MLP-Mixer based surrogate model for seismic ground motion with spatial source and geometry parameters

Hirotaka Hachiya HHACHIYA@WAKAYAMA-U.AC.JP
Graduate School of Systems Engineering, Wakayama University/Center for AIP, RIKEN

Yuto Kuroki KUROKI.YUTO@G.WAKAYAMA-U.JP
Graduate School of Systems Engineering, Wakayama University

Asako Iwaki IWAKI@BOSAI.GO.JP
National Research Institute for Earth Science and Disaster Resilience

Takahiro Maeda TMAEDA@BOSAI.GO.JP
National Research Institute for Earth Science and Disaster Resilience

Naonori Ueda NAONORI.UEDA@RIKEN.JP
Center for AIP, RIKEN

Hiroyuki Fujiwara FUJIWARA@BOSAI.GO.JP
National Research Institute for Earth Science and Disaster Resilience

Editors: Vu Nguyen and Hsuan-Tien Lin

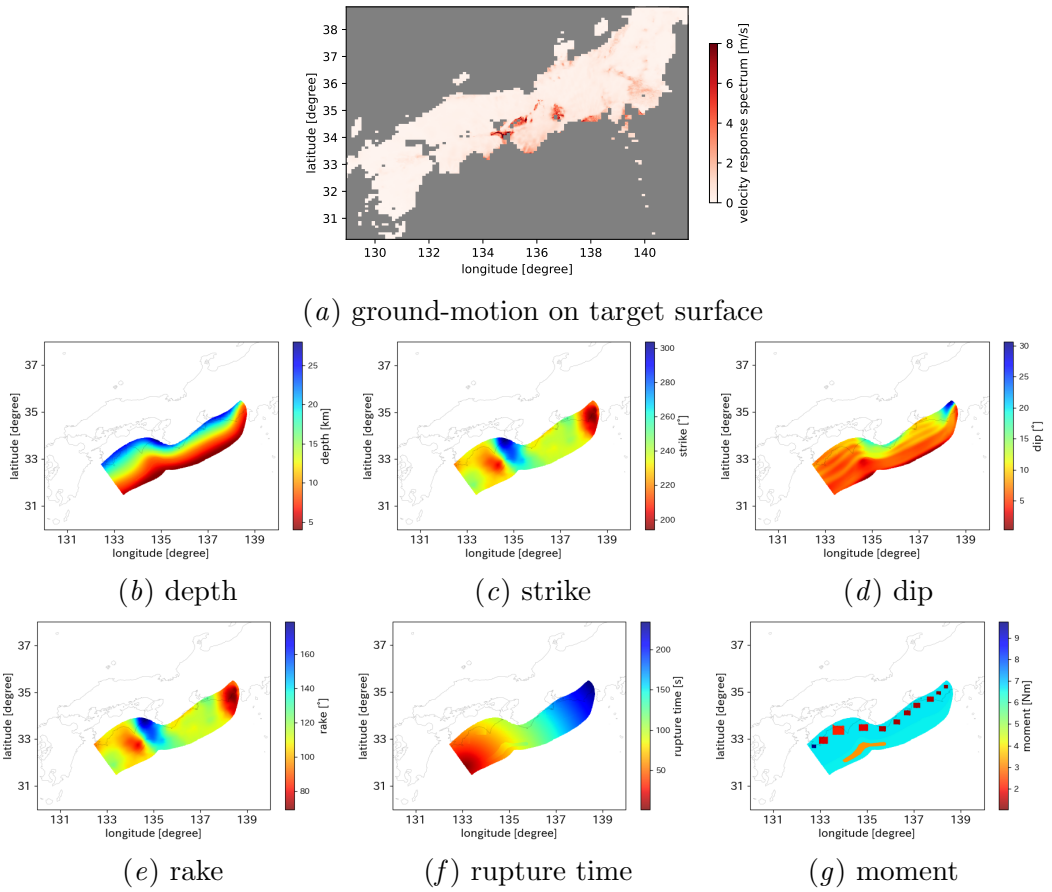
Abstract

Seismic motion simulations enable high-precision predictions, but are computationally demanding. This study introduces a deep learning surrogate model using the MLP-Mixer architecture to address this challenge. Traditional models using independent Multi-layer Perceptrons (MLPs) fail to capture spatial correlations, while U-shaped Neural Operators (U-NOs) require high computational costs for high-resolution inputs and outputs. Our proposed model, the Multiple MLP-Mixer (Multi-MLP-Mixer), integrates global and local spatial information through multiple MLP-Mixer blocks and dual patch-wise affine transformations. We demonstrate the effectiveness of our method with simulation data from anticipated megathrust earthquakes in the Nankai Trough, achieving performance comparable to state-of-the-art models with significantly improved computational efficiency.

Keywords: surrogate model; MLP-Mixer; seismic ground motion

1. Introduction

The recent years have seen numerous signs of a megathrust earthquake in Japan. Such earthquakes are likely to cause severe damage rapidly across a wide area of Japan. Simulating ground motion for anticipated rupture scenarios, predicting damages to buildings and infrastructure, and creating appropriate hazard maps could substantially reduce potential damage. For instance, ground-motion data for Nankai Trough megathrust earthquakes in southwest Japan can be generated using numerical finite-difference computation based on the theory of elastic wave propagation from spatially distributed point sources to the target surface. This considers various rupture scenarios such as source magnitude, asperity patterns, and rupture initiation points (Maeda et al., 2016; Moschetti et al., 2017). Fig. 1 depicts an example of simulated 5% damped velocity response spectra [m/s] with a period of 5 seconds for a megathrust earthquake occurring in the Nankai Trough, given spatially



(*a*) ground-motion on target surface

(*b*) depth (*c*) strike (*d*) dip

(*e*) rake (*f*) rupture time (*g*) moment

Figure 1: An example of simulated 5% damped velocity response spectra [m/s] with a period of 5 seconds for Nankai Trough megathrust earthquake (a), given spatially distributed point sources, each point of which includes depth (b), orientation: strike (c), dip (d), and rake (e), rupture time (f), and moment (g) parameters. The value of the response spectra is clipped at 8.0 for visualization purposes.

distributed point sources, including position (depth) and orientation (strike, dip, and rake), rupture time, and moment parameters.

However, simulations require enormous time and computational cost, making it challenging to validate various scenarios quickly. In addition, implementing these simulations can be complex, often requiring specialized knowledge and expertise in numerical modeling and computational seismology. To address this issue, the construction of surrogate models using deep models for ground motion simulations has been proposed in recent years (Lehmann et al., 2023b,a; Yang et al., 2023; Lehmann et al., 2024).

More specifically, the study by Lehmann et al. (2023b) introduced a single multi-layer perceptron (MLP) that individually predicts the ground motion value at each target point based on a combination of source and target geometry parameters. Additionally, the work by Lehmann et al. (2023a, 2024) employed a U-shaped Neural Operator (U-NO) which

processes the Fourier convolution with downsampled feature maps to convert 3D subsurface structures of target surface sites into time series of ground motion.

Existing surrogate methods for ground motion assume a single source point whose parameters are represented by a single feature vector. However, in large-scale spatiotemporal ground-motion simulations, such as those for the Nankai Trough, an earthquake source is modeled as the spatial and temporal evolution of rupture propagation on an earthquake fault. Therefore, hundreds of thousands of spatially distributed point sources are required (Maeda et al., 2016; Moschetti et al., 2017). Fig. 1 (b)-(g) depict examples of spatial point sources, each point of which includes position (depth), orientation (strike, dip, and rake), rupture time, and moment, used to generate ground motion values (a). For such high-resolution input and output, the UNO-based surrogate model (Lehmann et al., 2023a, 2024) requires significantly higher computational costs, specifically for the Fourier convolution in the shallow layers of encoder-decoder networks.

To handle high-resolution point sources in ground-motion surrogate models, in this study, we propose introducing the MLP-Mixer (Tolstikhin et al., 2021), which performs patch-embedding and alternates between a token (patch) mixer that integrates global spatial information and a channel mixer that integrates local channel information, where an MLP is shared at each mixer over tokens and channels. The proposed method transforms point source parameters into ground motion on the target surface based on global and local spatial correlation in high-resolution domains through flexible and computationally efficient multiple MLP-Mixer blocks. Additionally, we propose dynamically mixing multiple MLP-Mixers for integrating features of point source parameters with target geometry parameters, e.g., 3D subsurface structures, through dual patch-wise affine transformation. We demonstrate the effectiveness of the proposed method, called the Multiple MLP-Mixer (Multi-MLP-Mixer) based surrogate model, using simulation data of anticipated megathrust earthquakes in the Nankai Trough.

The main contributions of this paper are summarized as follows:

1. We propose a new deep surrogate model enabling fine-grained and computationally efficient transformation from spatial source and target geometry parameters into seismic ground motion through multiple MLP-Mixer blocks and dual patch-wise affine transformation in high-resolution domains.

2. We conduct extensive comparative experiments on predicting ground motion in the Nankai Trough, demonstrating that the proposed method provides performance comparable to state-of-the-art surrogate models with significantly more efficient training and inference computation.

After this introductory section, the rest of this paper is organized as follows. Section 2 reviews related works, and section 3 details the proposed method. Section 4 describes the experimental evaluation and discussion, and the conclusion is presented in Section 5.

2. Related works

2.1. Formulation

Let us consider training $\mathcal{D}^{\mathrm{tr}}$ and test $\mathcal{D}^{\mathrm{te}}$ data consisting of multiple tuples of target surface values $Y \in \mathbb{R}^{H^{\mathrm{Y}} \times W^{\mathrm{Y}} \times C^{\mathrm{Y}}}$, point source parameters $X \in \mathbb{R}^{H^{\mathrm{X}} \times W^{\mathrm{X}} \times C^{\mathrm{X}}}$, and target geometry

parameters $Z \in \mathbb{R}^{H^{\mathrm{Z}} \times W^{\mathrm{Z}} \times C^{\mathrm{Z}}}$ as follows:

$$\mathcal{D}^{\mathrm{tr}} = \{(Y_e, X_e, Z_e)\}_{e=1}^{N^{\mathrm{tr}}}, \quad \mathcal{D}^{\mathrm{te}} = \{(Y_e, X_e, Z_e)\}_{e=1}^{N^{\mathrm{te}}}, \tag{1}$$

where H^*, W^*, and C^* are the height, width, and the number of channels in a tensor $*$. N^{tr} and N^{te} are the number of training and test samples, respectively.

Then, let us consider a surrogate model $f_{\boldsymbol{\theta}}(\cdot)$ to convert a point source parameter tensor X to a target surface tensor Y given target geometry parameter tensor Z as $\widehat{Y}_e = f_{\boldsymbol{\theta}}(X_e, Z_e)$ where $\boldsymbol{\theta}$ is a parameter of the model to be tuned so as to minimize loss function $\mathcal{L}(\cdot)$ with training data $\mathcal{D}^{\mathrm{tr}}$ as follows:

$$\min_{\boldsymbol{\theta}} \left[\frac{1}{N^{\mathrm{tr}}} \sum_{e=1}^{N^{\mathrm{tr}}} \mathcal{L}\big(f_{\boldsymbol{\theta}}(X_e, Z_e), Y_e, S\big) \right], \tag{2}$$

where $S \in \mathbb{R}^{H^{\mathrm{Y}} \times W^{\mathrm{Y}} \times C^{\mathrm{Y}}}$ is the target mask to exclude outer region, e.g., sea and lakes, from the training. S is defined as follows:

$$S[h, w, c] = \begin{cases} 0 & \text{if pixel } (h, w, c) \text{ is in the outside of evaluation-region} \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

Let $\bar{Y} \in \mathbb{R}^{(H^{\bar{\mathrm{Y}}} \times W^{\bar{\mathrm{Y}}}) \times d}$, $\bar{X} \in \mathbb{R}^{(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d}$, and $\bar{Z} \in \mathbb{R}^{(H^{\bar{\mathrm{Z}}} \times W^{\bar{\mathrm{Z}}}) \times d}$ represent the tensors of patches, where the tensors Y, X, and Z are split into small, non-overlapping patches in the 2D spatial direction and embedding each patch into an d-dimensional vector.

2.2. Deep model-based surrogate model for predicting ground motion

Recent advances in deep models have significantly enhanced the application of surrogate models for the simulation of ground motion. These models aim to replicate complex seismic phenomena accurately while reducing computational effort.

Lehmann et al. (2023b) utilizes a single MLP to individually predict the ground motion value at each target surface point in Y, such as peak ground acceleration, given a single point source parameter X (e.g., depth, magnitude, and orientation) and a single target geometry parameter Z (e.g., hypocentral distance and azimuth). The input and output of the MLP are a concatenated vector of X and Z, and a scalar value, respectively. Although employing a single and individual MLP provides efficient memory usage, it cannot directly treat spatially distributed point sources X nor account for the correlation between target surface values Y.

Lehmann et al. (2023a) utilize a Fourier Neural Operator (FNO) (Li et al., 2021) to predict 3D seismic wave propagation Y on the target surface, given target 3D geological parameters Z, i.e., subsurface structures. The convolution in the Fourier space performed by the FNO corresponds to a global convolution in the feature space; therefore, it is effective for handling 3D geological parameters Z that contain various frequency components and global spatial correlations. In addition, a neural operator with a U-shaped encoder-decoder structure (U-NO) is employed for treating high-resolution input and output with efficient computation by processing the Fourier convolution with downsampled feature maps. Furthermore, Lehmann et al. (2024) extend the U-NO to incorporate multiple inputs, not only

3D geological parameters Z but also a variable point source parameter X, such as position and orientation.

However, even with U-NO, performing Fourier convolution with a relatively large kernel window for high-resolution feature maps needs to be done in the shallow layers, i.e., in the early encoder and late decoder layers. This results in a significantly larger number of model parameters and higher computational costs compared to the MLP-based surrogate model (Lehmann et al., 2023b), making the model more complex and resource-intensive. In addition, these existing SOTA surrogate models (Lehmann et al., 2023b,a, 2024) assume that the point source parameter X is fixed or set at a specific single point and, therefore, is represented in a single feature vector. However, a large-scale spatiotemporal ground motion simulation, such as the Nankai Trough (see Fig. 1), requires hundreds of thousands of spatially distributed point sources (Maeda et al., 2016; Moschetti et al., 2017).

2.3. Spatial feature extraction from images

To address the challenges of extracting features that consider global spatial correlations, recent methodologies have introduced advanced architectures such as Convolutional Vision Transformer (ConViT) (d'Ascoli et al., 2021), Adaptive Fourier Neural Operator (AFNO) (Guibas et al., 2022), and MLP-Mixer (Tolstikhin et al., 2021).

Vision Transformer (ViT) (Dosovitskiy et al., 2021), which applies self-attention to patch embeddings, outperforms convolutional neural networks (CNNs) but suffers from a low inductive bias, requiring pre-training on large external datasets. To overcome this problem, ConViT (d'Ascoli et al., 2021) leverages local feature extraction by CNNs and global spatial correlations by ViT through Gated Positional Self-Attention (GPSA) layers. The GPSA layers use trainable embeddings and relative position encodings to initially mimic the convolutional operation and incorporate gating parameters to balance attention and pseudo-convolution. Consequently, ConViT achieves high accuracy with limited data by enhancing ViT's learning efficiency.

Traditional self-attention mechanisms in ViT scale quadratically with the number of patches (tokens), creating challenges for high-resolution inputs. AFNO (Guibas et al., 2022) addresses this by replacing token mixing with the Discrete Fourier Transform (DFT) applied to each channel in the token direction, which corresponds to efficient global convolution, thus significantly reducing the computational complexity of extracting global spatial correlations. After channel mixing is performed using a shared MLP in the Fourier space, an inverse DFT is applied to convert them back to the feature space. Additionally, AFNO incorporates adaptive weight sharing in the channel mixing via a three-layer MLP and introduces sparsity through soft-thresholding in the Fourier domain to retain significant frequency components while discarding noise. These modifications result in quasi-linear complexity and linear memory requirements relative to the number of tokens, enabling efficient handling of high-resolution images.

MLP-Mixer (Tolstikhin et al., 2021) offers an alternative to CNN and ViT by using separate MLPs shared for token-wise mixing and channel-wise mixing of features. MLP-Mixer captures global spatial correlations without the complexity of convolutional or attention mechanisms by treating token and channel dimensions separately and iteratively mixing in-

formation. This simplicity allows MLP-Mixer to handle high-resolution inputs, i.e., a large number of tokens, with lower computational overhead compared to more complex models.

Given these advances, we propose developing a surrogate model utilizing the MLP-Mixer architecture. This model aims to efficiently capture global spatial correlations in both point source parameters X and target geometry parameters Z to predict large-scale spatiotemporal ground motion Y, thus enhancing prediction accuracy while maintaining computational efficiency.

2.4. Details of MLP-Mixer

The MLP-Mixer consists of a patch-embedding layer and multiple mixer layers. The patch-embedding layer divides the input image $X \in \mathbb{R}^{H^{\mathrm{X}} \times W^{\mathrm{X}} \times C^{\mathrm{X}}}$ and embeds each patch into a d-dimensional vector using Conv2D and reshapes to a matrix $\bar{X} \in \mathbb{R}^{(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d}$.

The mixer layer has two MLPs: token mixer and channel mixer. The token mixer applies a shared three-layer MLP along the token direction with $H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}$-dimension as follows:

$$\bar{U} = \mathrm{tokenMixer}(\bar{X}) = \bar{X} + \left\{ \sigma \left(\mathrm{LayerNorm}(\bar{X})^{\top} W^{\mathrm{tkn1}} \right) W^{\mathrm{tkn2}} \right\}^{\top} \in \mathbb{R}^{(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d}, \quad (4)$$

where $W^{\mathrm{tkn1}} \in \mathbb{R}^{(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d^{\mathrm{h}}}$ and $W^{\mathrm{tkn2}} \in \mathbb{R}^{d^{\mathrm{h}} \times (H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}})}$ are trainable weights, d^{h} is the dimension of the hidden layer, $\sigma(\cdot)$ is a activation function, e.g., GELU, and $\mathrm{LayerNorm}(\cdot)$ is a layer normalization.

Similarly, the channel mixer applies a shared three-layer MLP along the channel direction with $d0$dimension as follows:

$$\bar{X}' = \mathrm{channelMixer}(\bar{U}) = \bar{U} + \sigma \left(\mathrm{LayerNorm}(\bar{U}) W^{\mathrm{chnl1}} \right) W^{\mathrm{chnl2}} \in \mathbb{R}^{(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d} \quad (5)$$

where $W^{\mathrm{chnl1}} \in \mathbb{R}^{d \times d^{\mathrm{h}'}}$ and $W^{\mathrm{chnl2}} \in \mathbb{R}^{d^{\mathrm{h}'} \times d}$ are trainable weights, and $d^{\mathrm{h}'}$ is the dimension of the hidden layer.

By alternatively mixing features across two directions, the MLP-Mixer captures global and local spatial correlations over images, achieving high accuracy in image classification tasks with lower computational costs.

3. Proposed method

In this study, we propose a new deep surrogate model based on multiple MLP-Mixer streams, enabling to integrate spatiallydistributed point source parameters X and 3D target geometry parameters Z, to dynamically predict target surface values Y as depicted in Fig. 2.

3.1. Up-sampling and patch-embedding

To match the number of patches between $\bar{X}$ and $\bar{Z}$, and to increase the number of patches to mitigate discontinuities between patches, the resolution of X and Z increases through the up-sampling layer such as bilinear interpolation. Then, the patch-embedding layer divides the high-resolution images, embeds each patch into an d-dimensional vector using

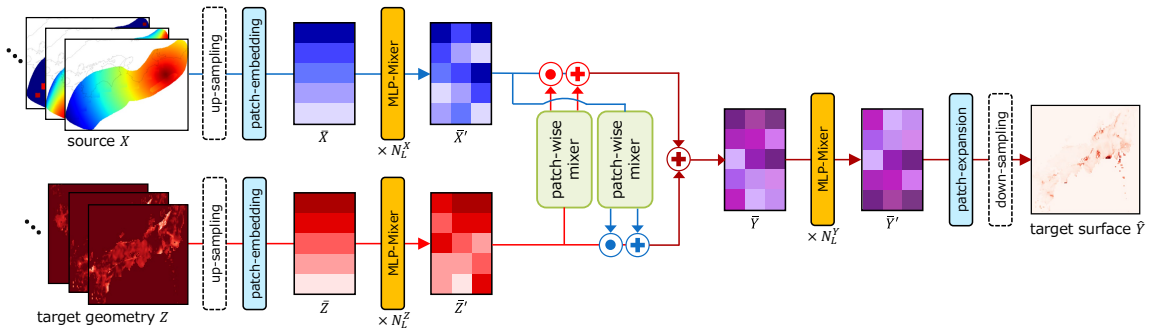


Figure 2: Architecture of the proposed surrogate model, Multi-MLP-Mixer, consisting of patch-embedding, up-sampling, three MLP-Mixer blocks, patch-wise affine transform mixer, patch-expansion, and down-sampling to predict target surface values $\widehat{Y}$ by converting point source parameters X and target geometry parameter Z.

2D convolution and reshapes to matrices as follows:

$$\bar{X} = \mathrm{reshape}\Big(\mathrm{patchEmbed}_{\boldsymbol{\alpha}^{\bar{X}}}\big(\mathrm{upsample}(X)\big)\Big) \in \mathbb{R}^{(H^{\bar{\mathrm{X}}}\times W^{\bar{\mathrm{X}}})\times d}, \tag{6}$$

$$\bar{Z} = \mathrm{reshape}\Big(\mathrm{patchEmbed}_{\boldsymbol{\alpha}^{\bar{Z}}}\big(\mathrm{upsample}(Z)\big)\Big) \in \mathbb{R}^{(H^{\bar{\mathrm{X}}}\times W^{\bar{\mathrm{X}}})\times d}, \tag{7}$$

where $\boldsymbol{\alpha}$ is a parameter in the 2D convolution.

3.2. Patch-wise affine-transform mixer of two-stream MLP-Mixers

Inspired by Feature-wise Linear Modulation (FiLM) (Perez et al., 2018), a general-purpose conditioning method for neural networks, we propose fusing two streams of MLP-Mixers through dual patch-wise affine transformations. Specifically, we apply affine transformations to $\bar{X}$ conditioned on $\bar{Z}$ and to $\bar{Z}$ conditioned on $\bar{X}$. More concretely, using scale function $\gamma(\cdot)$ and shift function $\beta(\cdot)$, $\bar{X}$ and $\bar{Z}$ are fused as follows:

$$\begin{aligned}\bar{Y} &= \frac{1}{2}\big\{\gamma(\bar{Z})\odot\bar{X} + \gamma(\bar{X})\odot\bar{Z} + \beta(\bar{Z}) + \beta(\bar{X})\big\},\\ \gamma(\bar{Z}) &= \bar{Z}W_Z^{\gamma} + \boldsymbol{b}_Z^{\gamma},\ \gamma(\bar{X}) = \bar{X}W_X^{\gamma} + \boldsymbol{b}_X^{\gamma}, \beta(\bar{Z}) = \bar{Z}W_Z^{\beta} + \boldsymbol{b}_Z^{\beta},\ \beta(\bar{X}) = \bar{X}W_X^{\beta} + \boldsymbol{b}_X^{\beta}\end{aligned} \tag{8}$$

where $W^{\gamma} \in \mathbb{R}^{d\times d}$, $W^{\beta} \in \mathbb{R}^{d\times d}$, $\boldsymbol{b}^{\gamma} \in \mathbb{R}^{1\times d}$, and $\boldsymbol{b}^{\beta} \in \mathbb{R}^{1\times d}$ are trainable weights and biases for scale and shift values, respectively.

3.3. Patch expansion and down-sampling

In the patch-expansion layer, after reshaping the patch tensor $\bar{Y}$ from $(H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}}) \times d$ to $H^{\bar{\mathrm{X}}} \times W^{\bar{\mathrm{X}}} \times d$, $\bar{Y}$ is converted into a high-resolution image through transposed convolution operations. Then, to smooth out the discontinuities between patches, down-sampling is

performed using methods such as bilinear interpolation to obtain the predicted target surface values $\widehat{Y}$ as follows:

$$\widehat{Y} = \text{downsample}\Big(\text{patchExpansion}_{\boldsymbol{\beta}}\big(\text{reshape}(\bar{Y}')\big)\Big) \in \mathbb{R}^{H^{\mathrm{Y}} \times W^{\mathrm{Y}} \times C^{\mathrm{Y}}}, \tag{9}$$

where $\boldsymbol{\beta}$ is a parameter in 2D convolution in patch-embedding functions.

3.4. Entire architecture and training

The architecture of the proposed method, called Multi-MLP-Mixer, is illustrated in Fig. 2, consisting of up-sampling, patch-encoding (in Sec. 3.1), three MLP-Mixer blocks each of which consists of N_L^*-layer MLP-Mixers (in Sec. 2.4), patch-wise affine transformation mixer (in Sec. 3.2), patch-expansion, and down-sampling (in Sec. 3.3).

Thus, the surrogate function $f(\cdot)$ in the architecture is represented by a composite function of tokenMixer$(\cdot)$ (Eq. 4), channelMixer$(\cdot)$ (Eq. 5), patchEmbed$(\cdot)$ (Eq. 7), $\gamma(\cdot)$, $\beta(\cdot)$ (Eq. 8), and patchExpansion$(\cdot)$ (Eq. 9), etc. Regarding the implementation, based on the official codes of MLP-Mixer (https://github.com/google-research/vision_transformer), we extended it with three MLP-Mixer blocks, a patch-wise affine transformation mixer, and replaced the head network for the classification task with patch-expansion and down-sampling (in Sec. 3.3), as depicted in Fig. 2.

Parameters $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \{W^{\text{tkn1}}\}, \{W^{\text{tkn2}}\}, \{W^{\text{chnl1}}\}, \{W^{\text{chnl2}}\}, \{W^{\gamma}\}, \{W^{\beta}\}, \{\boldsymbol{b}^{\gamma}\}, \{\boldsymbol{b}^{\beta}\}, \boldsymbol{\beta})$ are tuned so as to minimize the loss function (in Eq. 2), defined as $\mathcal{L}\big(\widehat{Y}_e, Y_e, S\big) = \mathcal{L}_{\text{SSIM}}(\widehat{Y}_e, Y_e, S)$ where $\mathcal{L}_{\text{SSIM}}$ is a masked version of Structural Similarity (SSIM) loss (Bergmann et al., 2018; Wang et al., 2004).

The SSIM is a metric used to evaluate the similarity between images by considering structural information, luminance, and contrast. The masked SSIM index between two patches P and Q with mask M is calculated using masked means μ_*, variance σ_{**}, and covariance σ_{PQ} where $* \in \{P, Q\}$ as follows:

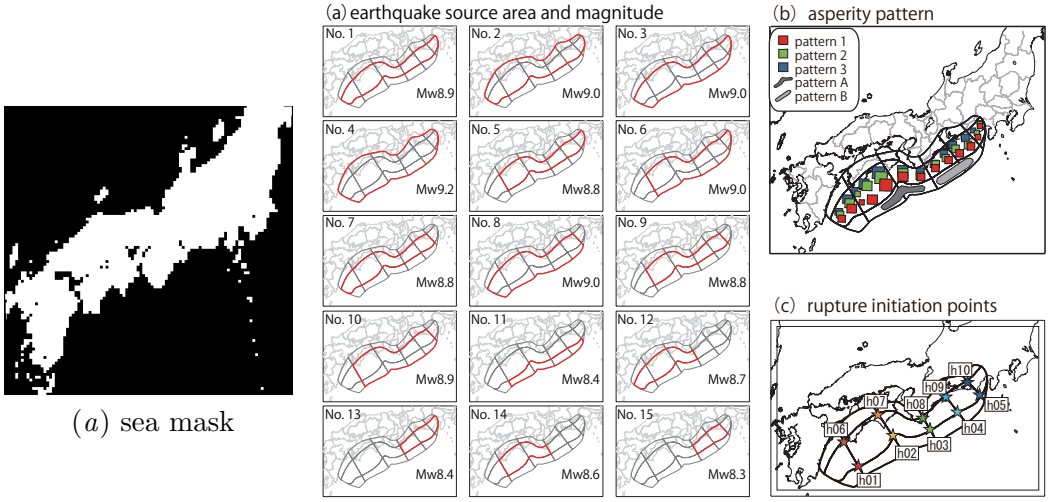
$$\text{SSIM}(P, Q, M) = \frac{(2\mu_P\mu_Q + C_1)(2\sigma_{PQ} + C_2)}{(\mu_P{}^2 + \mu_Q{}^2 + C_1)(\sigma_{PP} + \sigma_{QQ} + C_2)}, \tag{10}$$

$$\mu_* = \begin{cases} \frac{\text{sum}(*\odot M)}{\text{sum}(M)} & \text{if sum}(M) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad \sigma_{PQ} = \begin{cases} \frac{\text{sum}\big((P-\mu_P)(Q-\mu_Q)\odot M\big)}{\text{sum}(M)} & \text{if sum}(M) > 0 \\ 0 & \text{otherwise,} \end{cases} \tag{11}$$

where sum$(*)$ is the sum of all elements in a matrix $*$, $C_1 = 0.01^2$ and $C_2 = 0.03^2$ are constants to avoid the zero division. Then, the masked SSIM loss is obtained by the average of masked SSIM indices of corresponding patches of original Y and reconstructed $\widehat{Y}$ images as follows:

$$\mathcal{L}_{\text{SSIM}}(Y, \widehat{Y}) = \frac{1}{\|\mathcal{P}\|} \sum_{p \in \mathcal{P}} \big\{1 - \text{SSIM}(Y_p, \widehat{Y}_p, S_p)\big\}, \tag{12}$$

where Y_p, $\widehat{Y}_p$, and S_p represent p-th patch in each corresponding image Y, $\widehat{Y}$, and S, split by sliding windows with a stride of 1 and window size of 11, and $\mathcal{P}$ is the set of patch indices.



(*a*) sea mask

(*b*) earthquake rupture parameter

Figure 3: (a) Mask for excluding sea area defined in Eq. 3. (b) Variability of the possible earthquake rupture parameters for the anticipated megathrust earthquakes in the Nankai Trough (Maeda et al., 2016). (b-a) 15 patterns of the earthquake source area, denoted by red lines, and the magnitude of the earthquake labeled Mw. (b-b) Spatial patterns of the location of the asperities. (b-c) 10 patterns of the location of the rupture initiation points, denoted by stars.

Table 1: List of S-wave velocities v_s [km/s]

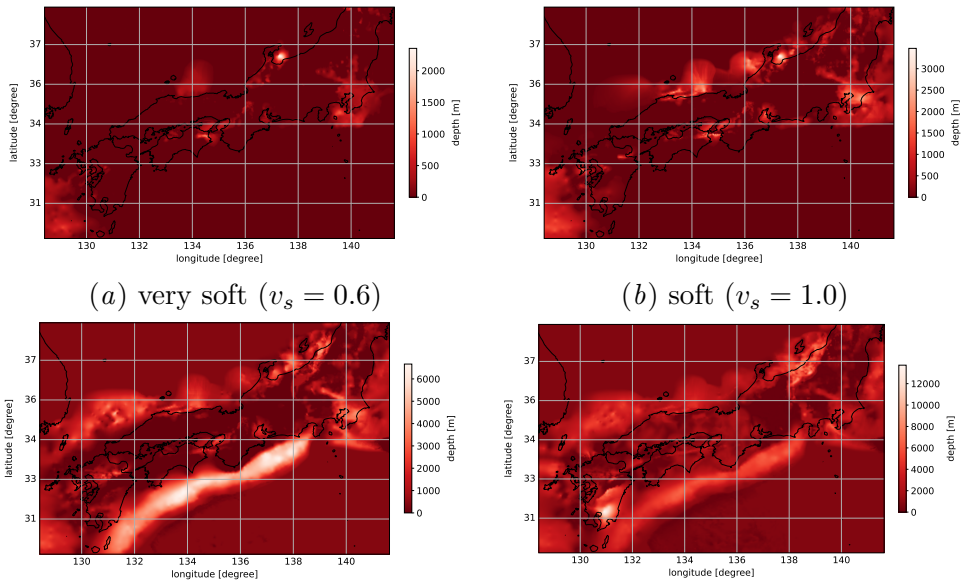
Layer	1	2	3	4	5	6	7	8	9	10	11	12	13
S-wave velocity	0.5	0.6	0.7	0.8	0.9	1.0	1.3	1.5	1.7	2.0	2.4	2.9	3.2

4. Experimental evaluation

To demonstrate the effectiveness of the proposed method, Multi-MLP-Mixer, we compared the performance of the prediction of ground-motion Y with SOTAs of global spatial feature extraction methods applied to surrogate models, such as ConViT (d'Ascoli et al., 2021) and AFNO (Guibas et al., 2022), and the SOTAs of surrogate models for ground motion, based on a single MLP (Lehmann et al., 2023b) and U-NO (Lehmann et al., 2023a) using large-scale ground-motion simulation data,

4.1. Nankai Trough simulated ground-motion data

As a large-scale spatiotemporal ground-motion simulation, we used Nankai Trough simulation dataset, provided by Hachiya et al. (2023); Maeda et al. (2016). The dataset consists of tuples of ground motion values Y on the target surface, spatially distributed point source parameters X, and 3D subsurface structure (target geometry parameters) Z with 360 rupture scenarios for the anticipated megathrust earthquakes in the Nankai Trough, combining possible rupture parameters; (a) the source area and magnitude, (b) the spatial pattern of



(a) very soft ($v_s = 0.6$) (b) soft ($v_s = 1.0$)

(c) hard ($v_s = 1.7$) (d) very hard ($v_s = 3.2$)

Figure 4: Examples of 3D subsurface structure Z consisting of the top surface depth of layers with S-wave velocities v_s: 0.6 (very soft), 1.0 (soft), 1.7 (hard), and 3.2 (very hard) [km/s].

Table 2: Data specifications for Nankai Trough ground motion simulation

	content of each channel	mesh shape	geographic coordinate
target surface Y	The ground motion represented by 5% damped velocity response spectra [m/s] with a period of 5 seconds. The range is clipped to $[0, 8]$ and normalized to $[0, 1]$ for experimental evaluation.	$(H^{\mathrm{Y}}, W^{\mathrm{Y}}, C^{\mathrm{Y}}) = (512, 512, 1)$	latitude $30.3° \sim 38.8°$ longitude $128.9° \sim 141.6°$ with intervals of 2km
source params X	The values of depth, strike, dip, rake, rupture time, and moment at each point.	$(H^{\mathrm{X}}, W^{\mathrm{X}}, C^{\mathrm{X}}) = (994, 1184, 6)$	latitude $30.3° \sim 38.8°$ longitude $128.9° \sim 141.6°$ with intervals of 0.25km
target geometry Z	The subsurface structure represented by top surface depth [m] of 13 layers in Table 1 at each target point. The value is normalized to $[0, 1]$ for experimental evaluation.	$(H^{\mathrm{Z}}, W^{\mathrm{Z}}, C^{\mathrm{Z}}) = (512, 512, 13)$	latitude $30.3° \sim 38.8°$ longitude $128.9° \sim 141.6°$ with intervals of 2km
mask S	The mask assigns 1 to land and 0 to sea as defined in Eq. 3 and described in Fig. 3(a).	$(H^{\mathrm{Y}}, W^{\mathrm{Y}}, C^{\mathrm{Y}})$	latitude $30.3° \sim 38.8°$ longitude $128.9° \sim 141.6°$ with intervals of 2km

the asperity locations, and (c) the location of the rupture initiation point, as depicted in Fig. 3(b).

Fig. 1 depicts an example of ground-motion Y which is simulated 5% damped velocity response spectra [m/s] with 5 seconds, given 6 different types of spatially distributed source parameters X where the earthquake rupture parameters (a), (b), and (c) in Fig. 3(b) are set to No. 6, the combination of pattern-3 and pattern-A, and h01.

Meanwhile, the 3D subsurface structure Z is represented by the top surface depth of layers with different S-wave velocities, ranging from 0.5 to 3.2 km/s—the higher S-wave velocities indicate a stiffer ground. Table 1 lists the S-wave velocities for each layer, assuming that the underground at each point consists of 13 layers. Furthermore, Fig. 4 presents top surface depth maps for layers with S-wave velocities of 0.6 (very soft), 1.0 (soft), 1.7 (hard) and 3.2 (very hard) km/s. Note that the subsurface structure is fixed for all 360 scenarios.

Table 2, lists the detailed information of ground-motion Y, source parameters X, subsurface structure Z, and mask S.

4.2. Settings

We performed $L = 10$-fold cross-validation where $\mathcal{D}^{\mathrm{tr}}$ were split into L-group $\{\mathcal{D}^l\}_{l=1}^{L}$ and a model was trained using $\{\mathcal{D}^l\}_{l\neq l'}$ and its performance was evaluated using $\mathcal{D}^{l'}$ for each $l' = 1, 2, \ldots, L$. Thus, in total, all data were used for the evaluation with L different models.

For the quantified evaluation, we used masked PSNR (Peak Signal-to-Noise Ratio) and SSIM. The masked PSNR is computed over test data $\mathcal{D}^{\mathrm{te}}$ across 10-fold CVs, as follows:

$$\mathrm{PSNR} = 10\log_{10}\frac{\mathrm{MAX}^2}{\mathrm{MAE}}, \quad \mathrm{MAE} \equiv \frac{1}{N^{\mathrm{te}}\mathrm{sum}(S)}\mathrm{sum}\Big(S \odot \big(Y_e - \widehat{Y}_e\big) \odot \big(Y_e - \widehat{Y}_e\big)\Big). \tag{13}$$

where N^{te} and MAX are the number of test data $\mathcal{D}^{\mathrm{te}}$ over 10 folds and the maximum possible error, respectively. Meanwhile, for masked SSIM, from each test data fold $\mathcal{D}^{\mathrm{te}}$, averaged masked SSIM (in Eq. 11) was calculated and averaged over 10 folds.

We compared the performance with SOTAs of global spatial feature extraction methods applied for surrogate models, such as ConViT (d'Ascoli et al., 2021) and AFNO (Guibas et al., 2022), and SOTAs of surrogate models for ground-motion, based on a single MLP (Lehmann et al., 2023b) and U-NO (Lehmann et al., 2023a) as follows:

- single MLP (Lehmann et al., 2023b) (in Sec.2.2): https://github.com/rauler95/gmacc. An MLP with eight hidden layers, each with 64 or 128 dimensions, was used to individually predict the value of the ground motion at each target surface point in Y. Following the paper (Lehmann et al., 2023b), a point source parameter $X \in \mathbb{R}^{1\times 6}$ was designed to include depth, strike, dip, rake, rupture time, and moment, and a target geometry $Z \in \mathbb{R}^{1\times 2}$ to include azimuth and hypocentral distance. The input and output of the MLP are a concatenated vector $\mathrm{concat}(X, Z) \in \mathbb{R}^{1\times 8}$ and a scalar value, respectively. We randomly selected points to train the model so that the number of points in $Y < 0.1$ and $Y \geq 0.1$ were equal.

- Multi-MLP-Mixer (proposed method) (in Sec. 3): The implementation is described in Sec. 3.4. In the up-sampling layers, the resolution of Z increases to 994×1184, which is the same as X (see Table 2). In the patch-embedding layers, X and Z are divided into 62×74 patches, and each patch is embedded into a d-dimensional vector—$\bar{Z}$, $\bar{X} \in \mathbb{R}^{(62\times 84)\times d}$ (in Sec. 3.1). As for the MLP-Mixer blocks, the numbers of MLP-Mixer layers in each block are $(N_L^X, N_L^Z, N_L^Y) = (4, 2, 4)$; the dimensions of the hidden layers in the token and channel mixers are $(d^{\mathrm{h}}, d^{\mathrm{h}'}) = (256, 1024)$; and the dimension of the embedding vector is $d \in \{64, 128\}$. In addition, batch size, learning rate, and dropout rate were set as 4, 0.001, and 0.1.

- MLP-Mixer (proposed method) (Tolstikhin et al., 2021): 8 MLP-Mixer layers with $(d^{\mathrm{h}}, d^{\mathrm{h}'}) = (256, 1024)$ were used to convert the source parameter X to the ground motion $\widehat{Y}$. This corresponds to the configuration of the proposed method (in Fig. 2) in which the stream for the target geometry Z and the patch-wise affine transformation mixer were removed. The dimension of the embedding vector is $d \in \{64, 128\}$.

Table 3: Performance and scale comparison in ground-motion prediction using metric masked PSNR, masked SSIM (in Sec. 4.2), parameter counts, training time, and inference time. The training and inference times are measured using an NVIDIA RTX6000 Ada. The best performance and smallest scale in each metric are indicated in bold in each group.

metric	dim	Spatial feature extraction				Surrogate model	
		AFNO	ConViT	MLP-Mixer (proposed)	Multi-MLP-Mixer (proposed)	single MLP	U-NO
PSNR	64	31.82	27.46	32.92	**33.46**	22.33	**33.51**
	128	32.58	27.48	33.33	**33.52**	-	**33.37**
SSIM	64	94.80	89.47	96.44	**96.77**	69.34	**96.80**
	128	95.39	88.65	**96.64**	**96.64**	-	**96.69**
parameter cnt. [M]	64	**1.23**	4.79	20.01	22.71	**0.03**	40.27
	128	**2.54**	15.87	21.17	24.25	-	161.08
training time [s]	64	2,055	2,191	679	**659**	**119**	1,456
	128	2,118	2966	796	**611**	-	1,989
inference time [ms]	64	1.63	26.59	**0.45**	0.84	**3.77**	4.78
	128	2.22	28.07	**0.48**	0.85	-	10.45

- AFNO (Guibas et al., 2022) (in Sec. 2.3): https://github.com/NVlabs/AFNO-transformer 8 AFNO layers with $d^{h'} = 1024$ were used to convert the source parameter X to the ground motion $\widehat{Y}$. Similarly to the proposed method, in the patch-embedding layer, X is embedded into $\bar{X} \in \mathbb{R}^{(62\times74)\times d}$, where the dimension of the embedding vector is $d \in \{64, 128\}$. The head network for classification was replaced with patch-expansion and down-sampling layers to produce the predicted ground-motion image $\widehat{Y}$.

- ConViT (d'Ascoli et al., 2021)(in Sec.2.3): https://github.com/facebookresearch/convit. 4 GPSA layers and 1 SA layer with 4 heads were used to convert the source parameter X to ground motion $\widehat{Y}$. Similarly to the proposed method, in the patch-embedding layer, X is embedded into $\bar{X} \in \mathbb{R}^{(62\times74)\times d}$, and the embedding vector was set to $d \in \{64, 128\}$. The head network for classification was replaced with patch-expansion and down-sampling layers to produce the predicted image $\widehat{Y}$.

- U-NO (Lehmann et al., 2023a) (in Sec. 2.2): https://github.com/ashiq24/uno. 8 U-NO layers (4 for the encoder and 4 for the decoder) with three skip connections were used to convert the source parameter X to the ground motion $\widehat{Y}$—note that the subsurface structure is fixed in this dataset and the method (Lehmann et al., 2023a) does not support two spatial inputs. The number of channels of the source parameters X increased from 6 to $d \in \{64, 128\}$ by a nonlinear lifting operation, while its resolution was reduced from $H^{\mathrm{X}} \times W^{\mathrm{X}} = 994 \times 1184$ to $H^{\mathrm{Y}} \times W^{\mathrm{Y}} = 512 \times 512$ by bicubic interpolation. In the encoder, the resolution of the feature map was reduced to $H^{\mathrm{Y}}/16 \times W^{\mathrm{Y}}/16 \times d/16$, and then in the decoder, it was expanded to $H^{\mathrm{Y}} \times W^{\mathrm{Y}} \times d$. Lastly, using an MLP, it was converted to the image of the predicted $\widehat{Y}$.

4.3. Quantitative evaluation

Table 3 presents a comparison of performance and scale using masked PSNR, masked SSIM (as described in Sec. 4.2), parameter counts, training time, and inference time. The training time for each fold was averaged over 10 folds, and the inference time for each test instance was averaged over all test data across 10 folds, using an NVIDIA RTX6000 Ada.

The table indicates that the single MLP (Lehmann et al., 2023b), which individually predicts each target value and therefore does not capture the spatial correlation between the target surface points, produces the lowest performance. In contrast, the state-of-the-art (SOTA) surrogate model, U-NO (Lehmann et al., 2023a), achieves the best performance utilizing Fourier convolution to capture various frequency components and global spatial correlations. However, the scale of the model is significantly larger than that of other methods. Although U-NO enhances computational efficiency by reducing resolution and the number of channels in the lower layers (late encoder and early decoder), it requires processing high-resolution and multi-channel features with FNO in the later stages of the decoder, resulting in a substantial increase in the number of trainable parameters.

Meanwhile, SOTA spatial feature extraction methods, such as AFNO (Guibas et al., 2022) and ConViT (d'Ascoli et al., 2021) provide performance comparable to U-NO. Specifically, AFNO achieves a significant reduction in parameter and inference time by discretizing the spatial domain, replacing the token mixer with a parameter-free DFT, and using a shared MLP in the frequency domain as a channel mixer. However, the Fourier transform extracts the global spatial correlation with sine and cosine waves along fixed periods and directions, which could introduce an inductive bias and lead to suboptimal performance. Although ConViT also achieves a significant parameter reduction, its training and inference time is the largest because self-attention requires quadratic computational complexity with respect to the number of tokens, i.e., $(62 \times 74)^2$.

Conversely, the proposed MLP-Mixer-based methods, by sharing a single MLP per layer in both spatial (token) and channel directions, apply transformations with low inductive bias in both spatial and channel directions. Although slightly higher in parameter count compared to AFNO, the proposed methods outperform in performance. Furthermore, the Multi-MLP-Mixer model, which integrates information extracted from two MLP-Mixer streams for source parameters $\bar{X}$ and subsurface structure $\bar{Z}$ using a patch-wise affine transformation mixer, further improves performance, making it well comparable to the SOTA surrogate model U-NO, while significantly enhancing computational efficiency in both training and inference. Furthermore, with $d = 128$, the proposed method improved performance without a significant reduction in computational efficiency.

4.4. Qualitative evaluation

Figs. 5 and 1 (in supplementary document) represent examples of the true image Y and the predicted image $\widehat{Y}$, and Figs. 6 and 2 (in supplementary document) represent true vs. predicted values. The figures show that given source parameters X (in Fig. 1) and subsurface structure Z (in Fig. 4), our proposed method, Multi-MLP-Mixer (c), can produce fine-grained predicted images $\widehat{Y}$ that look similar to the true ones Y (a) and have well comparable PSNR score with the SOTA surrogate model, U-NO.

In general, these experimental results indicate that the proposed MLP-Mixer-based methods could be effective solutions for the surrogate model of ground motion, considering spatially distributed point sources X and 3D subsurface structure Z. They provide a balanced combination of high performance and computational efficiency.

5. Conclusion

To overcome the computational challenges of seismic motion simulations, this study introduces the Multiple MLP-Mixer (Multi-MLP-Mixer) surrogate model. Traditional models, such as single MLP and U-NO, either fail to capture spatial correlations or require high computational costs for high-resolution inputs and outputs. Our proposed method leverages the MLP-Mixer architecture to efficiently integrate global and local spatial information through multiple MLP-Mixer blocks and dual patch-wise affine transformations.

We validated the effectiveness of our model using simulation data from anticipated megathrust earthquakes in the Nankai Trough. The results demonstrated that the Multi-MLP-Mixer achieved performance comparable to that of state-of-the-art surrogate models while significantly improving computational efficiency in both training and inference.

Overall, the Multi-MLP-Mixer offers a promising solution for seismic ground motion prediction, balancing high performance with computational efficiency.

Acknowledgments

This study was supported by the Grant-in-Aid for Scientific Research (A) (Kakenhi No. 23H00219) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

References

Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *Proceedings of International Conference on Computer Vision Theory and Applications (ICCVTA)*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021.

Stéphane d'Ascoli, Hugo Touvron, Matthew L. Leavitt, Ari S. Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 2286–2296. PMLR, 2021.

John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Efficient token mixing for transformers via adaptive fourier neural operators. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2022.

Hirotaka Hachiya, Kotaro Nagayoshi, Asako Iwaki, Takahiro Maeda, Naonori Ueda, and Hiroyuki Fujiwara. Position-dependent partial convolutions for supervised spatial interpolation. *Machine Learning with Application*, 14, 2023.

Fanny Lehmann, Filippo Gatti, Michaël Bertin, and Didier Clouteau. Fourier neural operator surrogate model to predict 3d seismic waves propagation. In *Proceedings of International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP)*, pages 1–12, 2023a. doi: 10.1109/UQ2023.00018.

Fanny Lehmann, Filippo Gatti, and Didier Clouteau. Multiple-input fourier neural operator (mifno) for source-dependent 3d elastodynamics. *arXiv preprint arXiv:2404.10115*, 2024.

Lukas Lehmann, Matthias Ohrnberger, Malte Metz, and Sebastian Heimann. Accelerating low-frequency ground motion simulation for finite fault sources using neural networks. *Geophysical Journal International*, 234:2328–2342, 2023b.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. 2021.

Takahiro Maeda, Asako Iwaki, Nobuyuki Morikawa, Shin Aoi, and Hiroyuki Fujiwara. Seismic-hazard analysis of long-period ground motion of megathrust earthquakes in the nankai trough based on 3d finite-difference simulation. *Seismological Research Letters*, 87 (6):1265–1273, 11 2016. doi: https://doi.org/10.1785/0220160093.

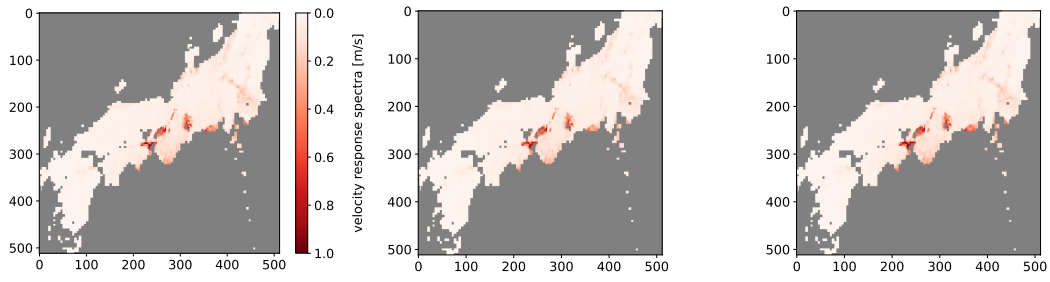
Morgan P. Moschetti, Stephen H. Hartzell, Leonardo Ramirez-Guzman, Arthur Frankel, Stephen J. Angster, and William J. Stephenson. 3d ground-motion simulations of mw 7 earthquakes on the salt lake city segment of the wasatch fault zone: Variability of long-period (t $\geq$ 1s) ground motions and sensitivity to kinematic rupture parameters. *Bulletin of the Seismological Society of America*, 107(64):1704–1723, 2017. doi: https://doi.org/10.1785/0120160307.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 32, 2018.

Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 24261–24272, 2021.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

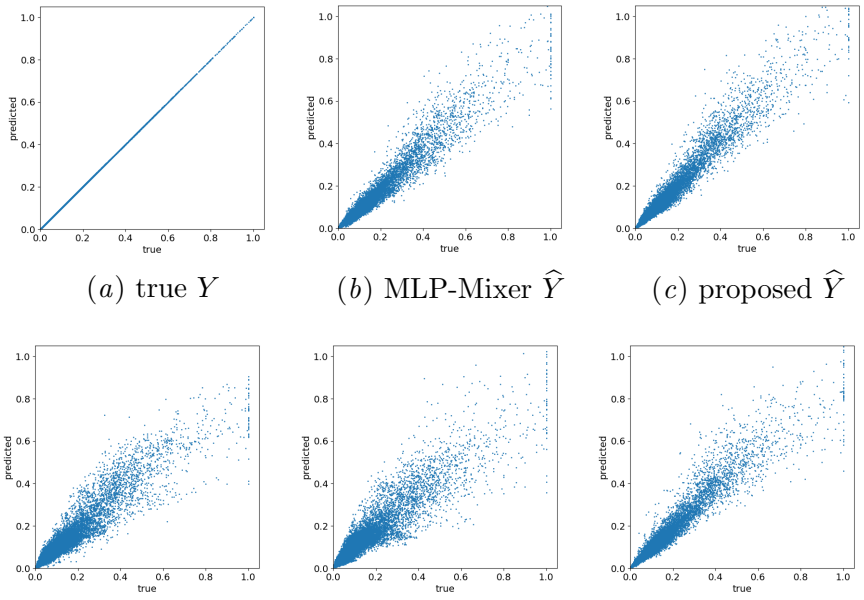
Xiaomei Yang, Miao Hu, Xin Chen, Shuai Teng, and Gongfa Chenand David Bassir. Predicting models for local sedimentary basin effect using a convolutional neural network. *Applied Sciences*, 13(2):300–312, 2023. doi: 10.3390/app13020300.



(a) true Y (b) MLP-Mixer (35.43) (c) proposed (35.89)

(d) AFNO (32.88) (e) ConViT (31.96) (f) U-NO (35.15)

Figure 5: Examples of true Y (a), and predicted $\widehat{Y}$ (b)-(f) ground-motion images given source parameters X in Fig 1 and subsurface structure Z in Fig. 4. The value in parentheses () indicates the PSNR score. Note that all images are masked by target-mask S for the visualization purpose.



(a) true Y (b) MLP-Mixer $\widehat{Y}$ (c) proposed $\widehat{Y}$

(d) AFNO $\widehat{Y}$ (e) ConViT $\widehat{Y}$ (f) U-NO $\widehat{Y}$

Figure 6: True vs. predicted values for Fig. 5.