0.60%	1.40%	0.06%	8.37%	33.47%	0.82%
Invertible X-Net	0.0671	0.0163	0.9157	0.0672	0.0164	0.9156	0.0674	0.0164	0.9152	0.0678	0.0166	0.9147	0.0719	0.0193	0.9090
Degradation (%)	(-)	(-)	(-)	0.07%	0.07%	0.009%	0.42%	0.54%	0.04%	1.00%	1.68%	0.10%	7.08%	18.18%	0.73%
Auto-Linear	0.1356	0.0866	0.8188	0.1356	0.0868	0.8187	0.1360	0.0873	0.8183	0.1365	0.088	0.8179	0.139	0.0919	0.8159
Degradation (%)	(-)	(-)	(-)	0.06%	0.14%	0.01%	0.34%	0.78%	0.05%	0.70%	1.61%	0.09%	2.53%	6.03%	0.35%

## H ADDITIONAL VISUALIZATIONS

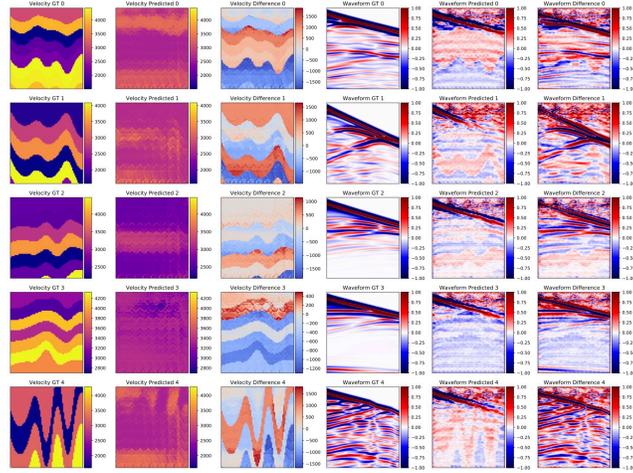
Here, we provide additional visualization of the waveform and velocity predictions for baselines and our models, namely Latent-UNet (small), Latent-UNet (large), Invertible-XNet, and Invertible-XNet (cycle), for both forward and inverse problems. Please note that we show the prediction of seismic waveform and velocity in the original space by unnormalizing the predictions for every model (Figures 23 - 34).

Table 14: Quantitative results on FlatFault-B with Gaussian noise of varying variance  $\sigma^2$  added during testing for the forward problem.

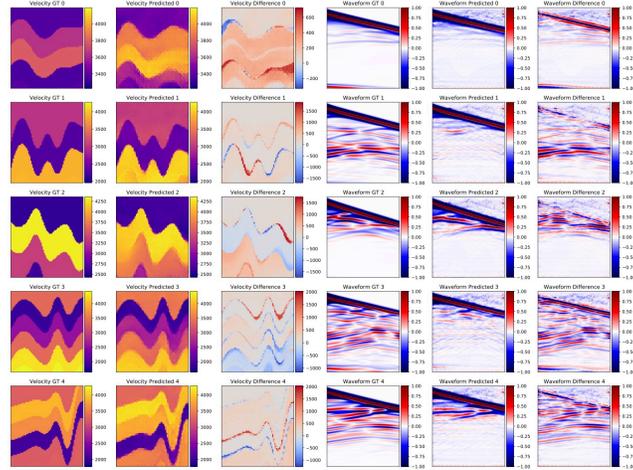
Model	$\sigma^2 = 0$			$\sigma^2 = 1e-5$ PSNR=56.02dB			$\sigma^2 = 5e-5$ PSNR=49.03dB			$\sigma^2 = 1e-4$ PSNR=46.02dB			$\sigma^2 = 5e-4$ PSNR=39.03dB		
	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM	MAE	MSE	SSIM
Latent U-Net	0.0594	0.0175	0.9283	0.0594	0.0175	0.9283	0.0598	0.0179	0.9279	0.0606	0.0187	0.9271	0.0694	0.0325	0.9178
Degradation (%)	(-)	(-)	(-)	0%	0%	0%	0.66%	2.16%	0.04%	1.99%	6.99%	0.12%	16.87%	85.99%	1.12%
Invertible X-Net	0.0477	0.0117	0.9540	0.0478	0.0117	0.9540	0.0482	0.0119	0.9536	0.0487	0.0122	0.9531	0.0535	0.016	0.9463
Degradation (%)	(-)	(-)	(-)	0.17%	0.33%	0%	0.94%	1.88%	0.04%	1.93%	4.26%	0.10%	12.16%	36.39%	0.80%
Auto-Linear	0.1048	0.0625	0.8736	0.1051	0.0628	0.8734	0.1061	0.0641	0.8726	0.1072	0.0656	0.8717	0.112	0.0727	0.868
Degradation (%)	(-)	(-)	(-)	0.25%	0.54%	0.02%	1.19%	2.59%	0.11%	2.24%	5.06%	0.21%	6.86%	16.37%	0.64%

## I CODE APPENDIX

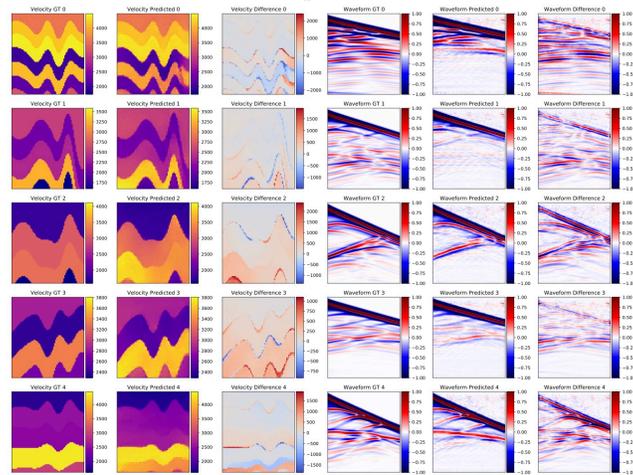
All the code required to train and evaluate the proposed methods, as well as the baselines, has been uploaded to an anonymous GitHub repository: <https://github.com/KGML-lab/Generalized-Forward-Inverse-Framework-for-DL4SI>. The data and corresponding processing code used in this work are sourced from the OpenFWI website: <https://openfwi-lanl.github.io/docs/data.html>.



(a) Epoch: 1

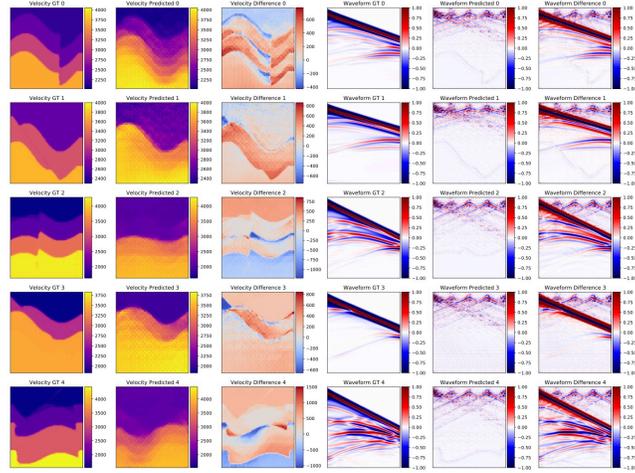


(b) Epoch: 10

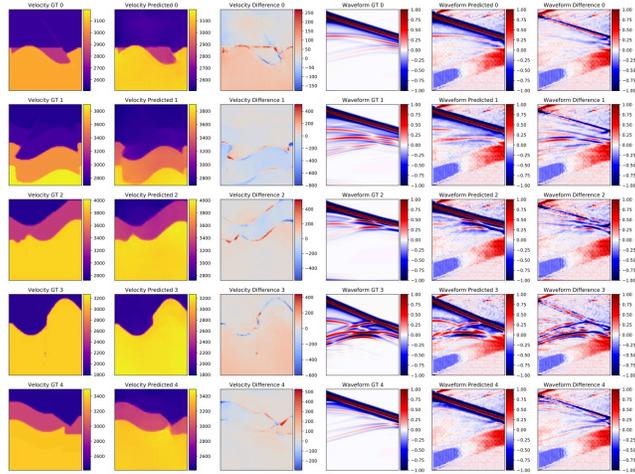


(c) Epoch: 20

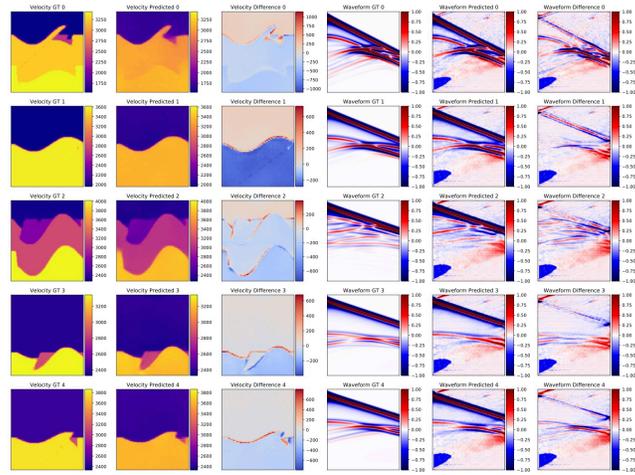
Figure 12: Training of Invertible X-Net model on the CVB dataset illustrating velocity and seismic waveform learning with epochs.



(a) Epoch: 1



(b) Epoch: 10



(c) Epoch: 20

Figure 13: Training of Invertible X-Net model on the CFA dataset illustrating velocity and seismic waveform learning with epochs.

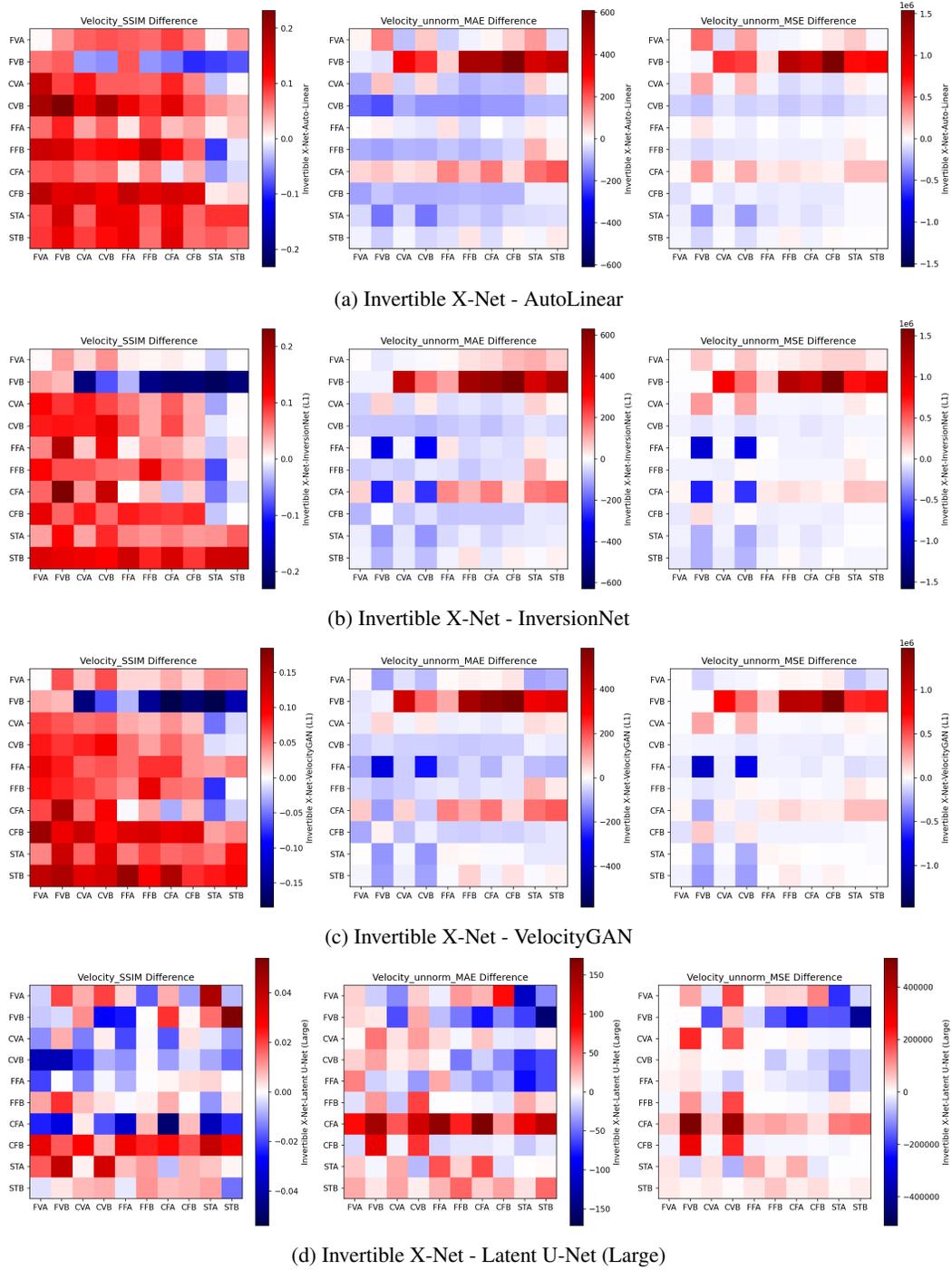


Figure 14: Out-of-distribution zero shot generalizations for the inverse problem of Invertible X-Net with AutoLinear, InversionNet, VelocityGAN, Latent U-Net (Large).

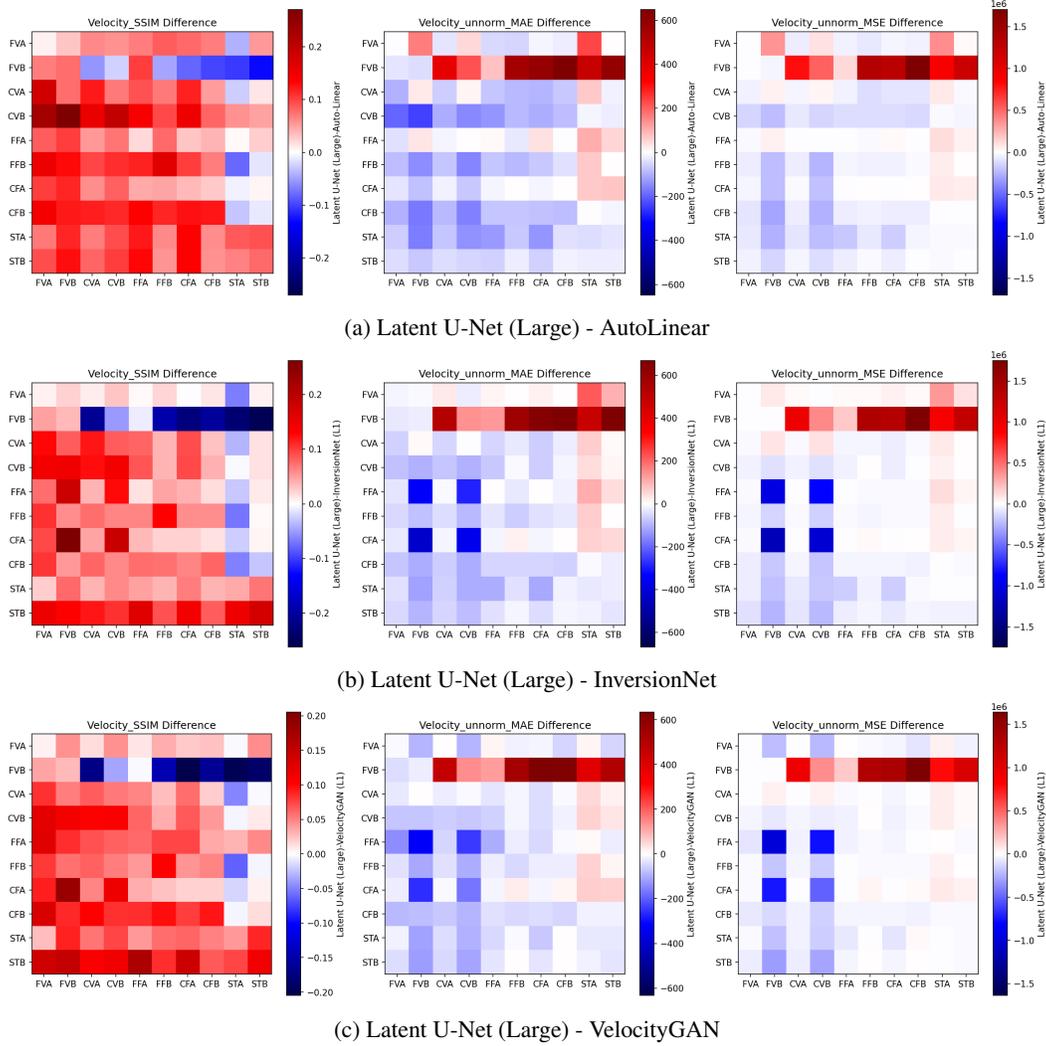


Figure 15: Out-of-distribution zero shot generalizations for the inverse problem of Latent U-Net (Large) with AutoLinear, InversionNet, VelocityGAN.

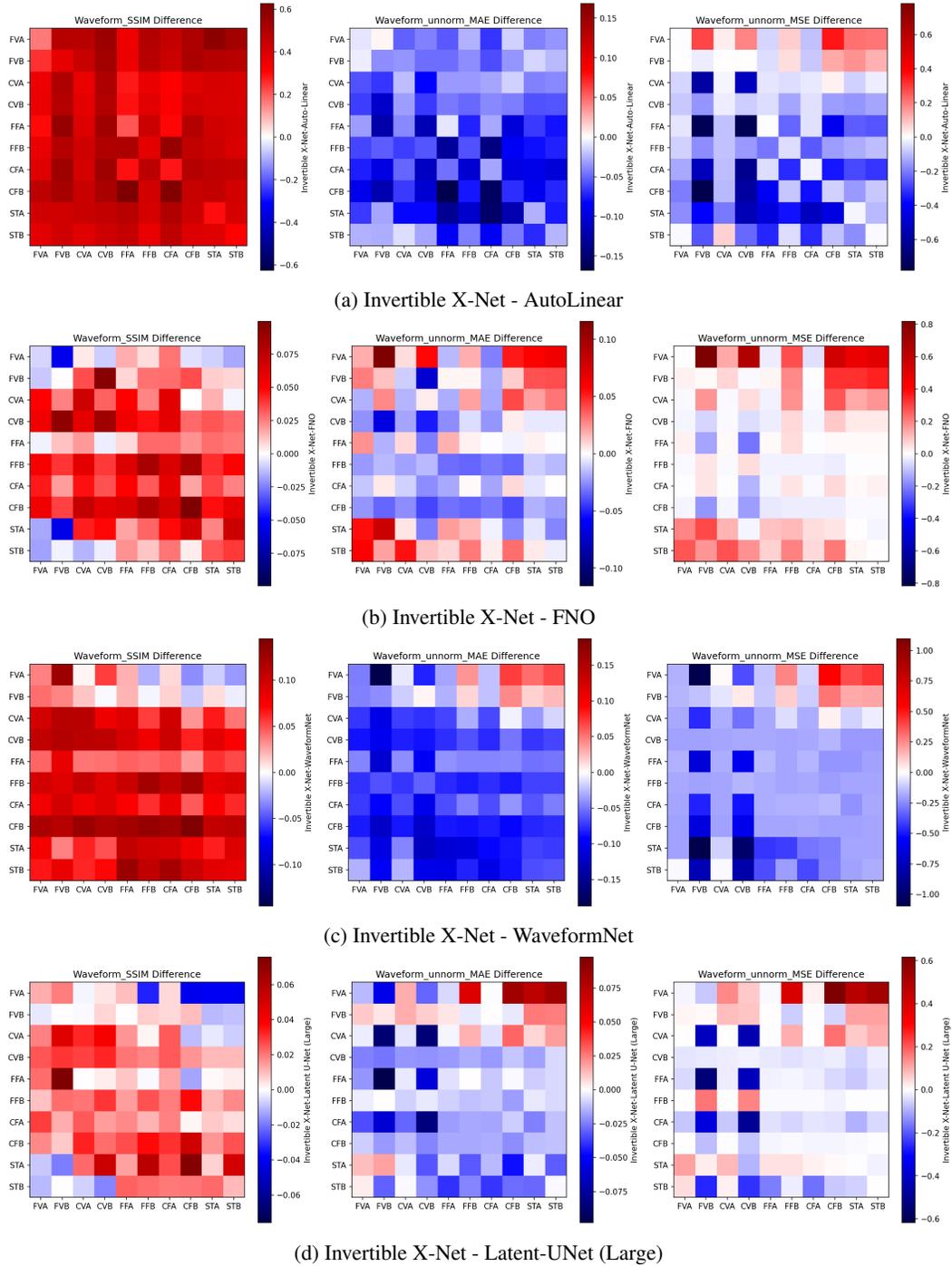
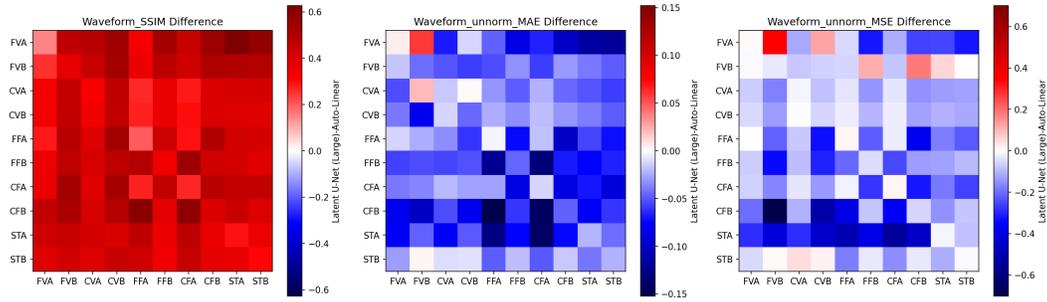
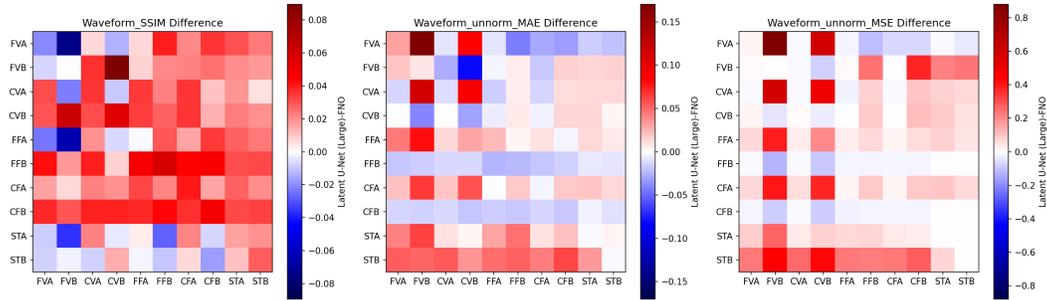


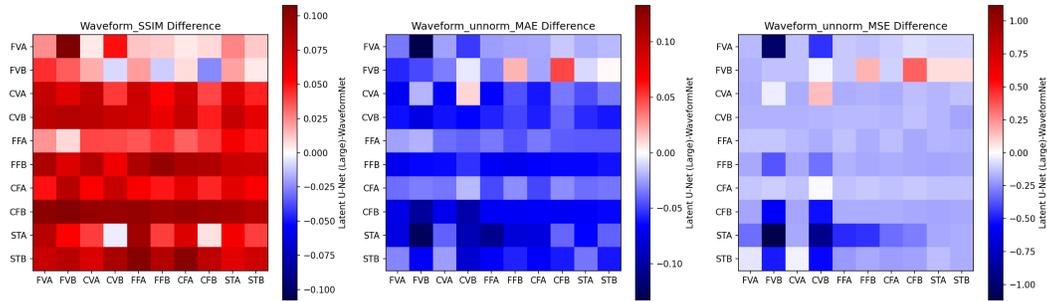
Figure 16: Out-of-distribution zero shot generalizations for the forward problem of Invertible X-Net with AutoLinear, FNO, WaveformNet (U-Net like model), and Latent U-Net (Large).



(a) Latnet U-Net (Large) - AutoLinear

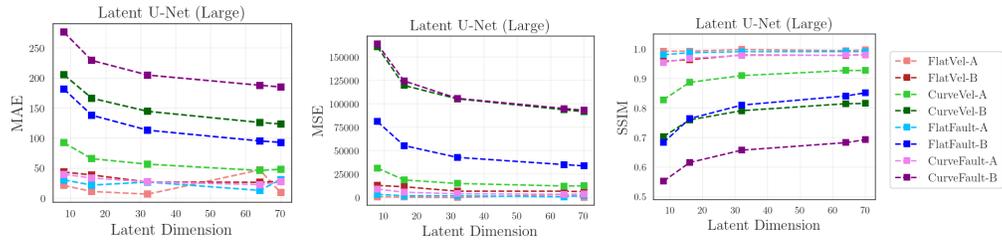


(b) Latnet U-Net (Large) - FNO

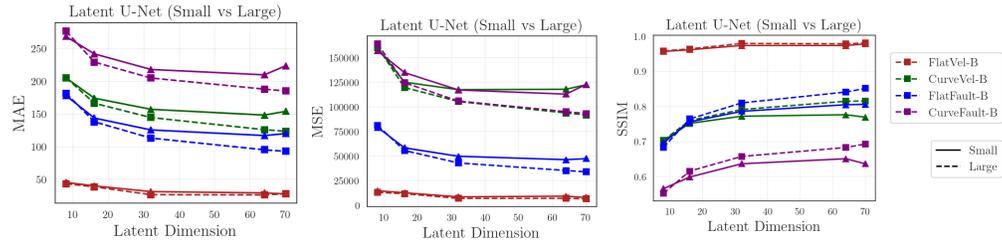


(c) Latnet U-Net (Large) - WaveformNet

Figure 17: Out-of-distribution zero shot generalizations for the forward problem of Latnet U-Net with AutoLinear, FNO, and WaveformNet (U-Net like model).



(a) Effect of Latent dimension on U-Net Large



(b) Effect of Latent dimension on U-Net Small in comparison to U-Net Small

Figure 18: Effect of the size of latent sizes on the performance of large and small Latent U-Net (small and large) on the OpenFWI datasets.

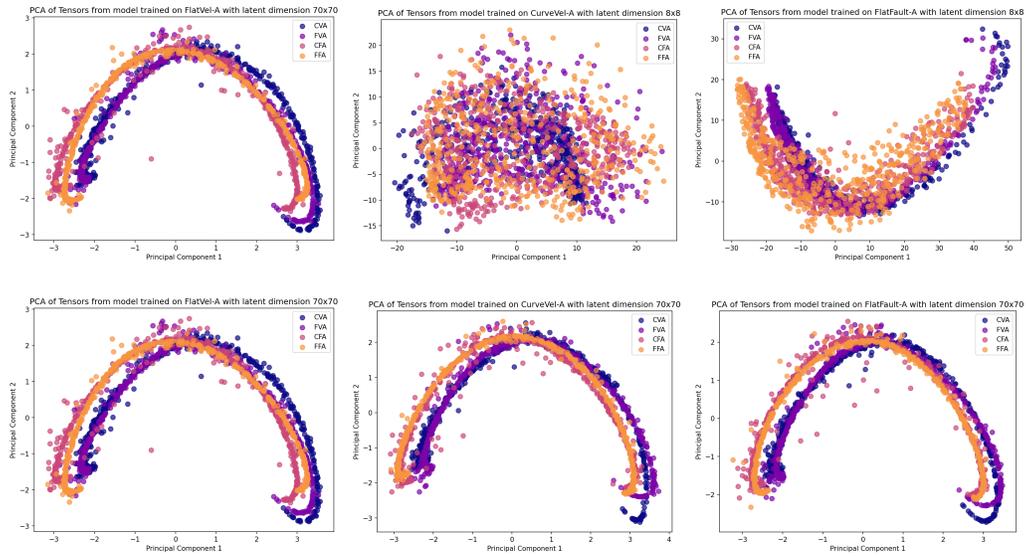


Figure 19: Visualizing the PCA projection of Velocity latent space for  $8 \times 8$  and  $70 \times 70$ .

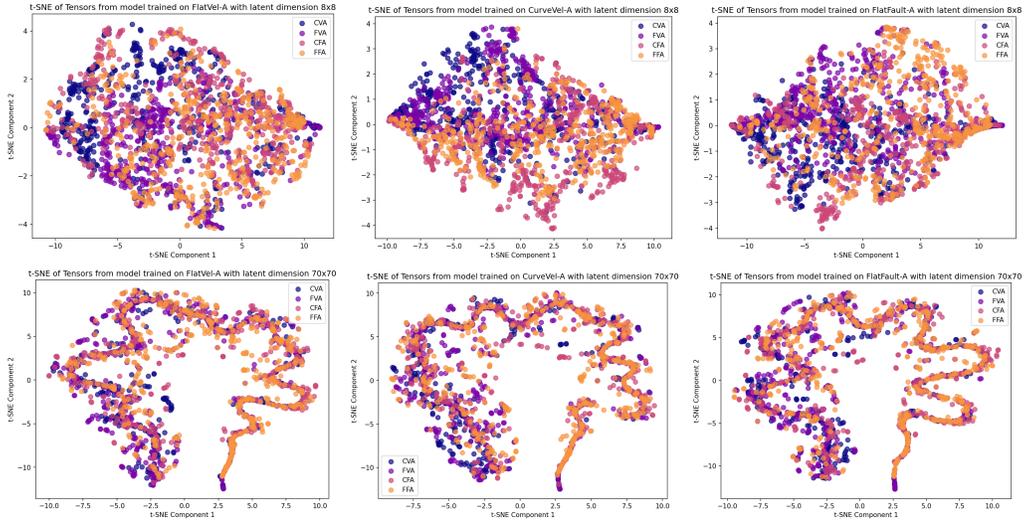


Figure 20: Comparing the t-SNE plot of latent space for velocity encoder.

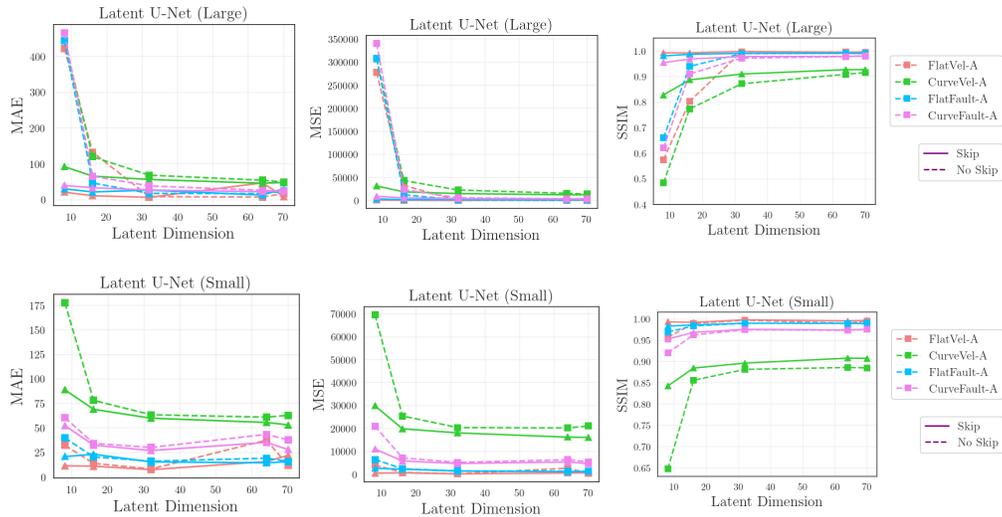


Figure 21: Effect of the size of latent space and skip vs no skip connections on the performance of large and small Latent U-Net models across OpenFWI datasets.

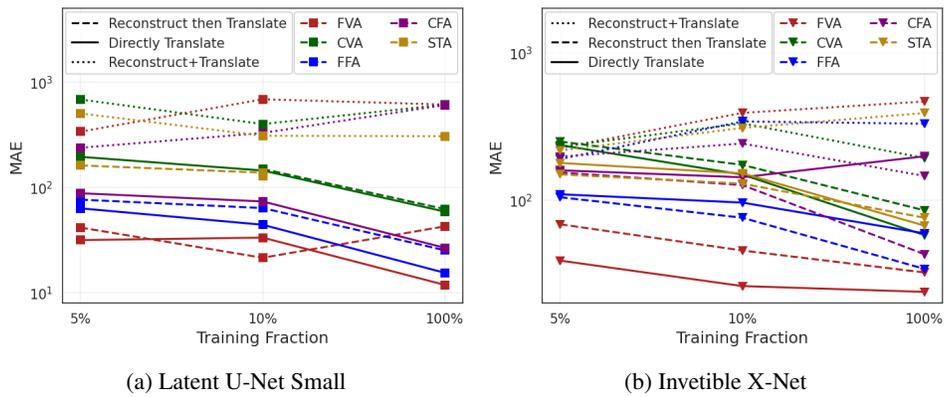


Figure 22: Comparison of Latent U-Net’s and Invertible X-Net’s performance across three learning objectives: translation directly, reconstruction followed by translation, and combined learning of both, evaluated at different training fractions.

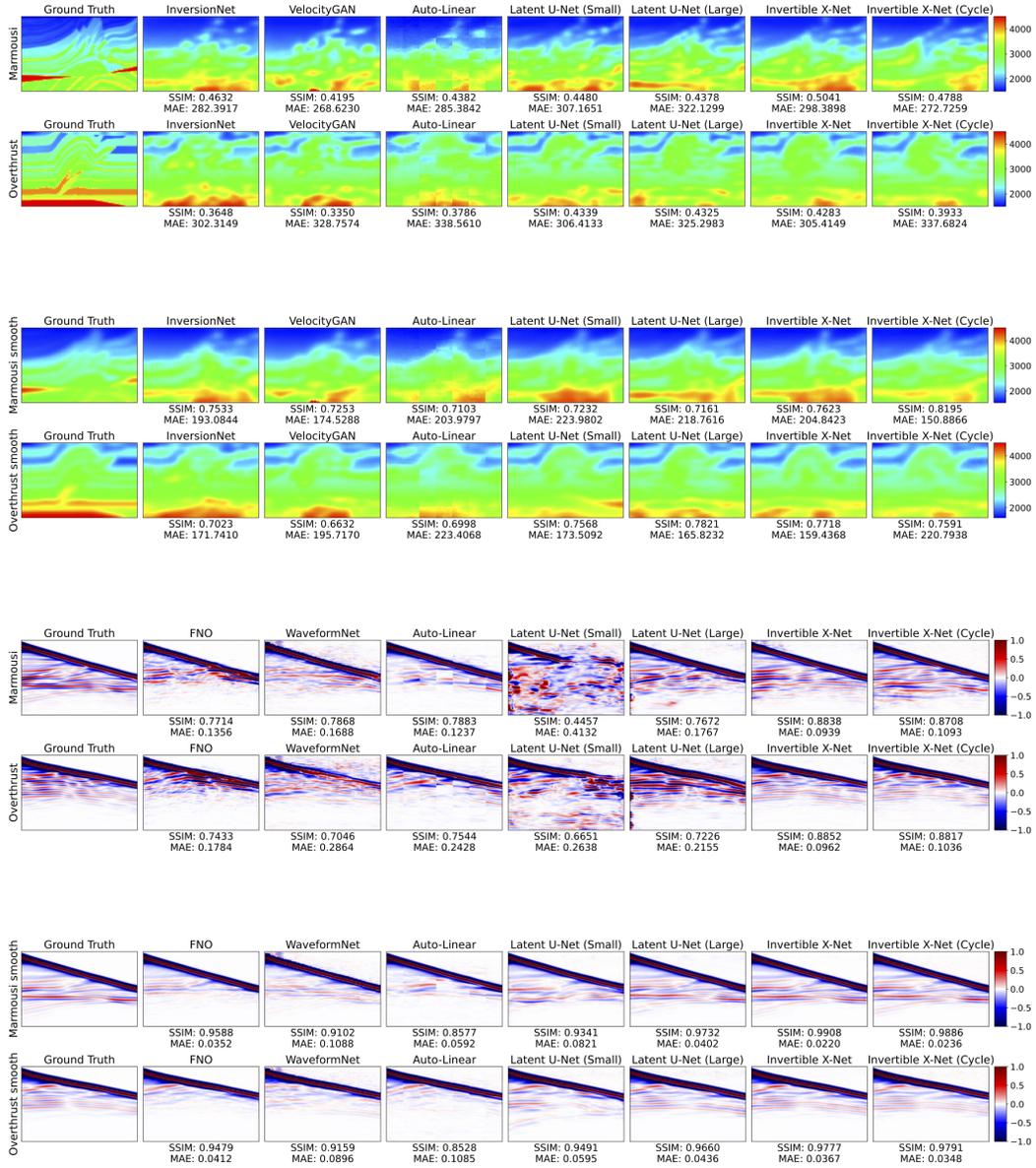


Figure 23: Zero shot generalization results of model trained on Style-A dataset on Marmousi and Overthrust dataset samples and their smoothed versions.

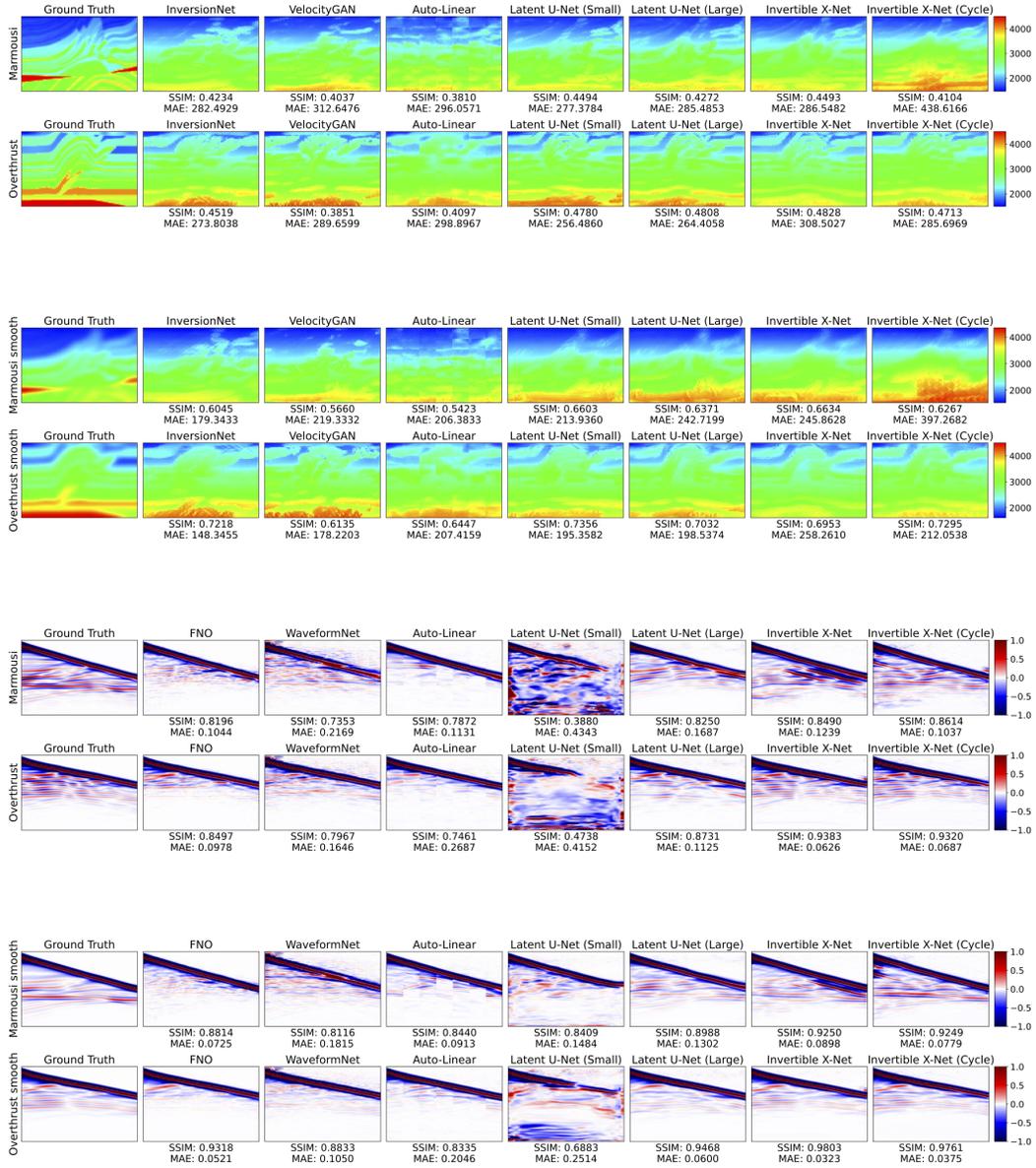


Figure 24: Zero shot generalization results of model trained on Style-B dataset on Marmousi and Overthrust dataset samples and their smoothed versions.



Figure 25: Visualization of predictions for forward and inverse problems on FVA dataset.

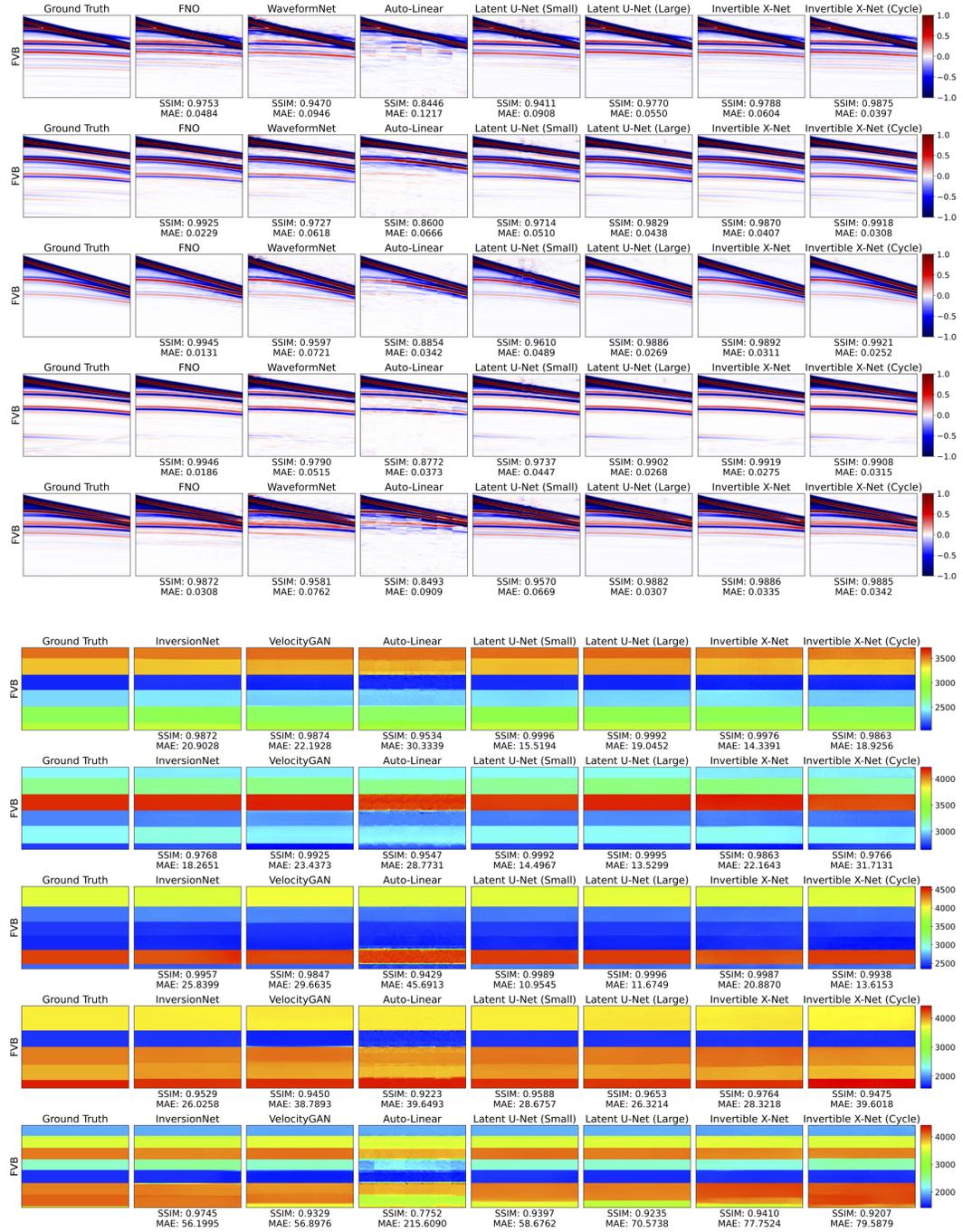


Figure 26: Visualization of predictions for forward and inverse problems on FVB dataset.

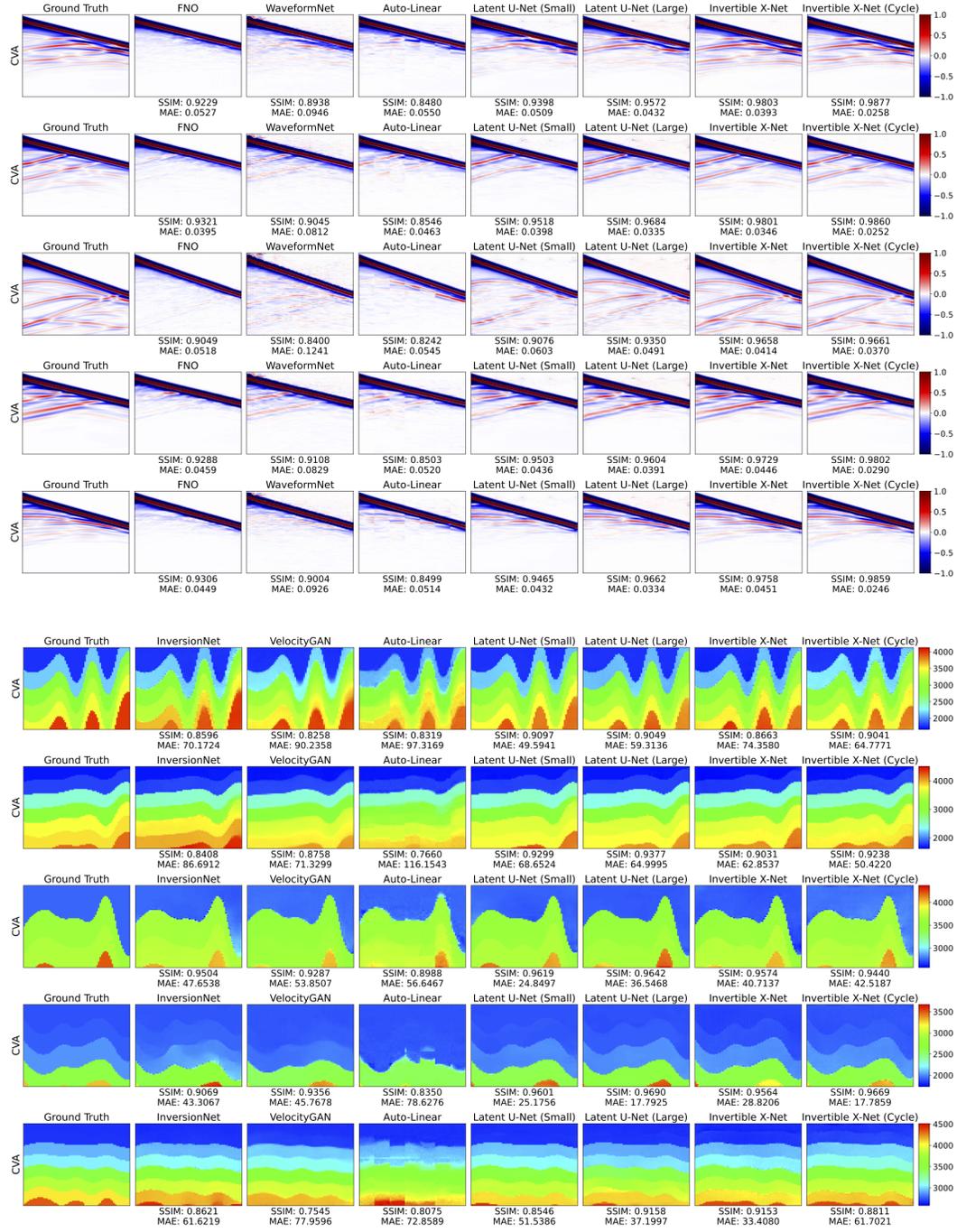


Figure 27: Visualization of predictions for forward and inverse problems on CVA dataset.

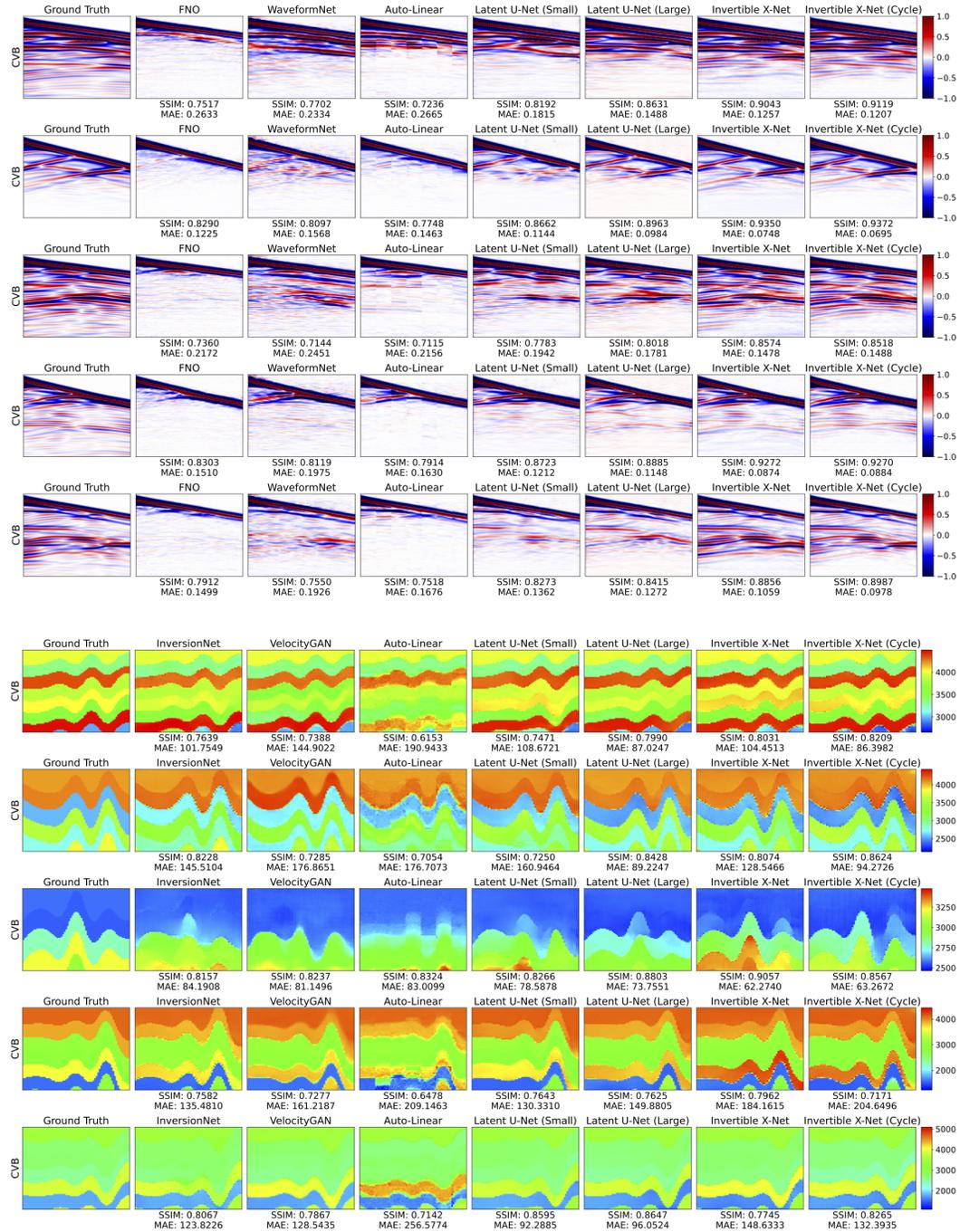


Figure 28: Visualization of predictions for forward and inverse problems on CVB dataset.

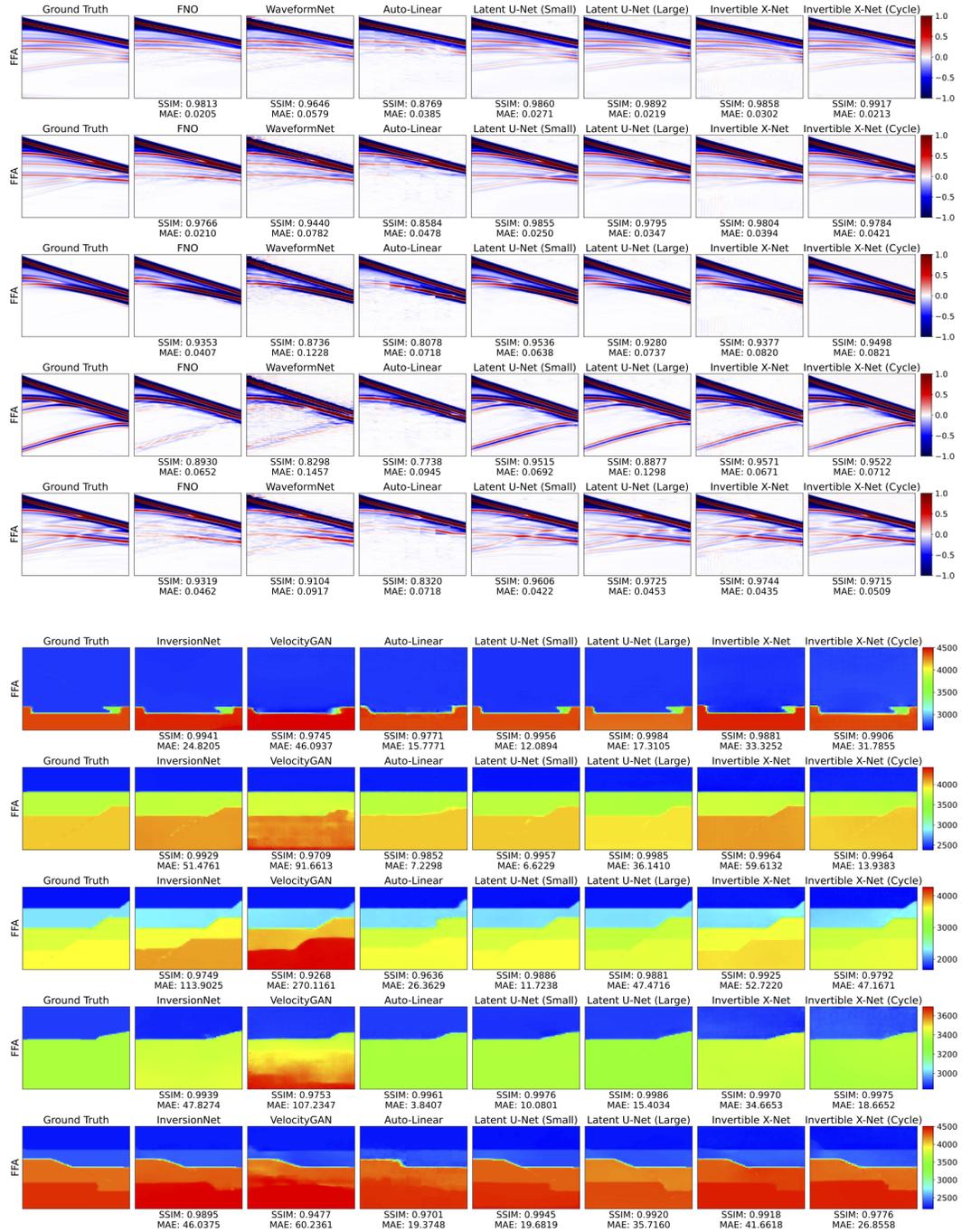


Figure 29: Visualization of predictions for forward and inverse problems on FFA dataset.

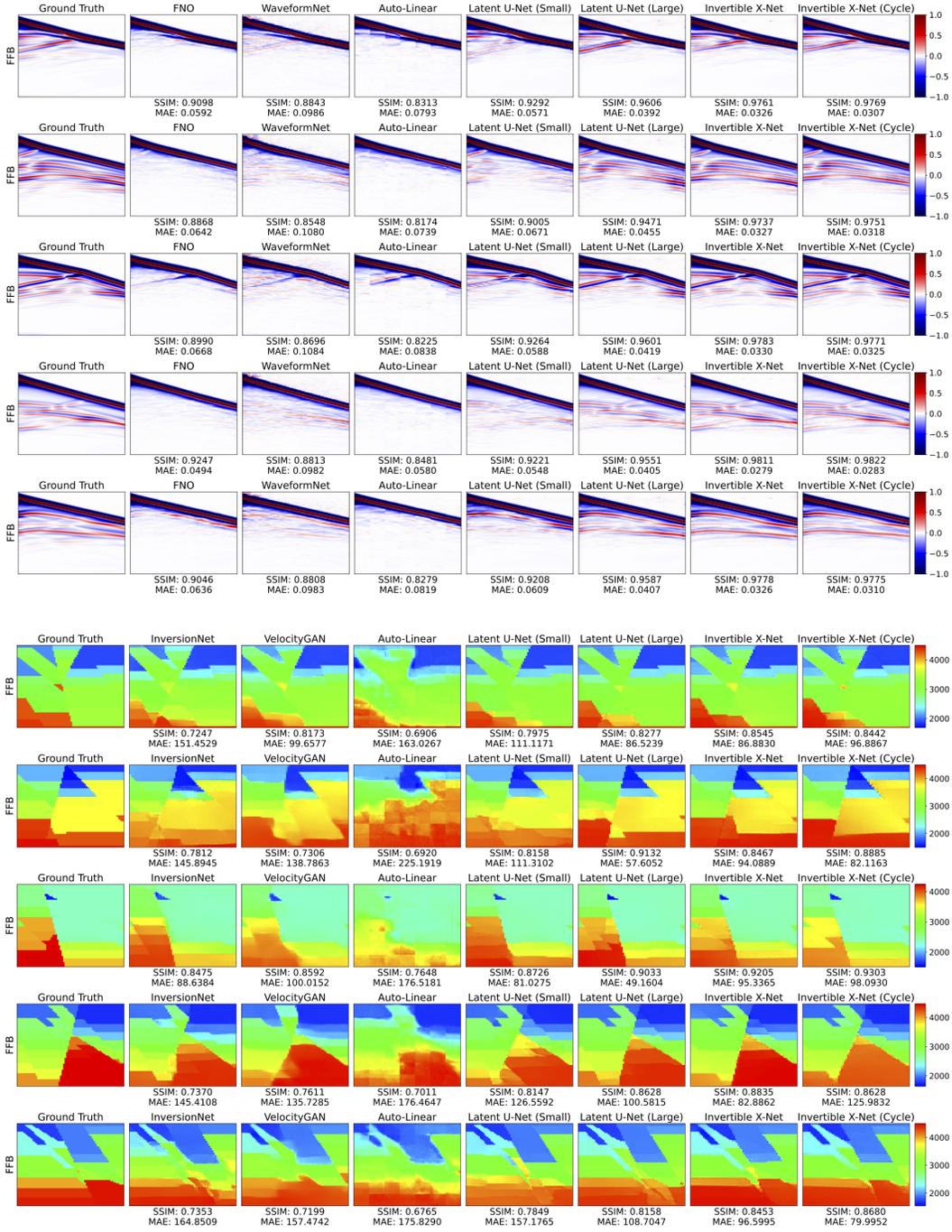


Figure 30: Visualization of predictions for forward and inverse problems on FFB dataset.

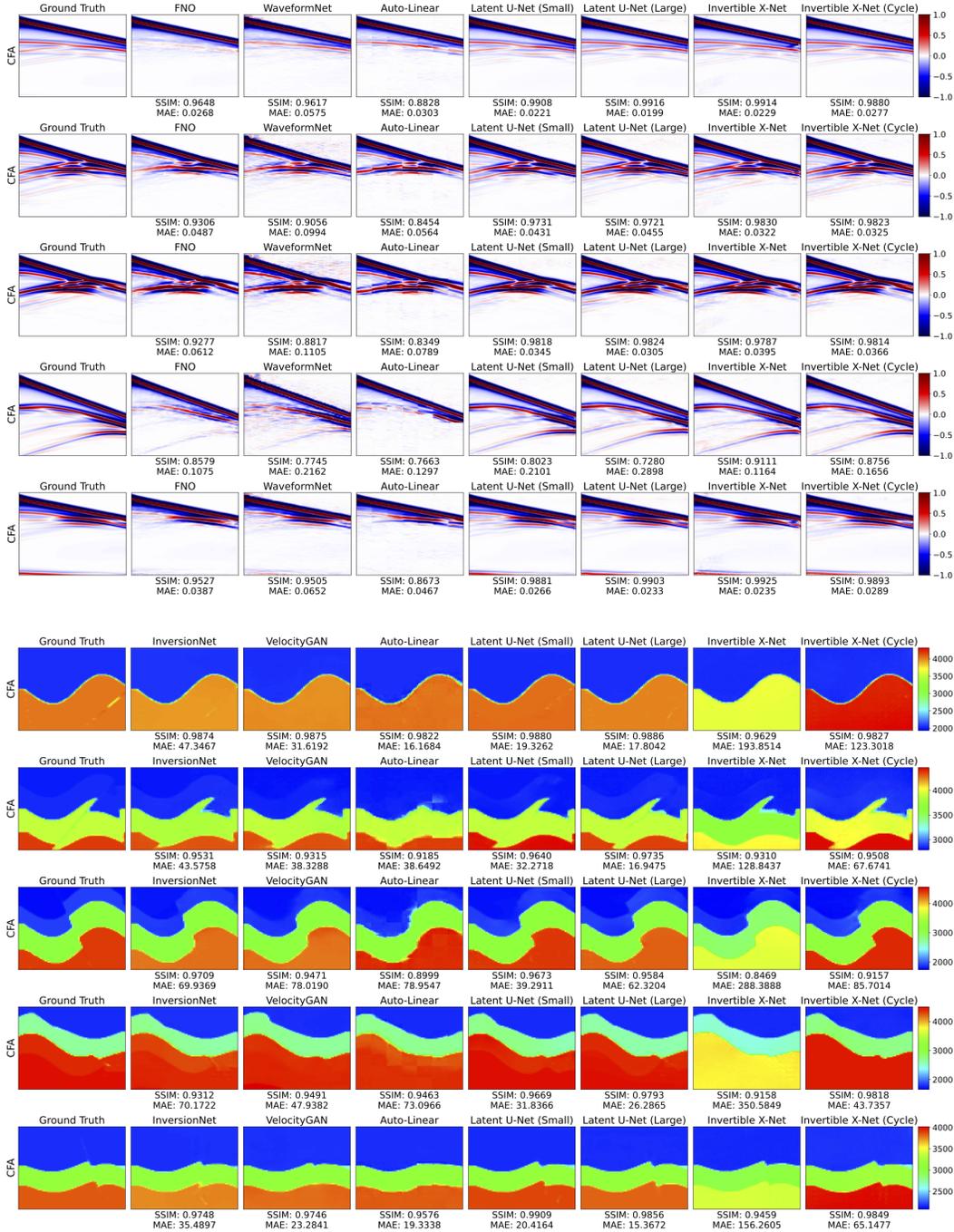


Figure 31: Visualization of predictions for forward and inverse problems on CFA dataset.

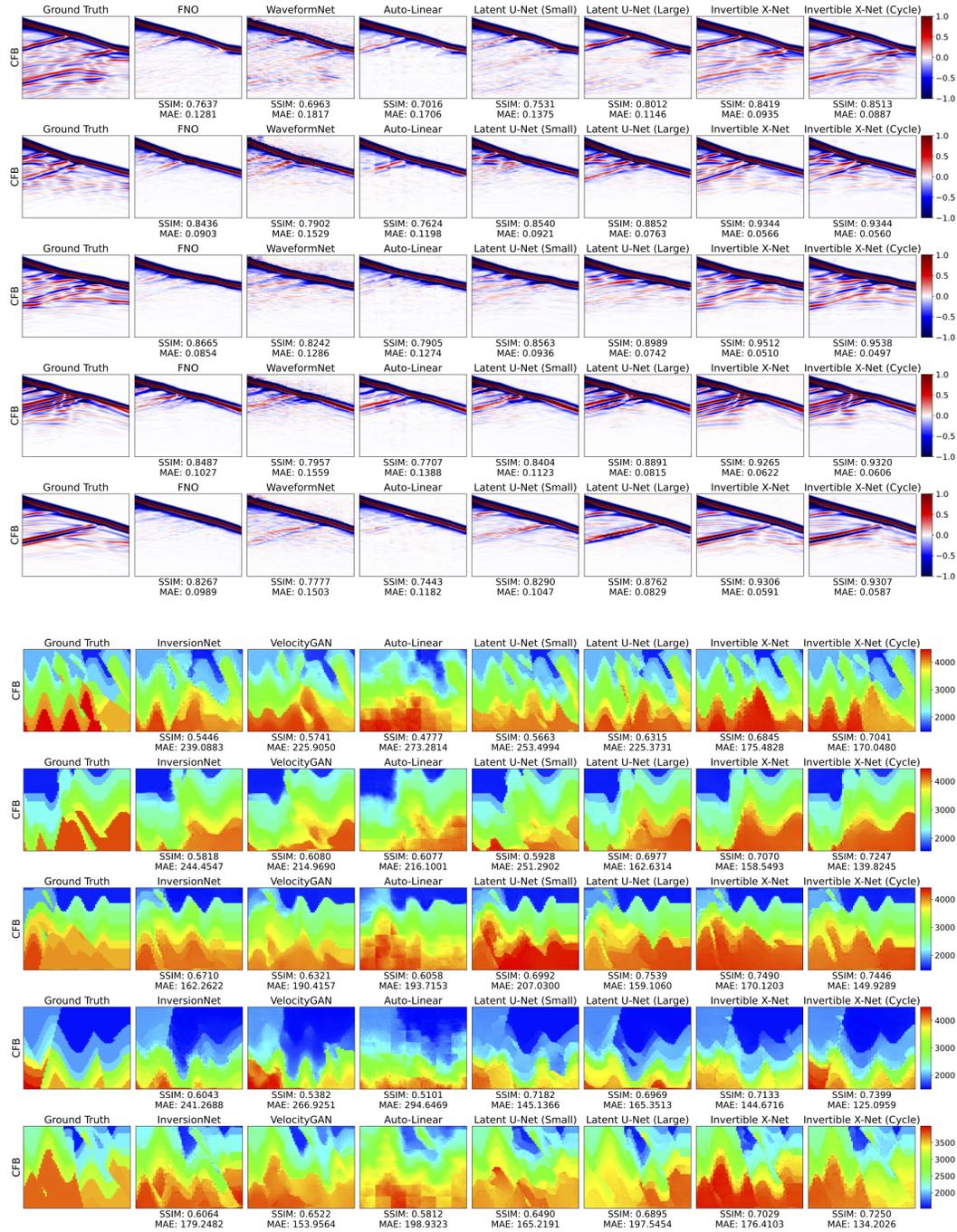


Figure 32: Visualization of predictions for forward and inverse problems on CFB dataset.

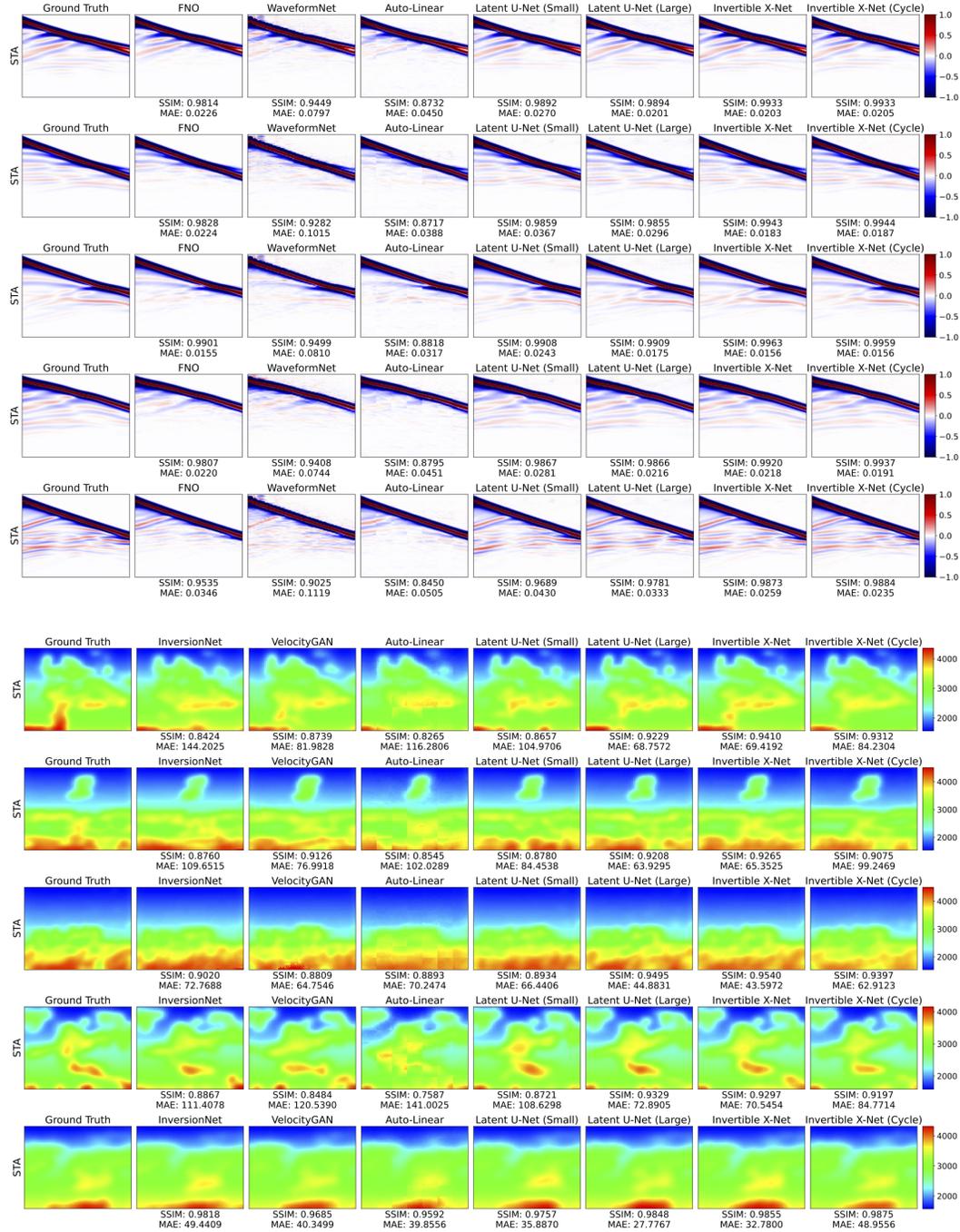


Figure 33: Visualization of predictions for forward and inverse problems on STA dataset.

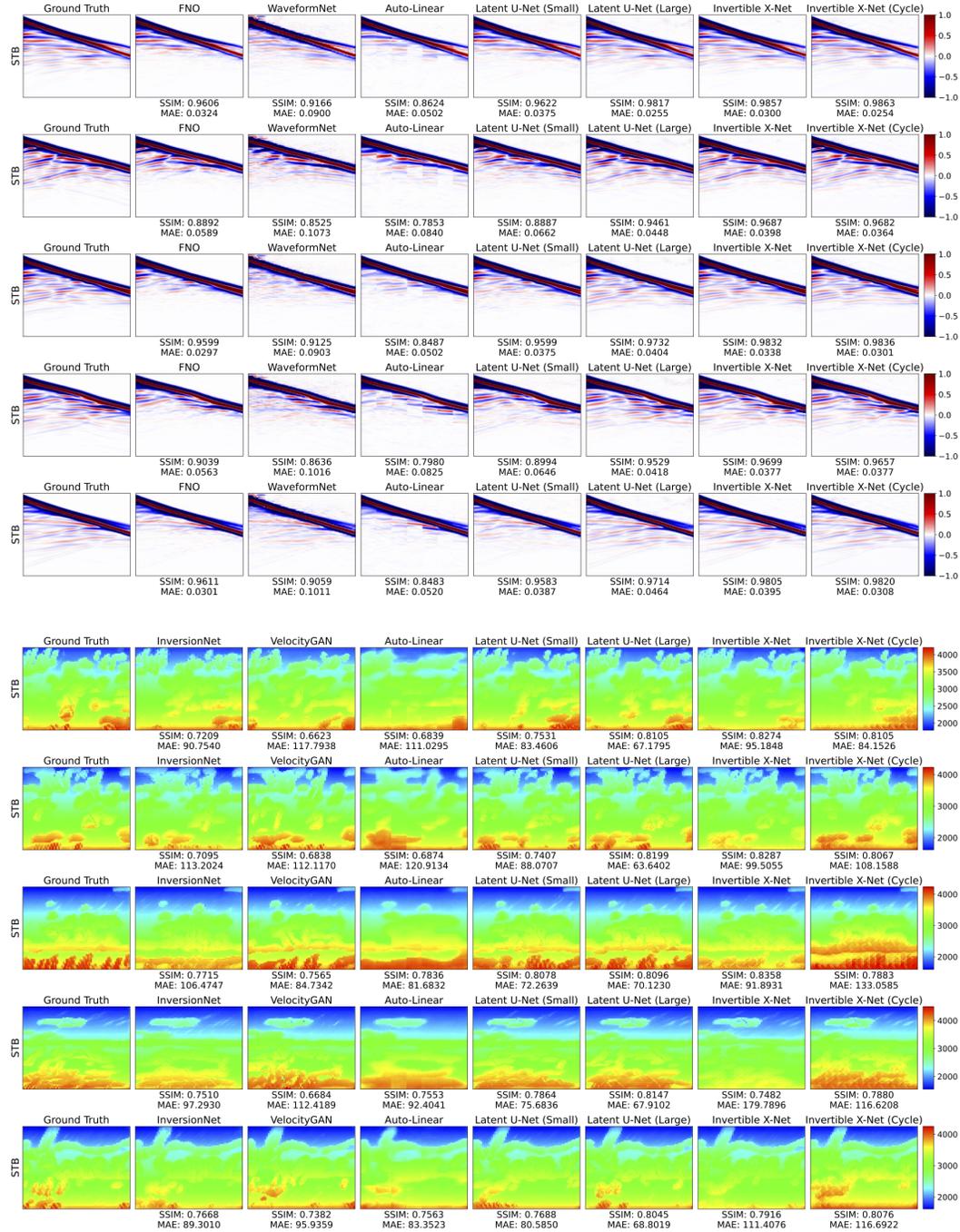


Figure 34: Visualization of predictions for forward and inverse problems on STB dataset.