Geophysical Journal International

Geophys. J. Int. (2024) **238,** 1353–1365
Advance Access publication 2024 June 14
GJI Seismology

https://doi.org/10.1093/gji/ggae212

Graph-learning approach to combine multiresolution seismic velocity models

Zheng Zhou,[1] Peter Gerstoft [1] and Kim Olsen [2]

[1] *Noise Lab, Scripps Institution of Oceanography, University of California San Diego, La Jolla, CA* 92093*, USA. E-mail: zhz039@ucsd.edu*
[2] *Department of Geological Sciences, San Diego State University, San Diego, CA* 92182*, USA*

Accepted 2024 June 6. Received 2024 June 5; in original form 2023 August 10

SUMMARY

The resolution of velocity models obtained by tomography varies due to multiple factors and variables, such as the inversion approach, ray coverage, data quality, etc. Combining velocity models with different resolutions can enable more accurate ground motion simulations. Toward this goal, we present a novel methodology to fuse multiresolution seismic velocity maps with probabilistic graphical models (PGMs). The PGMs provide segmentation results, corresponding to various velocity intervals, in seismic velocity models with different resolutions. Further, by considering physical information (such as ray path density), we introduce physics-informed probabilistic graphical models (PIPGMs). These models provide data-driven relations between subdomains with low (LR) and high (HR) resolutions. Transferring (segmented) distribution information from the HR regions enhances the details in the LR regions by solving a maximum likelihood problem with prior knowledge from HR models. When updating areas bordering HR and LR regions, a patch-scanning policy is adopted to consider local patterns and avoid sharp boundaries. To evaluate the efficacy of the proposed PGM fusion method, we tested the fusion approach on both a synthetic checkerboard model and a fault zone structure imaged from the 2019 Ridgecrest, CA, earthquake sequence. The Ridgecrest fault zone image consists of a shallow (top 1 km) high-resolution shear-wave velocity model obtained from ambient noise tomography, which is embedded into the coarser Statewide California Earthquake Center Community Velocity Model version S4.26-M01. The model efficacy is underscored by the deviation between observed and calculated traveltimes along the boundaries between HR and LR regions, 38 per cent less than obtained by conventional Gaussian interpolation. The proposed PGM fusion method can merge any gridded multiresolution velocity model, a valuable tool for computational seismology and ground motion estimation.

Key words: Image processing; Machine learning; Seismic tomography.

1 INTRODUCTION

The resolution of velocity models varies due to inversion approaches, ray coverage, etc. For example, large-scale community velocity models (CVMs) are typically characterized by relatively low resolution, while high-resolution velocity models are limited to smaller areas with dense station coverage. Combining such velocity models with different resolutions is useful to improve CVMs, for example for ground motion estimation or dynamic rupture modelling, where a range of scales is needed (e.g. Yeh & Olsen 2023). However, when models with different resolutions are directly superimposed, sharp boundaries caused by misaligned patterns may produce artefacts in the modelled seismic waves. The problem of fusing low- and high-resolution models is conventionally addressed by extracting patterns from various models and utilizing them as boundary conditions for wave propagation. Border merging has been performed by Gaussian smoothing (Wu & Janson 2017) and by defining a weighting region (Ajala & Persaud 2021). However, this process depends on experts manually choosing models, and fine-tuning parameters is often time-consuming.

The exploration of multiscale problems has surged across diverse geophysical fields, encompassing full wave inversion, ground motion modelling and surface-wave tomography. A pivotal discovery is the scale-dependent nature of anisotropy, a finding that has substantial implications for how we perceive the structure of the Earth (Van Houtte *et al.* 2006). Analyses from both physical and signal processing viewpoints bring to light key challenges associated with the effect of spatial heterogeneities on seismic wave propagation. These studies have shown that seismic waves cannot distinguish between large-scale anisotropy and small-scale isotropic heterogeneities much smaller than a wavelength (Barbarossa & Sardellitti 2020), enhanced by the potent spatial and directional dependence

of tomographic resolution. Such dependency may provoke space- and direction-oriented smoothing (Dhamo *et al.* 2020), eliciting apparent anisotropy fluctuations that bear no intrinsic connection to Earth's anisotropy.

The construction of multiscale models has traditionally used tactics such as multiscale element fitting (Fichtner *et al.* 2013), Gaussian kernel smoothing (Fang & Zhang 2014) and adaptive filtering methods, such as sparse learning (Zhu *et al.* 2015; Bianco & Gerstoft 2017, 2018). Zhang & Ben-Zion (2024) proposed a data-driven DL method for transformations between seismic velocity models of different resolutions, which involves a linear decomposition of an input signal using a small set of basis signals, or atoms, learned from HR and LR pairwise data. Although these methods have proved efficient, they might not fully grasp the vast complexity of the earth's structure.

Probabilistic Graphic Methods (PGMs) can process images with complex structures, owing to their power to extract the underlying relations among images (Shuman *et al.* 2013; Ortega *et al.* 2018). In modelling pixel points and their interactions using a graph, a structured environment emerges, shaped by the spatial characteristics of the pixels' geometric connections in various applications, including denoising (Liu *et al.* 2018), segmentation (Zhang *et al.* 2022a) and seismic detection problems (Cannavò *et al.* 2017). Beyond conventional graphs, PGMs have been extended to high-dimensional spaces, such as multilayer graphs (Das & Ortega 2020) and hypergraphs (Zhang *et al.* 2022b). Furthermore, graph neural networks (GNNs) and graph convolutional networks (GCNs) are important tools in image processing and computer vision (van den Ende & Ampuero 2020; Kim *et al.* 2021). Bayesian methods provide a framework for modelling uncertainty, learning from data and making predictions. They have found broad applications in a variety of seismic fields, such as seismic tomography (Zhao *et al.* 2022), full-wave inversion (Zhang *et al.* 2023) and ground motion prediction (Mu & Yuen 2016). Among all the PGMs and Bayesian methods, Markov Random Fields (MRFs) constitute a popular and effective approach for supervised structure learning tasks involving the mapping between complex geometric structures (Murphy 2001). MRFs provide an image restoration procedure, first suggested by Geman & Graffigne (1986), which is based on Bayesian inference for a spatially stochastic model. In contrast to convolution-based methods, the MRF procedure yields an optimal and mathematically tractable result for image processing (Blake *et al.* 2011; Zhou *et al.* 2024b).

Recently, many novel methodologies, within the category of physics-informed machine learning (Gou *et al.* 2023) have gained much attention. These methods combine conventional machine learning (primarily neural networks (Li *et al.* 2023)) and weighting mechanisms. They adapt the behaviour of the learning models to the non-linear features of the solution and introduce physical information as a guide, eventually improving the current limitations of learning models.

Combining the physics-informed mechanism and the MRF model, we propose a physics-informed probabilistic graphical model (PIPGM) that captures the relations between subdomains with multiple resolutions. The proposed method is inspired by the progress achieved in image super-resolution (Cheung *et al.* 2018; Zhou *et al.* 2024a) and image editing (Dhamo *et al.* 2020; Zhang *et al.* 2018), and is designed to fuse seismic velocity models with probabilistic graphical models (PGMs). We focus on models with well-defined, separate high-resolution (HR) and low-resolution (LR) areas, to enhance local HR structure and simultaneously preserve global smoothness in the fused model. By transferring the information from the HR subdomain, the details in the LR areas are enhanced by solving a maximum likelihood problem with prior knowledge from the HR areas. Evaluation tests on both a checkerboard and a fault zone model derived from the 2019 Ridgecrest, CA, earthquake are performed to demonstrate its efficacy. Quantitatively, model efficacy is evaluated by the deviation between observed and calculated traveltimes in fused models, relative to those from directly superimposed LR and HR models. Tests demonstrate that PIPGM is generally superior to widely used conventional methods (see Section 6).

The contributions of this article are two-fold: (1) we introduce a PIPGM fusion method for combining velocity maps with various resolutions and (2) we introduce physical information as a guide for the graph learning process.

2 FUNDAMENTAL MODEL SETUP

The objective is to estimate the true velocity field $\boldsymbol{A}$ from $\boldsymbol{A}_{\mathrm{LR}}$ (low-resolution) and $\boldsymbol{A}_{\mathrm{HR}}$ (high-resolution) models, focusing on optimal merging of their borders (see Fig. 1). A discrete class label map links the spatial velocity field together. It is initialized from a continuous velocity map $\boldsymbol{A}$ and a 6-cluster discrete label map $\boldsymbol{X}$. Each pixel, described by an (i, j) coordinate, contains a label $X_{i,j}$ and a velocity $A_{i,j}$, with velocities of the same label following the Gaussian distribution $\mathcal{N}(\mu_n, \sigma_n^2)$. Thus, in a graph, velocities $\boldsymbol{A}$ are on top of labels $\boldsymbol{X}$ (Fig. 2). d represents all possible labels of $X_{i,j}$ (namely, $d = \{1, 2, 3, 4, 5, 6\}$ here), and $\boldsymbol{D}$ represents all possible combinations of labels $\boldsymbol{X}$ for the entire map. The velocity map $\boldsymbol{A}$ is connected by class labels $\boldsymbol{X}$.

3 GAUSSIAN MIXTURE MODEL (GMM)

We use Gaussian mixture model (GMM) clustering to assign each continuous velocity pixel $A_{i,j}$ a discrete label $X_{i,j}$ (Fig. 3). We utilize these discrete labels to ensure the preservation of spatial coherence and the integrity of class boundaries, both essential for accurate data interpretation and model combining. GMM clustering is a widely used probabilistic clustering technique. In this model, data points are assumed to be generated from a mixture of a finite number of Gaussian distributions with unknown parameters (Shahrabi *et al.* 2016). Clustering aims at identifying these parameters, thus segmenting the data into clusters, each represented by a Gaussian distribution. The GMM clustering can be implemented using the Expectation-Maximization (EM) method. Here, the continuous velocity pixel is denoted by $A_{i,j}$ and the corresponding labels by $X_{i,j}$. The algorithm is described as follows:

(i) **Initialization:** Define $N = 6$ clusters (the selection of cluster number will be discussed in Section 6.2) and initialize their means μ_n, variance σ_n and mixing coefficients π_n are initialized either randomly or based on prior knowledge.

(ii) **E-Step:** Compute each data point's responsibility $p(x_i)$, the probability that belongs to the cluster n, using:

$$p(x_i) = \frac{\pi_n \mathcal{N}(A_{i,j}|\mu_n, \sigma_n)}{\sum_{n=1}^{N} \pi_n \mathcal{N}(A_{i,j}|\mu_n, \sigma_n)},$$

(iii) **M-Step:** Update parameters μ_n, σ_n and π_n to maximize observed data's log-likelihood:

$$\mu_n^{new} = \frac{1}{N_n} \sum_{i=1}^{N_{\mathrm{total}}} P(X_{i,j}) A_{i,j}, \tag{1}$$

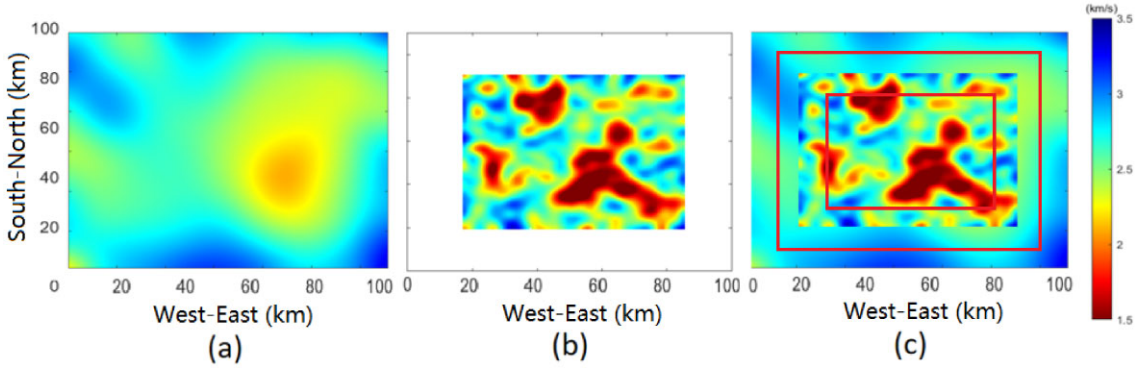

Figure 1. (a) Excerpt of *S*-wave velocities from the Southern California Earthquake Center (SCEC) Community Velocity Model (CVM) version S-4.26 (hereafter referred to as the low-resolution (LR) CVM) at 0.5 km depth around the Ridgecrest area. (b) High-resolution (HR) *S*-wave map from 1 Hz Rayleigh wave tomography from Zhou *et al.* (2022). (c) A direct superposition of the HR and LR models. These two models share some patterns in the low-velocity zones, but there are many mismatched detailed patterns where the two models overlap, which results in sharp and misaligned boundaries in those areas. PIPGM is applied to the mismatched boundary areas between the two red bounding boxes; note that the pixels in this area belong to the effective vertices set $\mathcal{V}$.

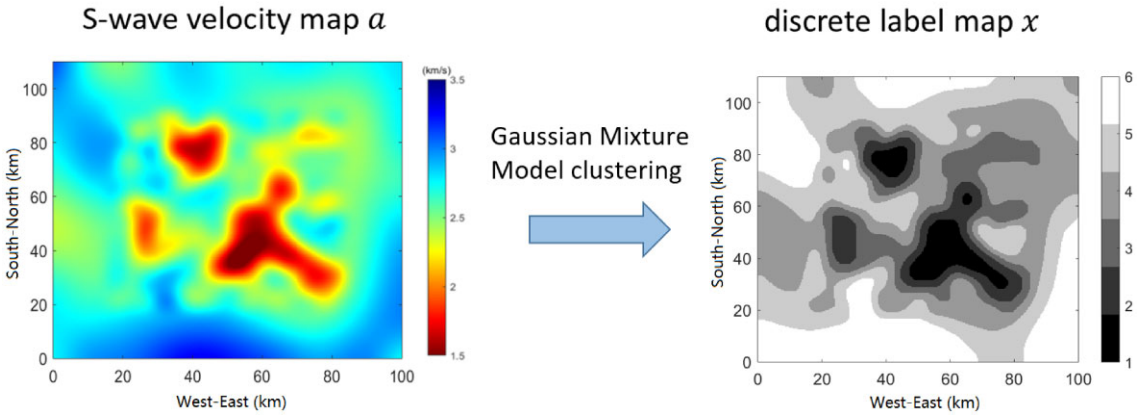

Figure 2. A 6-cluster Gaussian Mixture model clustering is applied on the continuous velocity map $\boldsymbol{A}$ (left-hand panel), and this derives a 6-cluster discrete label map $\boldsymbol{X}$ (right-hand panel). The pixels with similar velocity information have been assigned the same label.

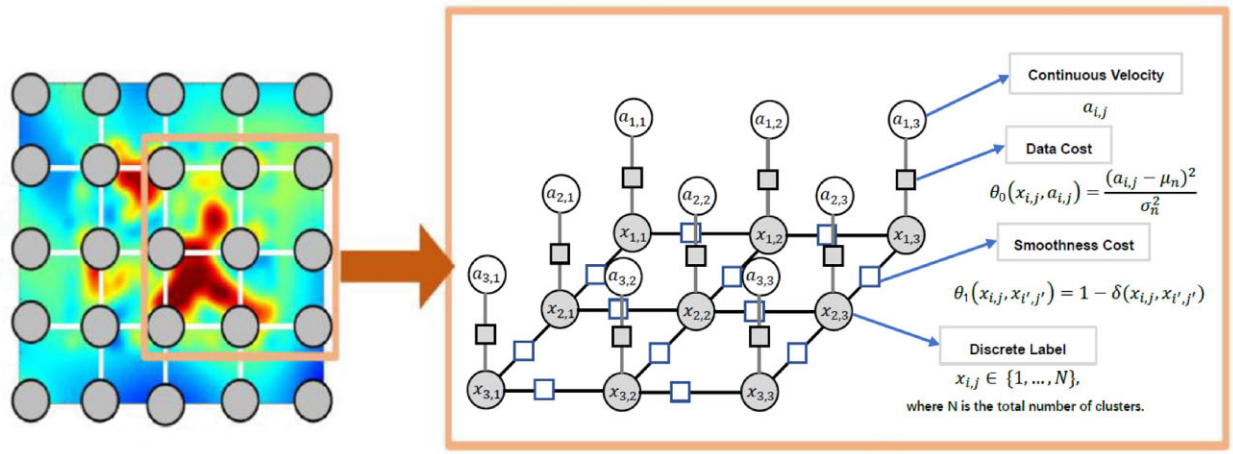

Figure 3. Each pixel has a continuous velocity value $a_{i,j}$ and a discrete label mask $x_{i,j}$. The object function designed for MAP estimation has two parts: (1) the data cost θ_0 (0th-order neighbouring potential) that forces the pixels with the same label to follow the same Gaussian distribution and (2) the smoothness cost θ_1 (1st-order neighbouring potential) that promotes smoothness among neighbouring pixels (Koller & Friedman 2009).

$$\sigma_n^{new} = \frac{1}{N_n}\sum_{i=1}^{N_{\text{total}}} P(X_{i,j})(A_{i,j}-\mu_n)(A_{i,j}-\mu_n)^T, \tag{2}$$

$$\pi_n^{new} = \frac{N_n}{N_{\text{total}}}, \tag{3}$$

(iv) **Convergence:** Stop if parameters or log-likelihood change is below a threshold or a maximum iteration count is reached. Assign $A_{i,j}$ to the cluster maximizing $P(X_{i,j})$.

4 MRF

4.1 Bayesian estimation framework

Given the prior probabilities $P(\boldsymbol{X})$ of label $\boldsymbol{X}$ and the likelihood densities $P(\boldsymbol{A} \mid \boldsymbol{X})$ of the observed velocity $\boldsymbol{A}$, the posterior probability is computed using the Bayes rule:

$$P(\boldsymbol{X} \mid \boldsymbol{A}) = \frac{P(\boldsymbol{A} \mid \boldsymbol{X})P(\boldsymbol{X})}{P(\boldsymbol{A})} \propto P(\boldsymbol{A} \mid \boldsymbol{X})P(\boldsymbol{X}). \tag{4}$$

Here, $P(\boldsymbol{A})$, the probability density function (PDF) of $\boldsymbol{A}$, is a fixed probability distribution (for given $\boldsymbol{A}$) and does not affect the maximum a posteriori (MAP) estimation solution. The Bayesian labelling problem requires finding the MAP configuration. The MAP of labelling for observation $\boldsymbol{A}$ is given by:

$$\boldsymbol{X}^* = \arg\max_{\boldsymbol{X}\in D} P(\boldsymbol{X} \mid \boldsymbol{A}), \tag{5}$$

where D denotes a set of possible candidates of the discrete labels $\boldsymbol{X}$, and $\boldsymbol{A}$ represents the observation of the continuous velocities.

We need the prior probabilities and likelihood functions to derive the MAP solution. The likelihood function $P(\boldsymbol{A} \mid \boldsymbol{X})$ depends on the noise statistics and the underlying transformation from truth to observation.

4.2 Neighbourhood system in MRF

The MRF, a model that analyses spatial relations, uses a neighbourhood system (Li 1994), defined as $\mathcal{N} = \{\mathcal{N}_{i,j} \mid \forall(i,j) \in \mathcal{V}\}$ (shown in Fig. 4), where $\mathcal{V}$ contains all pixel indices and $\mathcal{N}_{i,j}$ includes neighbouring pixels. The neighbouring system of the given pixel with index (i,j) can be decomposed as the union of 0th-, 1st-order, $\cdots$ neighbouring systems as

$$\mathcal{N}_{i,j} = \mathcal{N}^0_{i,j} \cup \mathcal{N}^1_{i,j} \cup \cdots, \tag{6}$$

and we define the nth-order neighbouring system of (i,j) as

$$\mathcal{N}^n_{i,j} = \{(i',j') \mid |i-i'| + |j-j'| = n\}, \tag{7}$$

which means (i',j') and (i,j) have a Manhattan distance (Liu *et al.* 2013) of n. The pair $(\mathcal{V},\mathcal{N})$ constitutes a graph, and the neighbouring system N for the pixel (i,j) consists of the centre node (0th-order neighbouring system) $\mathcal{N}^0_{i,j} = \{(i,j)\}$ and a pair of neighbouring pixels $\mathcal{N}^1_{i,j} = \{(i',j')\}$. To avoid artefacts at the boundary, we expand the image's boundaries by one pixel in all directions, where the values of the outermost pixels are directly replicated.

The MRF is on D with respect to $\mathcal{N}$ if (1) $P(X_{i,j}) > 0, \forall X_{i,j} \in d, \forall(i,j) \in \mathcal{V}$ (positivity) and (2) $P(X_{i,j} \mid X_{i',j'}, (i',j') \in \mathcal{V}) = P(X_{i,j} \mid X_{i',j'}, (i',j') \in \mathcal{N}_{i,j})$. Satisfying condition (1) above guarantees the model to be a random field. Condition (2) is the Markov local property. Considering that the label variable $\boldsymbol{X}$ depends on velocity $\boldsymbol{A}$ and is unobservable, it is assumed that the distribution of $P(\boldsymbol{X}|\boldsymbol{A})$ follows exponential distributions, using Bayes' rule and the conjugate distribution property (George *et al.* 1993).

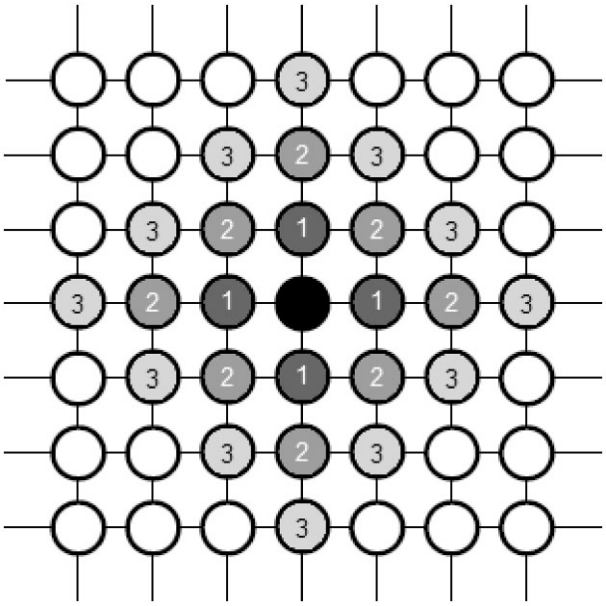

Figure 4. The neighbourhood system $\mathcal{N}_{i,j}$ (marked in grey/black) of the given centre node (i,j) (marked in black). The 1st-, 2nd- and 3rd-order neighbourhood systems of node (i,j) are marked with numbers '1', '2', '3' and they can be represented as $\mathcal{N}^1_{i,j}$, $\mathcal{N}^2_{i,j}$ and $\mathcal{N}^3_{i,j}$.

4.3 MRF prior and posterior energy

A model can be considered a valid MRF if and only if the probability distribution $P(\boldsymbol{X})$ of the configurations is an exponential distribution with normalization, defined as the following form

$$P(\boldsymbol{X}) = \frac{1}{Z_1} e^{-U_{\text{prior}}(\boldsymbol{X})}, \tag{8}$$

where Z_1 is a normalizing constant, and $U_{\text{prior}}(\boldsymbol{X})$ is the prior energy [section 4.2 in Koller & Friedman (2009)].

The prior energy $U_{\text{prior}}(\boldsymbol{X})$ can be expressed as the summation of neighbouring potentials

$$U_{\text{prior}}(\boldsymbol{X}) = \sum_{n\in\mathcal{N}} \theta_n(\boldsymbol{X}) = \sum_{\{(i,j)\}\in\mathcal{N}^0_{i,j}} \theta_0\left(X_{i,j}\right) + \sum_{\{(i',j')\}\in\mathcal{N}^1_{i,j}} \theta_1\left(X_{i,j}, X_{i',j'}\right) + \cdots, \tag{9}$$

where $\mathcal{N}$ is the set of all the possible neighbouring systems, and $\mathcal{N}^0_{i,j}$ and $\mathcal{N}^1_{i,j}$ denotes the 0th- and 1st-order neighbouring systems, with θ_0 and θ_1 representing the respective potentials. The 0th-order neighbouring system consists of the ensemble of all feasible indices (i,j), identified as the central pixel. This study focuses exclusively on the neighbouring potentials of 0th and 1st order while truncating higher order potentials as delineated in eq (9).

Because we assume that the velocities $\boldsymbol{A}$ with given labels $\boldsymbol{X}$ follow Gaussian distributions, the likelihood function can be expressed in an exponential format

$$P(\boldsymbol{A} \mid \boldsymbol{X}) = \frac{1}{Z_2} e^{-U_{\text{like}}(\boldsymbol{A}|\boldsymbol{X})}, \tag{10}$$

where $U_{\text{like}}(\boldsymbol{A} \mid \boldsymbol{X})$ is called the likelihood energy. According to the Bayes rule eq. (4), the posterior probability must be an exponential distribution

$$P(\boldsymbol{X} \mid \boldsymbol{A}) = \frac{1}{Z_3} e^{-U_{\text{post}}(\boldsymbol{X}|\boldsymbol{A})}, \tag{11}$$

where Z_2 and Z_3 are normalization constants. Taking the negative logarithm in eqs (10)–(11) gives the posterior energy

$$U_{\text{post}}(\boldsymbol{X} \mid \boldsymbol{A}) = U_{\text{prior}}(\boldsymbol{X}) + U_{\text{like}}(\boldsymbol{A} \mid \boldsymbol{X}) + C, \tag{12}$$

where C is a constant related to the normalization constants Z_1, Z_2 and Z_3. Hence, given a fixed $\boldsymbol{A}$, X is also an MRF on d with respect to $\mathcal{N}$. The MAP solution is equivalently found by

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}\in D} U_{\text{post}}(\boldsymbol{X} \mid \boldsymbol{A}), \tag{13}$$

which minimizes the negative log-likelihood problem of eq. (11).

In summary, the MRF modelling process consists of the following steps: Defining a neighbourhood system $\mathcal{N}$, defining the prior potentials θ_0, deriving the likelihood energy $U_{\text{like}}(\boldsymbol{A}|\boldsymbol{X})$ and deriving the posterior energy $E(\boldsymbol{X})$ (which can be expressed as the summation of neighbouring potential functions). Eqs (10)–(13) show that we can express the posterior probability $P(\boldsymbol{X}|\boldsymbol{A})$ into the prior energy $U_{\text{prior}}(\boldsymbol{X})$ (which can be measured by multiple potentials) and the likelihood function energy $U_{\text{like}}(\boldsymbol{A}|\boldsymbol{X})$, and this gives a good reason for using MRF priors, which means that we can measure conditional probabilities $P(\boldsymbol{X}|\boldsymbol{A})$ without knowing their specific expression.

5 PIPGM

5.1 Physics-informed mechanism

PIPGM follows a first-order MRF setting (Fig. 4) where each random variable has four neighbours on which it is conditionally dependent. The full conditional probability of the discrete random variable $X_{i,j} \in \{1, \ldots, 6\}$ is the exponential of the sum of potentials (four 1st-order neighbouring potentials θ_1 between cluster labels and one 0th-order centre data potential θ_0 between cluster label and velocity) in conventional MRF settings. In image process problems, optimizing the entire map can be broken down into several suboptimization problems that iteratively optimize each pixel (Pulli *et al.* 2012). Inserting eq. (9) into eq. (12), we have

$$\begin{aligned} -\log p(X_{i,j} \mid A_{i,j}) &= U_{\text{post}}(X_{i,j} \mid A_{i,j}) \\ &\propto \theta_0(X_{i,j}, A_{i,j}) \\ &\quad + \sum_{(i',j')\in\mathcal{N}_{i,j}} \theta_1(X_{i,j}, X_{i',j'}) + C, \end{aligned} \tag{14}$$

where C is a constant related to normalization factors Z_1, Z_2 and Z_3 (see eqs 8, 10 and 11), and

$$\theta_0\left(X_{i,j}, A_{i,j}\right) = \frac{(A_{i,j} - \mu_n)^2}{\sigma_n^2} \tag{15}$$

is the 0th-order neighbouring potential (Li 2012) (also known as the data cost function) that relates $X_{i,j}$ with the observed velocity data $A_{i,j}$. μ_n and σ_n^2 are the mean and variance of all pixels with the same cluster label $n = X_{i,j}$. It promotes that continuous velocity values $\boldsymbol{A}$ sharing pixels with the same discrete label $\boldsymbol{X}$ follow the same Gaussian distribution.

$$\theta_1\left(X_{i,j}, X_{i',j'}\right) = 1 - \delta(X_{i,j}, X_{i',j'}) \tag{16}$$

is the 1st-order neighbouring potential (Li 2012) (also known as the smoothness cost function) that relates $X_{i,j}$ to the 1st-order neighbouring variable $X_{i',j'}$ (see Fig. 4). This function encourages the neighbouring pixels to share the same discrete label $X_{i,j}$, promoting the model's local smoothness.

The effectiveness of conventional MRF approaches, or those based on potential functions, may be constrained when applied to complex geological structures. Assigning different neighbouring pixels with various importance weights based on physical information (also known as the physics-informed mechanism) can effectively remove the anisotropic features of the model gradients. This approach can significantly improve inversion results, particularly in the context of geological structure.

In seismic tomography, the coverage of the ray path varies, with the coverage of the densest ray path usually leading to more reliable estimations of velocities. Here, we introduce a confidence score $v_{R_{i,j}}$ based on the logarithmic format of the ray density for each pixel:

$$v_{R_{i,j}} = \alpha_R \log(D_{i,j} + 1) + \beta_R, \tag{17}$$

where $D_{i,j} \in [0, 315]$ is the number of ray paths through a given pixel. The logarithmic scaling parameters α_R and β_R are empirically chosen as 0.08 and 0.90, respectively, resulting in $v_{R_{i,j}} \in [0.9, 1.1]$.

We combine the gradient information from both the LR and HR maps as the prior estimation of the sharpness of the local patterns (as shown in Fig. 5). The gradient is calculated by the Prewitt operator approximating the 1st-order derivatives of 2-D images (Zuniga & Haralick 1987). The operator uses two 3×3 kernels convolved with the original image to calculate approximations of the derivatives—one for horizontal changes and one for vertical. The kernels for the Prewitt operator can be represented as

$$K^x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad K^y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \tag{18}$$

The convolution of these kernels ($\boldsymbol{K}^x$ and $\boldsymbol{K}^y$) with the original image yields two gradient images, one for the x-direction ($\boldsymbol{G}^x$) and one for the y-direction ($\boldsymbol{G}^y$).

Once $\boldsymbol{G}^x$ and $\boldsymbol{G}^y$ have been determined, we can find the gradient magnitude $G_{i,j}$ at the pixel with index (i, j) as

$$G_{i,j} = \sqrt{{G_{i,j}^x}^2 + {G_{i,j}^y}^2}. \tag{19}$$

The resulting image $\boldsymbol{G}$ is a gradient image showing the intensity of the edge. We applied the Perwitt operator on the LR and HR maps to generate the gradient images $\boldsymbol{G}^{\text{LR}}$ and $\boldsymbol{G}^{\text{HR}}$. We empirically choose the weighting parameter $\lambda = 0.2$ ($0 \le \lambda \le 1$) for balancing the LR and HR gradients:

$$G'_{i,j} = (1 - \lambda) G_{i,j}^{\text{LR}} + \lambda G_{i,j}^{\text{HR}}. \tag{20}$$

The overall range of the pixel values inside the gradient matrices $\boldsymbol{G}^{\text{LR}}$ and $\boldsymbol{G}^{\text{HR}}$ is [0.05,0.68]. The scaling parameters α_G and β_G are empirically set as 0.36 and 0.85, respectively, and this guarantees $v_{G_{i,j}} \in [0.9, 1.1]$:

$$v_{G_{i,j}} = \alpha_G \left(1 - G'_{i,j}\right) + \beta_G. \tag{21}$$

Generally, the larger the gradient weight term $v_{G_{i,j}}$, the lower the importance of the local smoothness cost. The gradient weight term $v_{G_{i,j}}$ is joined with the previous ray-density weight $v_{R_{i,j}}$, defining a physics-informed weight $\omega_{i,j}$

$$\omega_{i,j} = v_{R_{i,j}} v_{G_{i,j}}. \tag{22}$$

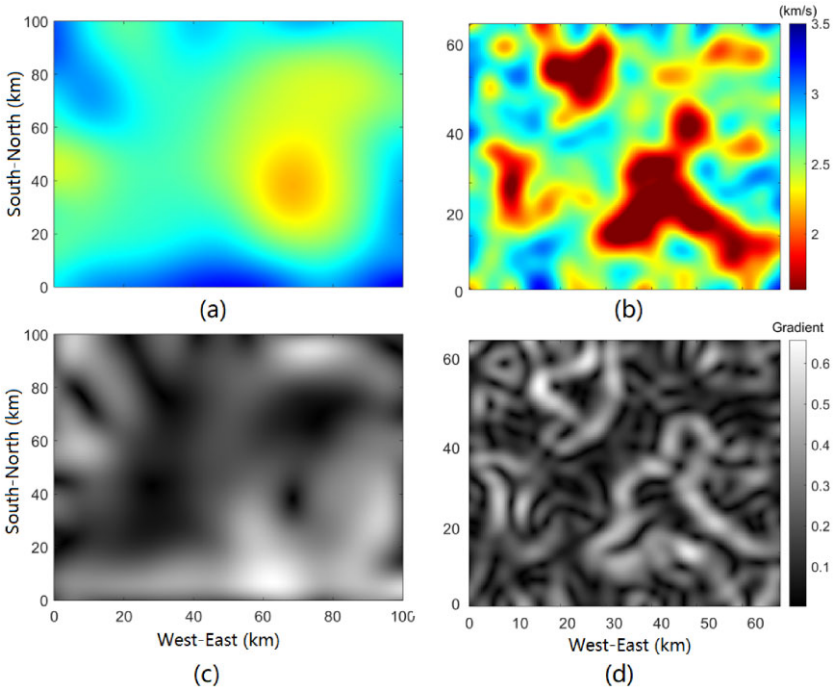

Figure 5. (a) LR CVM around the Ridgecrest area. (b) HR S-wave map from 1 Hz Rayleigh wave tomography. Panels (c) and (d) corresponding gradient maps of (a) and (b), respectively. In the gradient maps, brighter (darker) areas indicate that velocities change sharply (slightly).

With the parameters chosen above, the range of the physics-informed weight is [0.81,1.21] with a mean of around 1. The physics-informed weight $\omega_{i,j}$ can adaptively assign a larger weight to the trusted nodes based on existing physical information. We used ray-path density and gradient information, both exhibiting pronounced anisotropic characteristics (as shown in Figs 5 and 9), thereby facilitating the model's capability to explore more complex formations within these regions. Assigning the physics-informed weight in eq. (14), we obtain the posterior probability function in the PIPGM

$$-\log p(X_{i,j} \mid A_{i,j}) \propto \omega_{i,j}\theta_0(X_{i,j}, A_{i,j}) + \sum_{(i',j')\in\mathcal{N}_{i,j}} \omega_{i',j'}\theta_1(X_{i,j}, X_{i',j'}) + C, \tag{23}$$

where C is a constant related with to normalization factors Z_1, Z_2 and Z_3 (see eqs 8, 10 and 11). The objective function of the MAP problem of $X_{i,j}$ becomes

$$\begin{aligned} X^*_{i,j} &= \arg\max_{X_{i,j}} p(X_{i,j} \mid A_{i,j}) \\ &= \arg\min_{X_{i,j}} \omega_{i,j}\theta_0(X_{i,j}, A_{i,j}) \\ &\quad + \sum_{(i',j')\in\mathcal{N}_{i,j}} \omega_{i',j'}\theta_1(X_{i,j}, X_{i',j'}) + C. \end{aligned} \tag{24}$$

5.2 Markov Chain Monte Carlo (MCMC) and Gibbs sampling

MCMC is a statistical method used to sample probability distributions (Melas & Wilson 2002; Sambridge & Mosegaard 2002). Gibbs sampling is a specific MCMC algorithm that iteratively samples a multivariate probability distribution from the conditional distributions of each variable given the current values of the other variables (Carlo 2004). Combining MCMC with Gibbs sampling enables estimating complex probability distributions without explicit knowledge of the distribution.

We use the MCMC method with Gibbs sampling to solve eq. (13). The algorithm assigns a discrete label to each pixel, creating a map-wide label distribution overview. This enables calculating the discrete label probability $P(X_{i,j} = n)$ by finding the frequency of each label n in Step 10, giving the map's label distribution. In Step 12, the continuous variable $A_{i,j}$ is sampled from a Gaussian mixture model (GMM), allowing velocities beyond a single label's distribution and considering neighbouring clusters. In the MRF structure, the update is achieved by calculating the probability for each of the possible labels $n \in \{1, \cdots, 6\}$ at (i, j) using eq. (23) and randomly selecting from this distribution (see Fig. 6).

The velocity map $\boldsymbol{A}$ is initialized with the superimposed HR and LR velocity maps, see Figs 9(a1) and (b1), and the label map $\boldsymbol{X}$ is initialized with a Gaussian mixture model clustering with the total cluster number $N = 6$ (will be discussed in Section 6.1), similar to Figs 9(a3) and (b3). All velocities with the label n follow the same Gaussian distribution $N(\mu_n, \sigma_n^2)$. The expectation-maximization (EM) algorithm (McLachlan & Krishnan 2007), an iterative method to find MAP estimates of parameters, updates Gaussian parameters μ_n and σ_n^2. The termination criterion is either reaching 10 000 iterations or observing that the cumulative absolute difference across all pixels between consecutive iterations falls below 0.1 km s^{-1}, whichever is achieved first. The algorithm has been summarized in Algorithm 1.

Algorithm 1 MCMC Method for MRF

1. Input: $\boldsymbol{A}_{LR}$ and $\boldsymbol{A}_{HR}$

2. Initialize the velocity model $\boldsymbol{A}$ by superimposing $\boldsymbol{A}_{HR}$ over $\boldsymbol{A}_{LR}$
   Initialize $\boldsymbol{X}$, $\mu_{\mathbf{n}}$ and $\sigma_{\mathbf{n}}$ with Gaussian Mixture Model (GMM) clustering
3. **for** each EM iteration **do**
4. Construct PIPGM
5. **for** $t = 1$ to max iteration T **do**
6. (E-Step) Gibbs Sampling
7. **for** each pixel $(i, j) = (1, 1)$ to the maximum index (I, J) **do**
   (updating all the discrete labels $X_{i,j}$)
8. $X_{i,j}^{(t+1)} \sim P\left(X_{i,j} \mid X_{1,1}^{(t+1)}, \ldots, X_{i,j-1}^{(t+1)}, A_{i,j}^{(t)}, X_{i,j+1}^{(t)}, \ldots, X_{i,j}^{(t)}\right)$
9. **end for**
10. Update $P(X_{i,j} = n) = \frac{N_n}{N_{\text{total}}}$, where N_n is number of pixels with label n and N_{total} is number of pixels.
11. **for** each pixel $(i, j) = (1, 1)$ to the maximum index (I, J) **do**
    (updating all the continuous velocities $A_{i,j}$)
12. $A_{i,j} \sim \sum_{n=1}^{6} P(X_{i,j} = n) N(\mu_n, \sigma_n^2)$
13. **end for**
14. (M-Step) Update Gaussian parameters μ_n and σ_n^2 with the sample mean $\mu_n = \frac{1}{N_n} \sum_{X_{i,j}=n} A_{i,j}$ and sample variances $\sigma_n^2 = \frac{\sum_{X_{i,j}=n} (A_{i,j} - \mu_n)^2}{N_n}$
15. **end for**
16. **end for**

17. **return** $\boldsymbol{X}$, $\boldsymbol{A}$ (for each pixel)

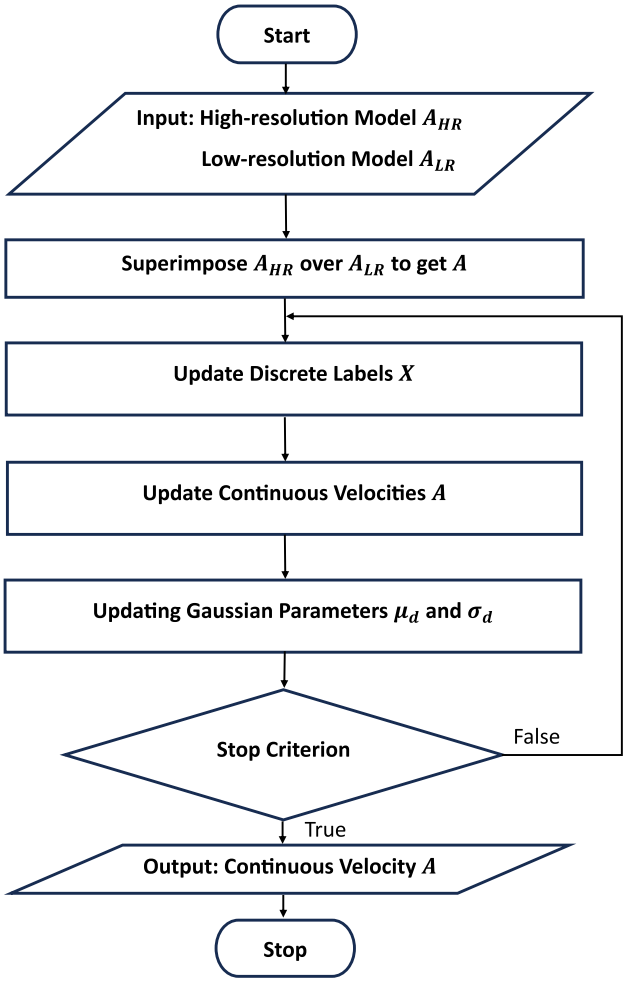

Figure 6. The pipeline of the iterative update policy for both pixel labels (discrete class labels) and pixel values (continuous velocity values).

6 EXPERIMENTS

To evaluate the efficacy of the PGM fusion method, we use both a checkerboard model and the fault zone structure imaged from the 2019 Ridgecrest, CA, earthquake sequence. The Ridgecrest fault zone image consists of a shallow (representing a depth of 0.5 km) high-resolution Rayleigh wave model (Zhou *et al.* 2022), from which the *S*-wave velocity is approximated by dividing by 0.9.

Fig. 7 shows the smoothed results with different smoothing levels. The aggressive, simple smoothing policy in Fig. 7(c) (7×7 average filter) removes the sharp boundaries while the details are lost. Simple and mild smoothing in Fig. 7(d) (3×3 average filter) preserves the details together with the artificial boundaries. It is essential to achieve a trade-off between the two cases, this is the aim of PIPGM. To assess the balance between traveltime among stations and their residuals, 36 evaluation points (marked as red 'X', with 10 situated along each edge) are positioned at the boundary dividing the LR and HR zones. These residuals gauge the extent of detailed information retained in the merged velocity model relative to the HR maps.

6.1 Graphical structure order test

In an MRF, the variables are represented as nodes in the graph, and their dependencies with the neighbours are decided by the truncation order in eq. (6). We name the MRF with the neighbouring system truncated at nth order as 'nth order MRF' for brevity. A 1st-order neighbouring system structure is a subset of nodes in which every two distinct nodes are directly adjacent. Usually, the neighbouring system's order significantly affects the model's smoothing results. If an MRF model is based on a first-order neighbouring system, it can account for interactions between immediate neighbours. In the context of image smoothing, it results in the enforcement of local smoothness, where each pixel is encouraged to be similar to its immediate neighbours. When a model involves a higher-order neighbouring system (the Manhattan distance of the pixels in this set from the centre pixel may be greater than 1), it can capture more complex relations and dependencies among variables. Higher-order MRFs can enforce smoothness over larger regions, allowing the model to preserve the high-level structures (usually related to the low-frequency patterns) and avoid sharp boundaries or noisy details.

Since information from a wider range of neighbours is considered, higher-order MRFs can be computationally more intensive in one iteration and may reach the optimum with fewer learning iterations. On the other hand, first-order MRFs are simple and efficient, and they can sometimes preserve some noisy details. The choice between first-order and higher-order MRFs depends on the specific requirements of the task and the trade-off between model complexity and computational efficiency. In the velocity fusion problem, we demonstrate the fused models with 1st-, 2nd- and 3rd-order MRF structures in Fig. 8. Considering that the Ridgecrest model measures the structure from a relatively limited region and that we prefer the rich detailed structures from the HR model, we choose the 1st-order MRF neighbouring structure in the following comparison experiments.

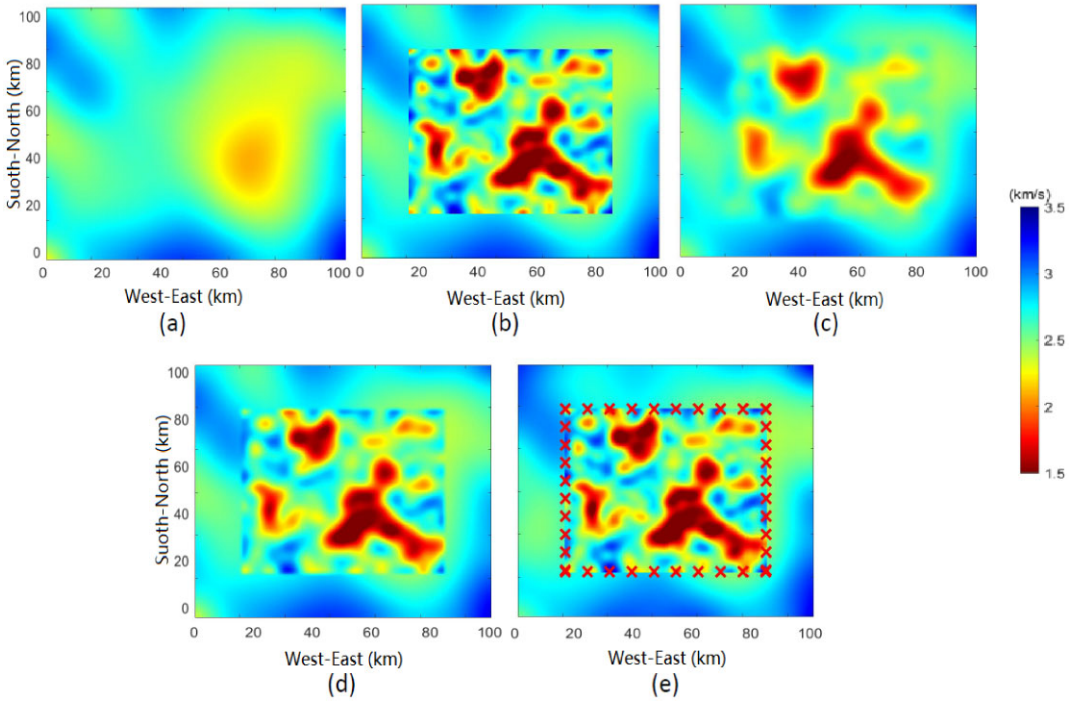

Figure 7. (a) LR model (Same as Fig. 1a). (b) Same as Fig. 1(c). (c–d) Combined LR and HR models, smoothed by (c) 7 × 7 and (d) 3 × 3 average filters. (e) Synthetic stations (red 'X's) are deployed on the boundaries between HR and LR models for evaluation.

6.2 Cluster number test

The number of GMM clusters influences the model complexity and interpretability of the results. More clusters result in a more complex model, which can capture intricate data structures better and lead to more detailed insights into the data. However, it increases the risk of overfitting.

The optimal number of clusters is crucial in GMM and other clustering techniques. Several methods help determine an appropriate number of clusters, such as the Akaike Information Criterion (AIC), the Bayesian Information Criterion (BIC) and the silhouette score. These methods balance the trade-off between the goodness-of-fit and the complexity of the model.

For this experiment, we tested and compared the cluster number sequences $N \in \{3, 4, 5, 6, 7, 9\}$, which is prevalent in practical uses of MRFs. Fig. 10 shows that the larger the number of clusters, the more detailed information is preserved in the HR models, and the larger the computation recourse it needs. This implies that a trade-off between cost and performance needs to be achieved. Fig. 11 demonstrates the number of clusters against RMSE error (left vertical axis) and computation time (right vertical axis) in the Ridgecrest test. The RMSE error experiences a significant decrease while the cluster number grows from 3 to 6, with limited decrease for larger cluster numbers, and there is a rapid growth in run time when the cluster number exceeds 6. For these reasons, we empirically choose the cluster number to be 6 as a trade-off between model performance and computation complexity.

6.3 Comparison with conventional methods

To demonstrate the advantage of the proposed PIPGM, we compare its performance with some commonly used conventional methods on both the synthetic checkerboard model and the real-data Ridgecrest model with interstation *S*-wave traveltime deviations before and after the merge. First, we briefly introduce several popular conventional fusion methods.

6.3.1 Gaussian smoothing filter (GF)

A GF is a data processing technique that reduces noise and smooths out signals or data distributions. This method involves applying a Gaussian kernel function to the data, a bell-shaped curve that weights the data points based on their distance from a central point. A smoothing parameter defines the Gaussian kernel called the kernel window size or standard deviation, which controls the amount of smoothing applied to the data. A larger window size leads to a wider, smoother curve, while a lower bandwidth results in a narrower, more detailed curve. GFs are commonly used in image and signal processing, and data analysis applications. It is a powerful technique that can effectively remove noise and improve the clarity of data, but can also introduce bias and distortions in the data if the smoothing parameter is not chosen carefully.

6.3.2 DL smoothing

DL smoothing (Yang *et al.* 2012) is a data processing technique used to de-noise and smooth misaligned patterns in the signals or images. It involves representing the data as a sparse linear combination of a set of basis vectors or atoms, which is learned from the data itself through an iterative process. In this process, the algorithm attempts to identify an array of basis vectors capable of representing the data with minimal error. Once the dictionary is learned, it can be used to transform the data into a sparse representation, where most of the coefficients are zero. This sparse framework is subsequently used in the denoising and smoothing of the data, achieved through the selective elimination or alteration of coefficients associated with noise

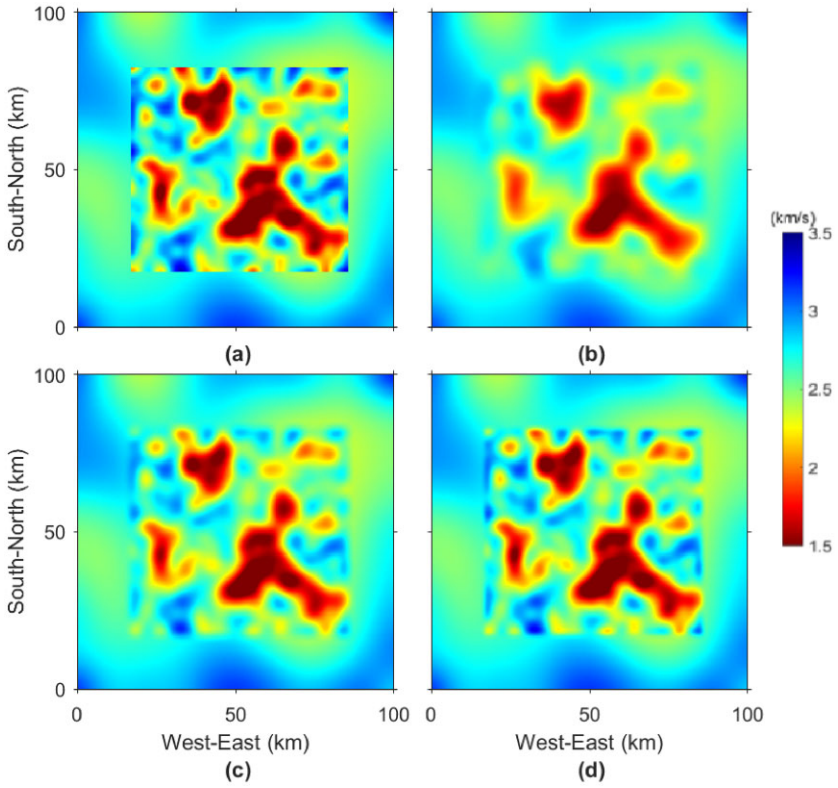

Figure 8. (a) Direct superposition of the HR 1 Hz Rayleigh wave tomographic velocity and CVM LR models for the Ridgecrest area. (b–d) Combined LR and HR models, smoothed by (b) 3rd-order, (c) 2nd-order and (d) 1st-order MRFs. Generally, the higher the order is, the larger the range of neighbouring pixels will be taken into consideration, and this leads to a model with smoother local patterns.

or extraneous features. DL smoothing has been applied to a wide range of data processing applications, including image processing, audio processing and signal processing. It is a powerful technique that effectively removes noise and preserves the underlying structure of the data, but it requires a large amount of training data and can be computationally expensive.

6.4 Results

We show the fusion of HR and LR components for directly superimposed checkerboard and Ridgecrest velocity models in Figs 9(a1) and (b1), which both have an HR region in the centre, surrounded by LR velocities in the surrounding areas. The checkerboard model has 40 × 40 pixels in the 100 km × 100 km LR area and 40 × 40 pixels in the 40 km × 40 km HR area. The fused model has 100 × 100 pixels in the 100 km × 100 km LR area. Similarly, the Ridgecrest model has 50 × 50 pixels in a 100 km × 100 km LR area and 192 × 224 pixels in a 58 km × 64 km HR area. The fused model has 330 × 350 pixels in a 100 km ×100 km area. Figs 9(a2) and (b2) shows the checkerboard and Ridgecrest stations along with the ray path density. For the checkerboard model, the stations are evenly distributed, whereas the stations for the Ridgecrest model are highly irregular, reflecting the pattern used in Zhou *et al.* (2022). Figs 9(a3) and (b3) shows the label mask maps from the last iteration of the PIPGMs. Pixels with the same label indicate that these areas potentially share similar velocity patterns and are sampled from the same distribution. The smoothed fusion results with the 5 × 5 GF, DL (Yang *et al.* 2012) and PIPGM are shown in Fig. 9(a4–a6) and (b4–b6). The results show that the learning methods (DL and PIPGM) preserve more detailed information than direct Gaussian smoothing. This is because the learning methods adaptively find the fusion parameters that optimize the accuracy of the representation, while Gaussian smoothing only combines the neighbouring pixels with a predefined kernel.

We use multiple metrics to evaluate the model fusion efficacy: traveltime root-mean-squared-error (RMSE, which measures information lost after model fusion Bianco *et al.* (2019)), Naturalness Image Quality Evaluator (NIQE, a common-used measurement for image quality Mittal *et al.* (2012)), Peak Signal-to-Noise Ratio (PSNR, measuring the sharpness of images Poobathy & Chezian (2014)) and the Frechet inception distance (FID, capturing similarities between the original and fusion models Chong & Forsyth (2020)) in Table 1. In the checkerboard test, because the pattern is simple and the stations are evenly distributed, all the learning methods achieve similar performance. For the realistic Ridgecrest model, however, the PGMs outperform the DL model, as the latter is sensitive to the orientation of the patches while the graphical models are rotationally invariant. We observe further improvements after the ray density information is informed in the PIPGM.

7 DISCUSSION

The proposed graphical model modifies each data point within a grid-based framework. Unlike traditional methodologies, which typically adjust individual data points in isolation, this model incorporates the influence of adjacent data points into its recalibration process. Specifically, the algorithm processes each pixel, modifying its values to ensure a more cohesive integration with its immediate

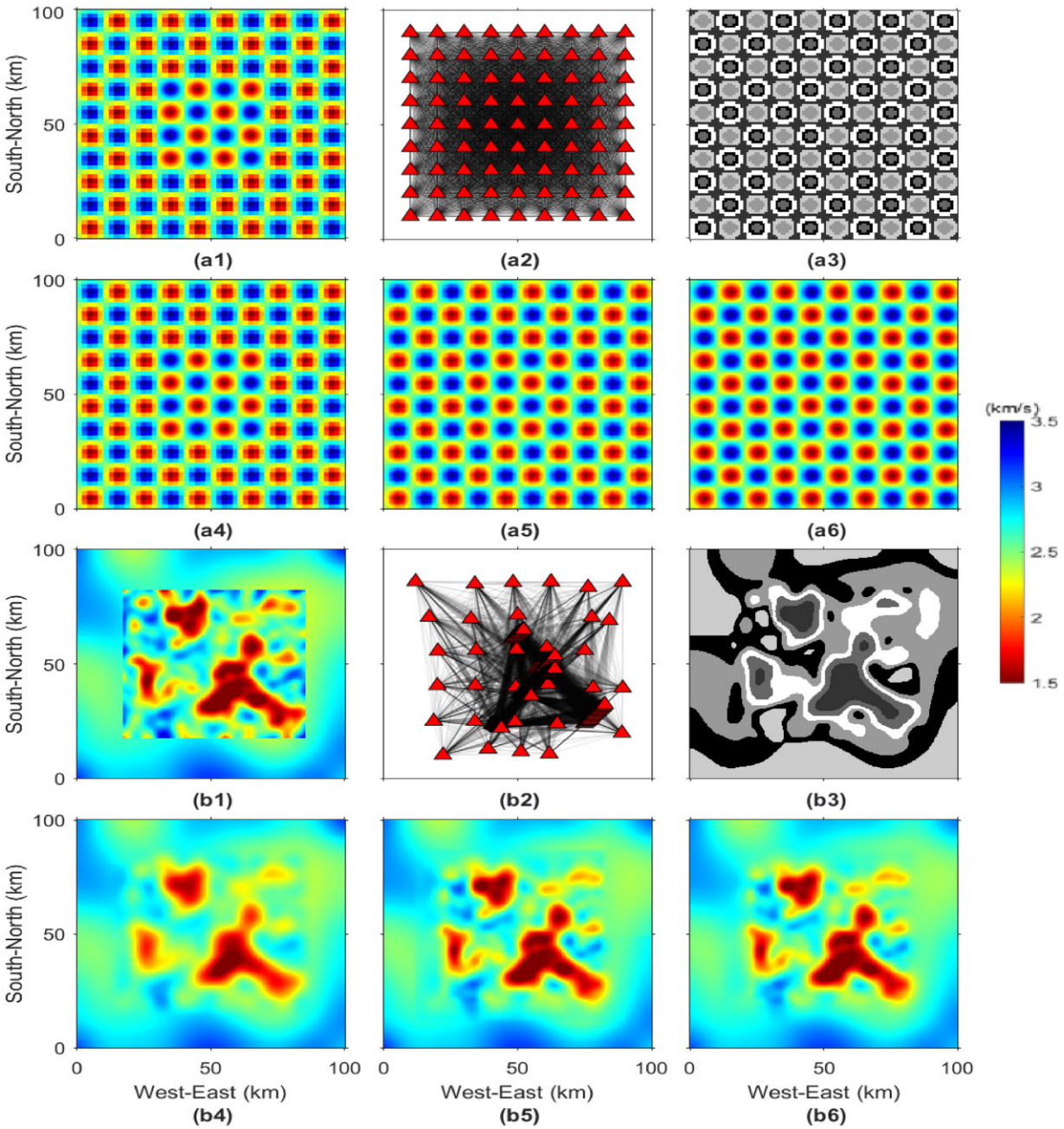

Figure 9. (top, 'a' panels) Checkerboard and (bottom, 'b' panels) Ridgecrest models. (1) Superimposed HR and LR models. (2) Station location and ray density. (3) 6-class label mask maps for HR models (pixels with the same label are learned together). (4) Smoothing results with a 5 $\times$ 5 Gaussian filter (GF). (5) and (6) Fusion results with DL and with PIPGM, respectively.

surroundings. This procedure is akin to a diffusion process, reminiscent of the dispersion patterns observed when ink is introduced into a body of clear water. The algorithm iterates over each pixel, fine-tuning its chromatic attributes to promote a more uniform alignment with proximal pixels. Such a strategy enhances the overall smoothness of the output and mitigates visual discrepancies, thereby elevating the uniformity and coherence across the model's entirety. The method assimilates vital elements of the rotation-invariant property and provides a flexible framework for varying data quantities for training. Consequently, this methodology promotes more inclusive, adaptable and precise modelling of Earth's subsurface structures.

Geological formations are often anisotropic, meaning their properties vary depending on the direction in which they are measured. For example, formations are often laterally continuous and vertically stratified. Standard MRF schemes, which assume homogeneous properties (same properties in all directions), can lead to errors when applied to such formations. PIPGM, on the other hand, considers the anisotropic nature of geological formations, leading to more accurate results. PIPGM provides an edge-preserving method based on the information from the neighbouring pixels, which is effective for reconstructing subsurface models. As a data-driven technique, PIPGM holds the capacity for adaptation across various

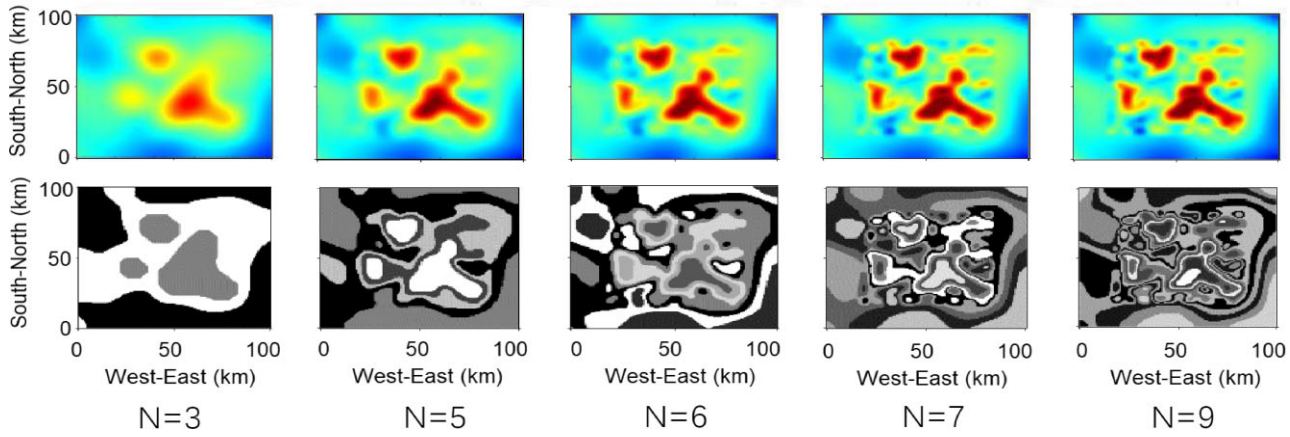

Figure 10. Fusion using cluster numbers $n = 3$, 5, 6, 7 and 9. (top row) fused velocity model and (bottom row) cluster distribution. Generally, the larger the number of clusters, the more detailed information is preserved in the HR models and the larger the computation recourse it needs. A trade-off is required to balance the cost and the performance.

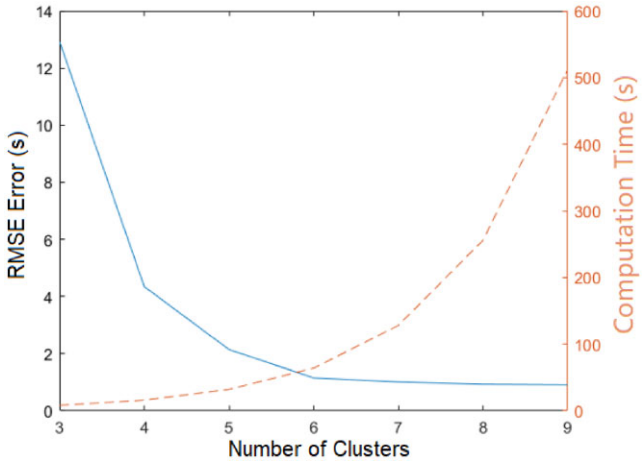

Figure 11. The number of clusters versus RMSE error (left vertical axis, corresponding to the blue solid line) and run time (right vertical axis, corresponding to the orange dashed line) on the Ridgecrest model. Generally, the larger the cluster number is, the smaller the RMSE error and the longer the running time is. We choose cluster number 6 to balance the model performance and the computation complexity.

Table 1. Evaluation results for Gaussian Filtering (GF), DL, classical PGM and PIPGM for both checkerboard and the Ridgecrest model. Evaluation metrics are traveltime root-mean-square error (RMSE), naturalness image quality evaluator (NIQE), peak signal-to-noise ratio (PSNR) and Fréchet inception distance (FID). ↓ indicates smaller is better, and ↑ opposite.

	RMSE/s↓	NIQE↓	PSNR/dB↑	FID↓
Checkerboard GF	1.65	7.68	14.58	45.75
” DL	1.18	5.44	15.70	33.85
” PGM	1.14	**5.40**	**16.14**	32.49
” PIPGM	**1.06**	5.41	**16.14**	**32.46**
Ridgecrest GF	3.52	12.41	21.80	61.39
” DL	2.61	7.29	22.36	54.25
” PGM	2.27	6.70	23.04	47.49
” PIPGM	**2.17**	**6.59**	**23.16**	**47.18**

seismological simulation methods. Integrating PIPGM into simulations such as the Finite-Difference Method (Olsen *et al.* 1995) or the Spectral-Element Method (Tromp *et al.* 2008) might generate more accurate ground motion simulations by ensuring smoother transitions between HR and LR areas, highlighting its potential for refined seismic analysis and modelling in complex geological settings.

8 CONCLUSIONS

We developed an approach to fuse multiresolution seismic velocity maps with PGMs. PIPGM achieves a velocity model by smoothing the generally undesired sharp boundaries between LR and HR components and preserving the detailed information from the HR models.

We tested the efficacy of the proposed fusion method using a checkerboard model and a realistic, complex fault zone model around the 2019 *M*7.1 Ridgecrest earthquakes. The tests for the checkerboard model, which is simple with evenly distributed stations, show that PIPGM outperforms all the baseline methods. All the learning-based techniques used to combine the maps (including the proposed PGM and PIPGM methods) significantly outperform the conventional methods, since the parameters are adaptively learned from the pairwise data. For Ridgecrest, the PIPGM methodology demonstrates a 38 per cent reduction of traveltime residuals compared to Gaussian kernel smoothing. This result is likely due to the poor performance of traditional techniques in handling the nonuniform-distributed data evenly, while PIPGM adaptively chooses the weights according to the intricate structure of the Ridgecrest model.

The proposed PIPGM outperforms traditional techniques in integrating maps of varying resolution, particularly when the map complexity escalates and the distribution of data points is uneven. Prospective future work involves addressing the challenge of irregular resolution within the HR domain, which is crucial for enhancing the fidelity and applicability of our models and potentially improves the understanding and application of various real-world models. Finally, we recommend testing PIPGM directly by comparison of synthetic and observed waveforms.

ACKNOWLEDGMENTS

This research was supported by the Statewide California Earthquake Center (SCEC Contribution Number 13436) and is supported by SCEC based on Award DE-SC0016520 from the U.S. Department of Energy.

DATA AND RESOURCES

The used seismic and station data are accessed from the FDSN 3J:RAMP deployment of 3C nodal for the July Searles Valley 2019 Earthquake https://doi.org/10.7914/SN/3J_2019. The low-resolution velocity model can be accessed from SCEC CVM version S4.26-M01 https://doi.org/10.1002/2014JB011346. The code supporting this study is available on Zenodo https://doi.org/10.5281/zenodo.11411903 and GitHub (https://github.com/zhz039/GLvelocitymodel).

REFERENCES

Ajala, R. & Persaud, P., 2021. Effect of merging multiscale models on seismic wavefield predictions near the southern San Andreas fault, *J. geophys. Res.,* **126**(10), e2021JB021915.

Barbarossa, S. & Sardellitti, S., 2020. Topological signal processing over simplicial complexes, *IEEE Trans. Sig. Process.,* **68,** 2992–3007.

Bianco, M. & Gerstoft, P., 2017. Dictionary learning of sound speed profiles, *J. acoust. Soc. Am.,* **141**(3), 1749–1758.

Bianco, M.J. & Gerstoft, P., 2018. Travel time tomography with adaptive dictionaries, *IEEE Trans. Comput. Imag.,* **4**(4), 499–511.

Bianco, M.J., Gerstoft, P., Olsen, K.B. & Lin, F.-C., 2019. High-resolution seismic tomography of Long Beach, CA using machine learning, *Sci. Rep.,* **9**(1), 1–11.

Blake, A., Kohli, P. & Rother, C., 2011. *Markov Random Fields for Vision and Image Processing,* MIT Press.

Cannavò, F. *et al.*, 2017. A multivariate probabilistic graphical model for real-time volcano monitoring on Mount Etna, *J. geophys. Res.,* **122**(5), 3480–3496.

Carlo, C.M., 2004. Markov chain Monte Carlo and Gibbs sampling, Lecture Notes for EEB 581.

Cheung, G., Magli, E., Tanaka, Y. & Ng, M.K., 2018. Graph spectral image processing, *Proc. IEEE,* **106**(5), 907–930.

Chong, M.J. & Forsyth, D., 2020. Effectively unbiased fid and inception score and where to find them, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 6070–6079.

Das, P. & Ortega, A., 2020. Graph-based skeleton data compression, in *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP),* pp. 1–6, IEEE.

Dhamo, H., Farshad, A., Laina, I., Navab, N., Hager, G.D., Tombari, F. & Rupprecht, C., 2020. Semantic image manipulation using scene graphs, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 5213–5222.

Fang, H. & Zhang, H., 2014. Wavelet-based double-difference seismic tomography with sparsity regularization, *J. geophys. Int.,* **199**(2), 944–955.

Fichtner, A., Trampert, J., Cupillard, P., Saygin, E., Taymaz, T., Capdeville, Y. & Villasenor, A., 2013. Multiscale full waveform inversion, *J. geophys. Int.,* **194**(1), 534–556.

Geman, S. & Graffigne, C., 1986. Markov random field image models and their applications to computer vision, in *Proceedings of the International Congress of Mathematicians,* 3–11 August 1986, Berkeley, CA, USA, Vols. **1 and 2**.

George, E.I., Makov, U. & Smith, A.F., 1993. Conjugate likelihood distributions, *Scand. J. Stat.,* **20**(2), 147–156.

Gou, R., Zhang, Y., Zhu, X. & Gao, J., 2023. Bayesian physics-informed neural networks for the subsurface tomography based on the eikonal equation, *IEEE Trans. Geosci. Remote Sens.,* **61,** doi:10.1109/TGRS.2023.3286438.

Kim, G., Ku, B., Ahn, J.-K. & Ko, H., 2021. Graph convolution networks for seismic events classification using raw waveform data from multiple stations, *IEEE Geosci. Remote Sens. Lett.,* **19,** 1–5.

Koller, D. & Friedman, N., 2009. *Probabilistic Graphical Models: Principles and Techniques,* MIT Press.

Li, S.Z., 1994. Markov random field models in computer vision, in *Proceedings of the Computer Vision—ECCV'94: Third European Conference on Computer Vision,* Stockholm, Sweden, 2–6 May 1994, Vol. II 3, pp. 361–370, Springer.

Li, S.Z., 2012. *Markov Random Field Modeling in Computer Vision,* Springer Science & Business Media.

Li, Y.E., O'Malley, D., Beroza, G., Curtis, A. & Johnson, P., 2023. Machine learning developments and applications in solid-Earth geosciences: fad or future?, *J. geophys. Res.,* **128**(1), doi:10.1029/2022JB026310.

Liu, H., Liu, C., Liu, Y., Zhang, Y. & Gao, F., 2013. Application of coherence technique based on Manhattan distance, *Global Geol.,* **32**(1), 144–151.

Liu, L., Ma, J. & Plonka, G., 2018. Sparse graph-regularized dictionary learning for suppressing random seismic noise, *Geophysics,* **83**(3), V215–V231.

McLachlan, G.J. & Krishnan, T., 2007. *The EM Algorithm and Extensions,* John Wiley & Sons.

Melas, D.E. & Wilson, S.P., 2002. Double Markov random fields and Bayesian image segmentation, *IEEE Trans. Signal Process.,* **50**(2), 357–365.

Mittal, A., Moorthy, A.K. & Bovik, A.C., 2012. No-reference image quality assessment in the spatial domain, *IEEE Trans. Image Process.,* **21**(12), 4695–4708.

Mu, H.-Q. & Yuen, K.-V., 2016. Ground motion prediction equation development by heterogeneous Bayesian learning, *Comp.-Aided Civil Infrastruct. Eng.,* **31**(10), 761–776.

Murphy, K., 2001. An introduction to graphical models, *Rap. tech,* **96,** 1–19.

Olsen, K., Archuleta, R. & Matarese, J., 1995. Magnitude 7.75 earthquake on the San Andreas Fault: three-dimensional ground motion in Los Angeles, *Science,* **270**(5242), 1628–1632.

Ortega, A., Frossard, P., Kovačević, J., Moura, J.M. & Vandergheynst, P., 2018. Graph signal processing: overview, challenges, and applications, *Proc. IEEE,* **106**(5), 808–828.

Poobathy, D. & Chezian, R.M., 2014. Edge detection operators: peak signal to noise ratio based comparison, *IJ Image, Graph. Sig. Process.,* **10,** 55–61.

Pulli, K., Baksheev, A., Kornyakov, K. & Eruhimov, V., 2012. Real-time computer vision with OpenCV, *Commun. ACM,* **55**(6), 61–69.

Sambridge, M. & Mosegaard, K., 2002. Monte Carlo methods in geophysical inverse problems, *Rev. Geophys.,* **40**(3), 3–1.

Shahrabi, M.A., Hashemi, H. & Hafizi, M.K., 2016. Application of mixture of Gaussian clustering on joint facies interpretation of seismic and magnetotelluric sections, *Pure appl. Geophys.,* **173,** 623–636.

Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A. & Vandergheynst, P., 2013. The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains, *IEEE Sig. Process. Mag.,* **30**(3), 83–98.

Tromp, J., Komatitsch, D., Liu, Q. *et al.*, 2008. Spectral-element and adjoint methods in seismology, *Commun. Comput. Phys.,* **3**(1), 1–32.

van den Ende, M.P. & Ampuero, J.-P., 2020. Automated seismic source characterization using deep graph neural networks, *Geophys. Res. Lett.,* **47**(17), e2020GL088690.

Van Houtte, P., Kanjarla, A.K., Van Bael, A., Seefeldt, M. & Delannay, L., 2006. Multiscale modelling of the plastic anisotropy and deformation texture of polycrystalline materials, *Eur. J. Mech.-A/Solids,* **25**(4), 634–648.

Wu, X. & Janson, X., 2017. Directional structure tensors in estimating seismic structural and stratigraphic orientations, *J. geophys. Int.,* **210**(1), 534–548.

Yang, J., Wang, Z., Lin, Z., Cohen, S. & Huang, T., 2012. Coupled dictionary training for image super-resolution, *IEEE Trans. Image Process.,* **21**(8), 3467–3478.

Yeh, T.-Y. & Olsen, K., 2023. Fault damage zone effects on ground motions during the 2019 Mw 7.1 Ridgecrest, CA, earthquake, *Bull. seism. Soc. Am.,* **113(4),** 1724–1738.

Zhang, H. & Ben-Zion, Y., 2024. Enhancing regional seismic velocity models with higher-resolution local results using sparse dictionary learning, *J. geophys. Res.,* **129**(1), e2023JB027016.

Zhang, H., Zhu, P., Liao, Z. & Li, Z., 2022a. SaltiSCG: interactive salt segmentation method based on CNN and graph cut, *IEEE Trans. Geosci. Remote Sens.,* **60,** 1–14.

Zhang, S., Deng, Q. & Ding, Z., 2022b. Multilayer graph spectral analysis for hyperspectral images, *EURASIP Journal on Advances in Signal Processing,* **2022**(1), 1–25.

Zhang, X., Lomas, A., Zhou, M., Zheng, Y. & Curtis, A., 2023. 3-D Bayesian variational full waveform inversion, *J. geophys. Int.,* **234**(1), 546–561.

Zhang, Z., Zhang, Y., Zhou, Z. & Luo, J., 2018. Boundary-based image forgery detection by fast shallow CNN, in *Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR),* pp. 2658–2663, IEEE.

Zhao, X., Curtis, A. & Zhang, X., 2022. Bayesian seismic tomography using normalizing flows, *J. geophys. Int.,* **228**(1), 213–239.

Zhou, Z., Bianco, M., Gerstoft, P. & Olsen, K., 2022. High-resolution imaging of complex shallow fault zones along the July 2019 Ridgecrest ruptures, *Geophys. Res. Lett.,* **49**(1), e2021GL095024.

Zhou, Z., Gerstoft, P. & Olsen, K., 2024a. Fusion of multi-resolution seismic tomography maps with physics-informed probability graphical models, in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* pp. 2540–2544, IEEE.

Zhou, Z., Gerstoft, P. & Olsen, K.B., 2024b. 3D multiresolution velocity model fusion with probability graphical models, *Bull. seism. Soc. Am.,* **114**(3), 1279–1292.

Zhu, L., Liu, E. & McClellan, J.H., 2015. Seismic data denoising through multiscale and sparsity-promoting dictionary learning, *Geophysics,* **80**(6), WD45–WD57.

Zuniga, O.A. & Haralick, R.M., 1987. Integrated directional derivative gradient operator, *IEEE Trans. Syst., Man, Cybernet.,* **17**(3), 508–517.