Seismic source characterization from GNSS data using deep learning

Giuseppe Costantino[1], Sophie Giffard-Roisin[1], David Marsan[1], Lou Marill [1], Mathilde Radiguet[1], Mauro Dalla Mura[2,3], Gaël Janex [1], Anne Socquet[1]

[1]Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, IRD, Univ. Gustave Eiffel, ISTerre, 38000 Grenoble, France
[2]Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France
[3]Institut Universitaire de France (IUF), France

Key Points:

- We develop deep learning approaches on synthetics mimicking the spatio–temporal structure of static deformation and realistic GNSS noise
- We design three deep learning models and we test them over three GNSS data representations
- Transformers and image time series of deformation can effectively characterise small deformation patterns associated with the seismic source

Corresponding author: Giuseppe Costantino, giuseppe.costantino@univ-grenoble-alpes.fr

Abstract

The detection of deformation in GNSS time series associated with (a)seismic events down to a low magnitude is still a challenging issue. The presence of a considerable amount of noise in the data makes it difficult to reveal patterns of small ground deformation. Traditional analyses and methodologies are able to effectively retrieve the deformation associated to medium to large magnitude events. However, the automatic detection and characterization of such events is still a complex task, because traditionally–employed methods often separate the time series analysis from the source characterization. Here we propose a first end–to–end framework to characterize seismic sources using geodetic data by means of deep learning, which can be an efficient alternative to the traditional workflow, possibly overcoming its performance. We exploit three different geodetic data representations in order to leverage the intrinsic spatio–temporal structure of the GNSS noise and the target signal associated with (slow) earthquake deformation. We employ time series, images and image time series to account for the temporal, spatial and spatio–temporal domain, respectively. Thereafter, we design and develop a specific deep learning model for each data set. We analyze the performance of the tested models both on synthetic and real data from North Japan, showing that image time series of geodetic deformation can be an effective data representation to embed the spatio–temporal evolution, with the associated deep learning method outperforming the other two. Therefore, jointly accounting for the spatial and temporal evolution may be the key to effectively detect and characterize fast or slow earthquakes.

Plain Language Summary

The continuous monitoring of ground displacement with Global Navigation Satellite System (GNSS) allowed, at the beginning of the 2000's, the discovery of slow earthquakes – a transient slow slippage of tectonic faults that releases stress without generating seismic waves. Nevertheless, the detection of small events is still a challenge, because they are hidden in the noise. Most of the methods which are traditionally employed are able to extract the deformation down to a certain signal–to–noise level. However, one can ask if deep learning can be a more efficient and powerful alternative. To this end, we address the problem by using deep learning, as it stands as a powerful way to automatize and possibly overcome traditional methods. We use and compare three data representations, that is time series, images and image time series of deformation, which account for the temporal, spatial and spatio–temporal variability, respectively. We train our methods on synthetic data, since real data sets are still not enough to be effectively employed with deep learning, and we test on synthetic and real data as well, claiming that image time series and its associated deep learning model may be more effective towards the study of the slow deformation.

1 Introduction

Global Navigation Satellite System (GNSS) is one of the reference sources of information in geodesy. Geodetic data can help analyze the ground displacement with millimeter precision as well as monitor its evolution through time (Blewitt et al., 2018). Such data is commonly used to monitor the ground displacement as a response to environmental (*e.g.*, tides, snow pack or hydrology), tectonic or seismic forcing, and to characterize the mechanical response of the Earth to these forcings. Notably, GNSS data has been widely used to study the deformation associated with the different phases of the earthquake cycle. This lead to a better understanding of the loading of faults between earthquakes, of the seismic ruptures studied with either static or kinematic approaches, and of the processes driving the post-seismic relaxation (Bock & Melgar, 2016; Bürgmann, 2018, and references therein). At the beginning of the 2000's, the discovery of slow slip events (Lowry et al., 2001; Rogers & Dragert, 2003; Dragert et al., 2001; Ozawa et al.,

2002) was made possible by the continuous monitoring of ground displacement with GNSS, and constituted a paradigm shift in the understanding of fault mechanics and earthquake physics. Since then, research on slow slip events has been very active. Several studies focus on one particular event or tectonic area, involving visual inspection of the data and dedicated modelling method with a fine-tuning of the parameters (Ozawa et al., 2001; Hirose & Obara, 2005; Wallace & Beavan, 2010; Radiguet et al., 2011; Ozawa et al., 2013; Radiguet et al., 2016; Socquet et al., 2017; Wallace et al., 2016; Wallace, 2020; Itoh et al., 2022; ?, ?). Another research direction aims at performing a systematic characterization of slow slip events (Michel et al., 2019; Frank & Brodsky, 2019; Takagi et al., 2019; Nishimura, 2014, 2021; Donoso et al., 2021; Okada et al., 2022; Rousset et al., 2017), with two underlying objectives: (1) construct catalogues of events allowing for an inter-event comparison and extraction of scaling laws (Ide et al., 2007; Gomberg et al., 2016), and (2) improve the signal over noise ratio in order to detect and characterize events that are at the limit of detection capabilities. The present work is in line with this latter approach. Our ultimate goal will be to develop a method able to systematically detect and characterize potential Slow Slip Events (SSEs) on active faults, including small ones, by taking advantage of the large availability of GNSS data. When looking for slow slip events, analyzing their static deformation signature in GNSS time series becomes mandatory since they are not associated with wave arrivals, and because longer time spans need to be considered, with respect to regular earthquakes. To this end, as the static deformation associated with regular or slow earthquakes can be approximated with a similar simple dislocation model (Okada, 1985), we use GNSS data to characterize the static deformation signature of earthquakes as a preliminary step towards the SSE analysis. Catalogues listing the source of all M_w earthquakes are made available by the routine analysis of seismic recordings by seismological agencies, allowing for a benchmarking with real GNSS data against an independent ground truth.

In this work we focus on the North Japan subduction, offshore Honshu, which is one of the seismic regions that is best instrumented in the world. In this area, slow slip phenomena are scarce compared to warmer subduction zones, such as Cascadia or South Japan, and its kinematics is essentially associated with regular earthquakes (Fukuda, 2018; Hirose et al., 2014). Therefore, new observations, such as seafloor data (Ito et al., 2013; Nishikawa et al., 2019), or new methods applied to terrestrial GNSS records (Nishimura, 2014, 2021; Rousset et al., 2017; Khoshmanesh et al., 2020, e.g.) or to seismic records (Marsan et al., 2013; Gardonio et al., 2018, 2019; Uchida et al., 2016, e.g.) are required to detect potential bursts of slow slip.

Machine learning and deep learning methodologies have recently been successfully applied to geosciences. In seismology, they have been used to address topics such as earthquake detection and phase selection resulting in seismic catalogues of unprecedented density (Zhu & Beroza, 2019; Mousavi et al., 2020; Ross et al., 2019; Kong et al., 2019; Zhu et al., 2019; Seydoux et al., 2020), earthquake early warning (X. Zhang et al., 2021; Münchmeyer et al., 2021; Saad et al., 2020), prediction of ground deformation (Kong et al., 2019; Mousavi et al., 2020), earthquake magnitude estimation (Mousavi & Beroza, 2020; Münchmeyer et al., 2020; Saad et al., 2020). However, machine learning techniques applied to the analysis of geodetic time series are less numerous. Relevant applications in the frame of the analysis of the slow slip events have been presented by Rouet-Leduc et al. (2019, 2020); Hulbert et al. (2019, 2020); He et al. (2020), with notable applications to InSAR data by Rouet-Leduc et al. (2021); Anantrasirichai et al. (2019). As we can remark from the literature, seismic recordings are still the main source of information for the analysis of surface ground movements, linked to either slow or regular earthquakes. Thus, this is another motivation to explore the potential of machine learning to analyse GNSS times series. We want to explore and test recent developments in machine learning applied to time series or image analysis, to be able to mine the geodetic data and characterize the events with a physics based approach.

In this paper, we address the problem of the fast seismic source characterization, *i.e.*, estimating the location and magnitude of a "regular" seismic event, based on deep learning applied to GNSS position time series. To the best of our knowledge, this is the first attempt of using machine learning–based techniques in such a direction. Again, this is not a goal *per se*, but must be rather seen as a first step towards the development of methods dedicated to the detection of slow slip events. We solve our problem as a regression in the framework of supervised learning, meaning that the input data used during the training are labelled. The data ground truth comes from seismic catalogs, serving as a benchmark for our analyses. We explore three different ways to represent GNSS data (time series, images, image time series) taking into account both the spatial coherency and the temporal variability of GNSS data. We associate a customized deep learning model to each data representation either by re-adapting already existing methods or by designing it afresh. Training and testing of the different methods is first made on synthetics. The performance of our methods is then evaluated against real GNSS data using an independent benchmark coming from actual earthquakes catalogs. The strengths and the pitfalls of the presented methods are discussed by envisioning some possible strategies to improve the results. The same analysis, applied to SSEs, would not be as straightforward, since SSE catalogs are still scarce. However, this methodology can serve as basis for further development.

2 Methods

2.1 Background work and positioning

2.1.1 Machine learning and deep learning methods for the seismic source characterization

In the frame of the source characterization, deep learning has proven to be particularly effective, as demonstrated by van den Ende and Ampuero (2020) and Münchmeyer et al. (2021), among the most recent works. As pointed out, a multi-station approach may more effectively locate the seismic source, in spite of other approaches using single-station waveforms, as (Mousavi & Beroza, 2020). Yet, combining observations from multiple stations is indeed a non-trivial task. It is possible to assign a weight to each (seismic or GNSS) station which depends on certain metrics, as done by Rousset et al. (2017), albeit addressing a different problem. van den Ende and Ampuero (2020) explicitly inject the location of each seismic stations in form of latitude and longitude coordinates, while Münchmeyer et al. (2021) employ a sinusoidal embedding (*i.e.*, the position is encoded through sinusoidal functions (Vaswani et al., 2017)) for the station locations, outperforming already existing methods and showing promising results in terms of earthquake early warning and source characterization. Nevertheless, as a general remark, no straightforward guideline is available to effectively take both the temporal and the network geometry into account. Therefore, exploiting the spatial distribution is indeed a key problem which we are willing to address in this work.

2.1.2 Followed approach

An overview of the proposed methodology is shown in Figure 1. As any standard machine learning model, the pipeline consists in a training and an inference phase. During the training process, a model is provided with data to learn from. In case of supervised learning, a couple $\langle$ input, desired output $\rangle$ is presented to the model, which *learns* by minimizing a certain error metric between the estimated output and the desired output, which serves as a reference. We use epicenter position and the magnitude of the event as a target output for the characterization, with GNSS data as input. In the inference phase, the trained model is used to make predictions on new data. We will test our methods both with synthetic and with real data. We provide new input data to the trained model and we compare the outcomes with the reference outputs, *i.e.*, the epicenter po-

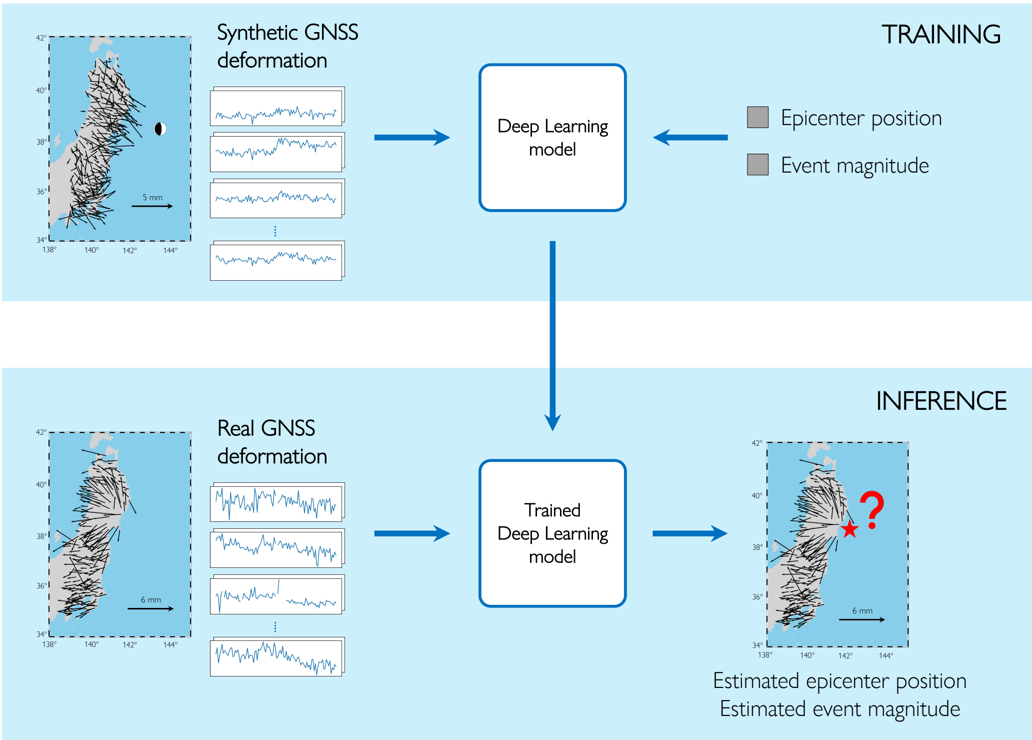

Figure 1. Schema of the workflow, summarizing the training and the inference phases. A given deep learning model is trained by providing an input and a desired output. Here we use GNSS data as input and a couple consisting of $\langle$ epicenter position, magnitude $\rangle$ for each event. During the training process, the model will learn a nonlinear function to map GNSS inputs to an approximation of its position and magnitude. Once trained, this model can be used to perform tests on new, independent data. Here we train on synthetic data and we test both on synthetic and real data.

sition and the event magnitude associated to this new input data. Training our models with supervised learning applied to earthquakes allows us to benefit from a benchmark coming from real earthquakes catalogs.

We make use of synthetic data to train and validate our deep learning models and we test on synthetic and real data afterwards. Japan is probably one of the best instrumented regions in the world, with GNSS data among the cleanest and the densest ones. Yet, we did not train our models with real data for the following two main reasons.

1. GNSS data suffers from the presence of data gaps and missing stations. They can be associated to station inactivity (*e.g.*, electricity blackouts) or to inconsistent daily measurements, for example due to large earthquakes. Moreover, the number of GNSS stations may evolve over time, due to the installation of new receivers or to the temporary unavailability of certain ones. It can moreover make it hard to collect regular and well formatted subsets of data to train on. This drastically reduces the number of exploitable training samples, which is indeed a key issue when training deep learning models (LeCun et al., 2015).
2. Real data is not uniformly distributed in terms of source parameters, most notably position and magnitude. Since we are dealing with subduction events, most of the actual epicenters will be located on the subduction interface. This can constitute a limitation since a deep learning model trained on such a configuration might not generalize well for events which would be located inshore or sufficiently far from the training area. In addition, the magnitude distribution follows the Gutenberg–Richter scaling law (Gutenberg, 1956). As a consequence, the deep learning methods would be biased because of the small magnitude events, which will be more numerous, thus possibly resulting in worse performance on the larger ones. To this end, we generate synthetic ruptures whose source parameters are assumed to be random variables drawn from an uniform distribution.

By employing synthetic data, it is possible to generate as many samples as needed, overcoming the lack of data and exploiting the features of deep models. Nonetheless, the resemblance between the synthetic data and the real one plays a critical role, since it will have an impact on how well the deep learning model will perform on real data: we need to generate ultra realistic time series. To this end, we add realistic noise computed from actual GNSS data, as it will be detailed in section 2.2.

2.2 Generation and representation of synthetic data

We generate synthetic data samples as the sum of a modeled displacement signal and a realistic noise sample. We rely on three data representations both for synthetic and for real samples and we associate each of them to a different deep learning model. More formally, the synthetic data set is represented as a set of N couples $\{\mathbf{x}_n, \Theta_n\}_{n=1}^{N}$, with Θ a set of source parameters (epicenter position, magnitude, focal mechanism, etc.) and $\mathbf{x}$ being the data following an additive model:

$$\mathbf{x} = \mathbf{s} + \boldsymbol{\varepsilon} \tag{1}$$

with $\mathbf{s}$ the synthetic signal (cf. section 2.2.1) and $\boldsymbol{\varepsilon}$ the noise term (cf. section 2.2.2).

2.2.1 Synthetic displacement

We obtain the synthetic displacement signals $\mathbf{s}$ by relying on Okada's dislocation model (Okada, 1985). The model input parameters are generated as follows. Earthquake hypocentral positions (longitude, latitude, depth) are assumed to be uniformly distributed random variables, with longitude $x \sim \mathcal{U}(139°, 146°)$, latitude $y \sim \mathcal{U}(35°, 41°)$ and depth

$d \sim \mathcal{U}(2 \text{ km}, 100 \text{ km})$. Event magnitudes are generated as $m \sim \mathcal{U}(5.8, 8.5)$ and static moments M_0 are computed accordingly, as (Hanks & Kanamori, 1979):

$$M_0 = 10^{1.5m+9.1} \text{ N} \cdot \text{m} \tag{2}$$

Fault azimuth direction ϕ_S (strike), dip angle δ and slip angle λ (rake) are constrained to a thrust focal mechanism, by allowing for a certain variability of fault slip combinations: $\phi_S \sim \mathcal{U}(160°, 240°)$, $\delta \sim \mathcal{U}(20°, 30°)$, $\lambda \sim \mathcal{U}(75°, 100°)$. Static stress drop $\Delta\sigma$ is assumed to be a lognormal random variable with an average value of 3MPa and a standard deviation of ± 30 MPa. A circular crack is assumed with radius R computed as (Aki & Richards, 2002):

$$R = \left(\frac{7}{16}\frac{M_0}{\Delta\sigma}\right)^{1/3} \tag{3}$$

which can be used to approximate a rectangular dislocation, having length L and width W, by imposing the equality of the surfaces:

$$\pi R^2 = L \cdot W \tag{4}$$

The fault aspect ratio is assumed such that the fault length L and width W satisfy: $W = L/2$, with L computed as $L = \sqrt{2\pi}R$. It should be noticed that the dislocation surface does not change as a function of the aspect ratio between L and W. The average slip $\bar{u}$ is also derived for a circular crack and it is computed as (Aki & Richards, 2002):

$$\bar{u} = \frac{16}{7\pi}\frac{\Delta\sigma}{\mu} \tag{5}$$

with μ the shear modulus, assumed equal to 30 GPa.

Okada's dislocation model is applied to each one of this set of earthquake sources to compute the predicted synthetic displacement at each GNSS station in Honshu from the Earth Observation Network System in Japan (GEONET). Hence, the theoretical deformation field at all station locations in Honshu is obtained for each dislocation setting.

2.2.2 Realistic noise computation

Noise in GNSS time series constitutes one of the most critical issues, as it is spatially and temporally correlated (Ji & Herring, 2013; Dong et al., 2002). Here we define noise as everything which is not the signal of interest, being the coseismic signal offsets. At first approximation, its spectrum can be represented as a white noise at the lowest frequencies, and a colored noise having a $1/f^\kappa$ decay starting from a certain corner frequency, with the spectral index κ being usually fitted from the highest frequencies of the periodogram (Williams et al., 2004; J. Zhang et al., 1997; Mao et al., 1999). The spatial distribution of such a noise is not random. On one hand, some common patterns must be found among near stations, therefore it can be helpful to discriminate noise from other types of signals. On the other hand, making this type of analysis is difficult, because of the unpredictability of those spatial patterns as well as the intrinsic difficulty in handling such topological consistency in a consistent manner.

Realistic perturbations, *i.e.*, noise, are needed to mimic real displacement data. Here we rely on realistic noise samples computed from real GNSS time series by following an existing approach for surrogate data generation (Schreiber & Schmitz, 1996; Prichard

& Theiler, 1994). By removing known signals (e.g. earthquakes, postseismic relaxation, SSEs, jumps associated with antenna changes etc) from GNSS time series from a quadratic trajectory model (Marill et al., 2021), we obtain GNSS residual time series that contain the noise that we want to mitigate. Then, a Principal Component Analysis (PCA) is performed on 100–days windows, by taking into consideration all the stations at the same time. Afterwards, a Fourier Transform (FT) is applied and the phase spectrum is randomized by picking a new phase $\varphi \sim \mathcal{U}(0, 2\pi)$. The same shuffling sequence is adopted for the whole network in order to preserve the spatial coherency between stations. After this process, an Inverse FT and an Inverse PCA are performed. As a result, the transformed noise samples $\boldsymbol{\varepsilon}$ will have, on average, the same spatial covariance. Moreover, we can build new noise samples by randomizing the phase, since the Power Spectral Density (PSD) of the transformed samples and the actual ones will be asymptotically equivalent.

2.2.3 GNSS data representations

We build three data types: time series, images and image time series. The raw data come in the form of time series. Then, we derive images to take the spatial information into account, and image time series to take advantage of both the time and space patterns. A schematic view is provided in Figure 2. Moreover, here we do not aim at estimating the hypocentral depth, therefore we do not consider the vertical component of GNSS data, as it does not provide any additional constraint.

Time series. We build synthetic position time series by considering a noise window of 100 days (cf. section 2.2.2). We add a Heaviside step to simulate the coseismic displacement (Bevis & Brown, 2014), with the onset time (cf. t_c in Figure 2) being at the center of the window. The step amplitude for each station depends on the modeled displacement (cf. section 2.2.1). More formally, the time series structure is represented by a tensor $\mathbf{X} \in \mathbb{R}^{L \times T \times 2}$, with L the number of stations and T the number of time steps, the location (latitude, longitude) of the station being given by $\mathbf{S} \in \mathbb{R}^{L \times 2}$.

Differential images. Images of interpolated deformation field are computed as follows. By assuming the coseismic onset at time t_c, we consider the difference between the displacement at time $t_c + 1$ day and $t_c - 1$ day, namely the differential coseismic displacement field for each station in the GNSS network. We interpolate the deformation field in space as follows. We first employ a median anti-aliasing filter with a grid spacing of 25 arc minutes (≈ 45 km), then we interpolate the points in space by using adjustable tension continuous curvature splines (with tension factor $T = 0.25$) (Smith & Wessel, 1990). The resulting image dimensions are $76 \times 36 \times 2$ pixels. Afterwards, we mask the sea by forcing to zero all the offshore pixels, in order not to extrapolate offshore, which may degrade the performance of the deep learning methods. Mathematically, the differential images are obtained by rasterizing for a given time step t_c an image as a tensor $\mathbf{D} \in \mathbb{R}^{I \times J \times 2}$ being $I \times J$ the resolution of the image $\mathbf{D}$ and $\mathbf{D}(\mathbf{S}(k)) = \mathbf{X}(k, t_c + 1) - \mathbf{X}(k, t_c - 1)$ with $\mathbf{S}(k)$ the position (latitude, longitude) of the k-th station and t_c the time of the coseismic offset. The value of I and J, as well as the content of the pixels $\mathbf{D}(\mathbf{S}(k))$, for $k \notin \mathbf{S}$, have been described before.

Image time series. Image time series are built from position time series by interpolating the position information at each frame with the same approach employed for the differential images. We consider 15 days of data, with the first 7 frames corresponding to the week before the coseismic displacement, the central frame corresponding to the coseismic offset, and the remaining 7 days corresponding to the week after the coseismic. Each frame of the image time series has dimensions $76 \times 36 \times 2$ pixels. Formally, an image time series is represented by tensor $\mathbf{T} \in \mathbb{R}^{M \times I \times J \times 2}$, with M the length of the image time series and $\mathbf{T}(t_i, \mathbf{S}(k)) = \mathbf{X}(k, t_c + i)$, $i \in (-\lfloor \frac{M}{2} \rfloor, ..., 0, ..., \lfloor \frac{M}{2} \rfloor)$.

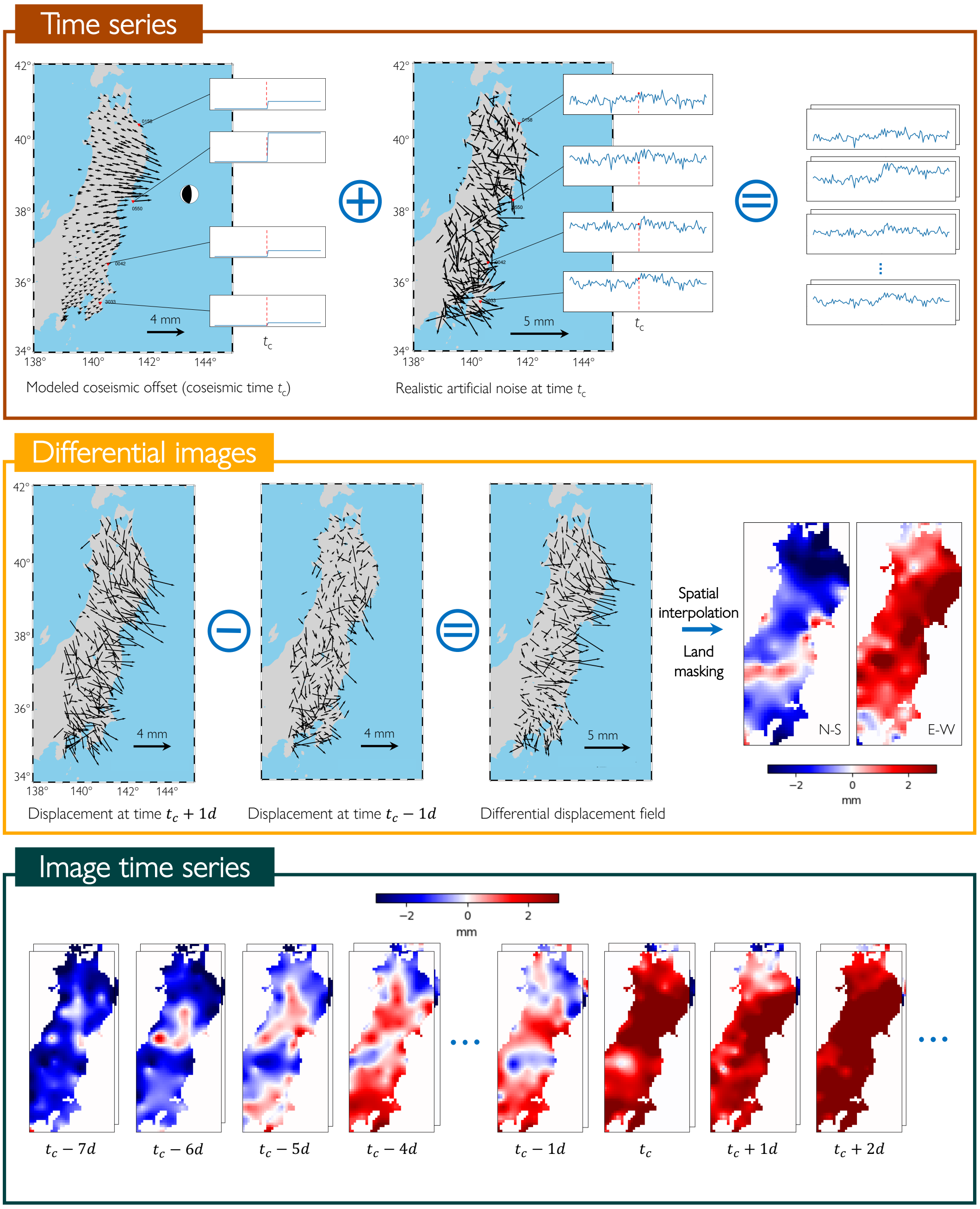

Figure 2. Outline of the three designed data representations. Each arrangement is designed for a specific deep learning model (cf. Figure 3 with corresponding colors). The data–arrangement procedure is shared between synthetic and real data, except for time series, which are directly available from GNSS recordings. **(time series)** is associated to the TS model. Synthetic position time series are built by adding a modeled signal (cf. section 2.2.1) to a realistic noise time series (cf. section 2.2.2) by imposing the time of the coseismic offset to be at the center of the window (cf. section 2.2.3). **(differential images)** is associated to the IMG model. Differential images of ground deformation are built by differentiating the GNSS displacement at the day following the coseismic time and the day before. Then, the differential deformation field is interpolated in space for each direction. **(image time series)** is associated to the TRA model. Image time series are the 3D–equivalent of position time series. A total of 15 days of deformation is collected, by selecting the week before and the week after the coseismic offset (included). For each day, a spatial interpolation is performed by employing the same method as for differential images to produce a couple of images representing a frame in the whole time series.

In all the three representations, we consider that the coseismic offset time t_c is known. Indeed, we focus here only on the characterization part, and not the detection.

2.3 Employed deep learning methods

We developed a deep learning method specifically designed for the characteristics of each chosen data representation. We designed three methods by adapting different state-of-the-art methods that were not originally designed for geodetic data, in order to best address our specific problem. A graphical outline of the methods is provided in Figure 3.

2.3.1 Time–series based CNN (TS)

Time series can be effectively processed by Convolutional Neural Networks (CNNs), extracting succinct information coming from temporal domain, as reviewed by Bergen et al. (2019); Kong et al. (2019). Here we rely on the architecture proposed by van den Ende and Ampuero (2020), originally proposed for seismic data. Their model has been selected as a potential candidate as it presents several interesting features that can be leveraged also when dealing with geodetic data. The first portion of their network consists in three convolutional blocks with an increasing number of feature maps. In each block, three convolutional layers are used for the feature extraction, followed by a max–pooling layer, employed for subsampling the data. Afterwards, the coordinates of every station associated with an input waveform are injected into the model, as taking into account the location of seismic stations can improve the performance, which is the key characteristic of the model. The max–reduce strategy helps in aggregating the features related to the stations, in order to select the feature from the station corresponding to the most relevant contribution for the prediction. We exploited these features in our re-elaboration. Moreover, in order to further mitigate the vanishing gradient problem, the rectified linear unit (ReLu) (Agarap, 2018) activation function has been chosen for the hidden layers. Since the output variables are uniformly distributed, such an activation function would not squash the predictions in the boundaries of the output range, possibly making the model more flexible when predicting patterns laying outside of the ranges used in the training process, notably when testing for very small or large magnitude earthquakes. The injected horizontal coordinates (latitude, longitude) of GNSS stations are previously scaled in $[0, 1]$. The original model is also equipped with weights associated to the waveforms accounting for inactivity or missing data from a station. We set them to 1 as the GNSS network in Japan is quite dense and all the stations in synthetic data were assumed to be functioning. Yet, it can represent a further useful development, as it will make the model more flexible when testing on actual data as well as testing against other regions. A visual summary of the model is outlined in the first box of Figure 3.

2.3.2 Image–based CNN (IMG)

We use a 2D CNN to analyze and extract features from interpolated deformation images. They are an effective solution to leverage the spatial coherency and covariance of data structured as images (LeCun et al., 2015) and have become one of the reference architectures for image–based tasks (Goodfellow et al., 2016), also with relevant applications in the geosciences (Rouet-Leduc et al., 2020; Anantrasirichai et al., 2019).

Here we rely on the architecture of MobileNetV2 (Sandler et al., 2018) as the feature extractor. This particular architecture has been chosen as it is lighter (in terms of the number of parameters) with respect to other state–of–art models, such as the VGG family (Simonyan & Zisserman, 2014). Yet, it presents some interesting features, such as the linear bottleneck layers and the depth–wise convolutions. The architecture presents a first convolutional layer followed by seven bottleneck layers. These layers perform an efficient convolution by relying on point–wise and depth–wise convolutions, presenting

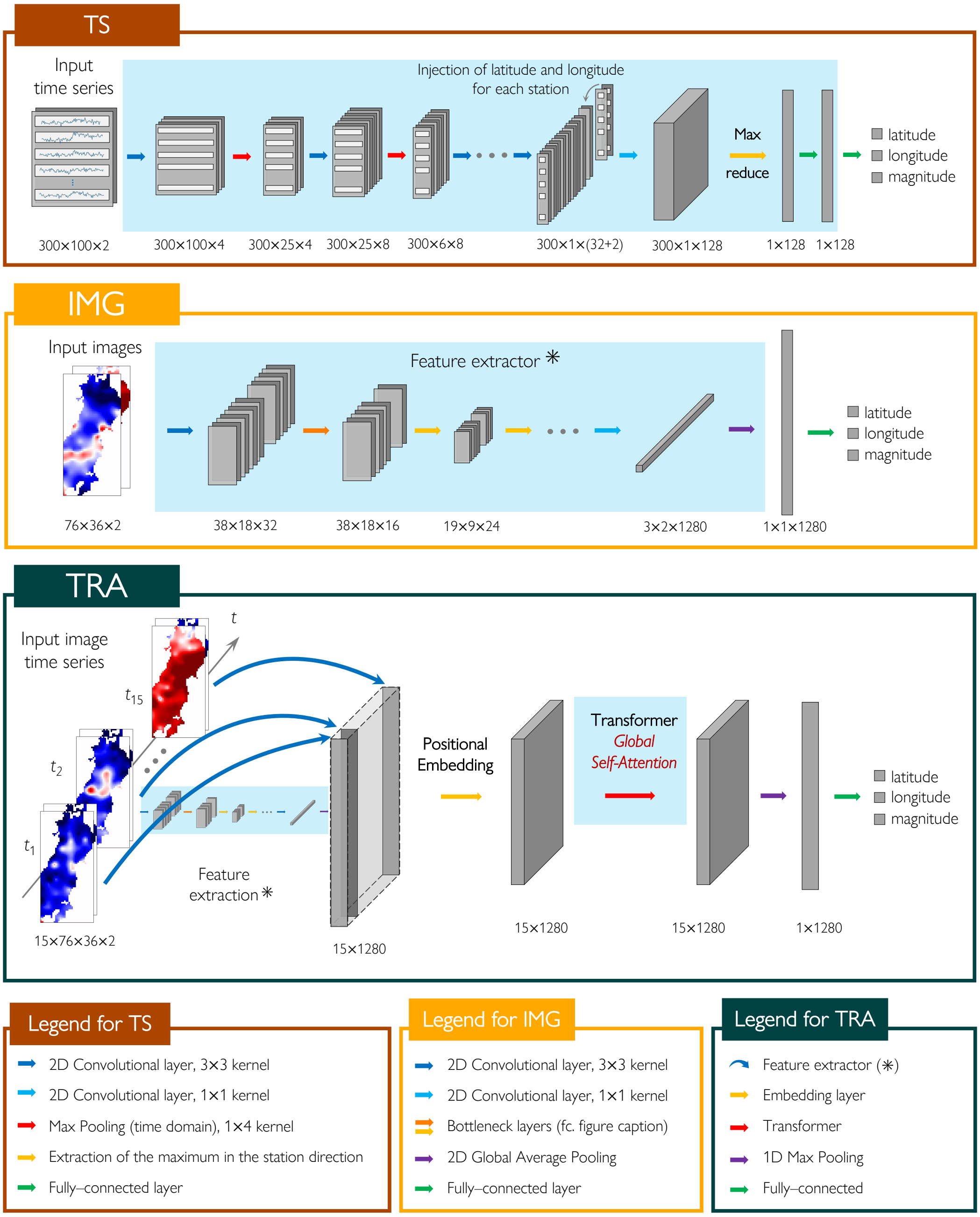

Figure 3. The three reference deep learning methods designed in this work. Shaded cyan rectangles represent existing state-of-the-art models. Such models have been slightly modified or adapted, where specified (cf. section 2.3). Further details, such as dropout layers, stride and activation functions, have not been depicted to facilitate the reading. Arrows represent the layers operating between the input (left) and the produced output (right). **(TS)** The network progressively computes features from convolutions and downsamplings in the time dimension. The latitude and longitude information is then injected. The resulting 2D–array is finally expanded and the contribution coming from the most informative GNSS station is taken (*max–reduce* operation in yellow). Model readapted from (van den Ende & Ampuero, 2020). **(IMG)** was inspired by the MobileNetV2 architecture (Sandler et al., 2018). The input two–channel image is processed with convolutions and downsamplings by employing bottleneck layers (cf. section 2.3.2) with and without residual connections (orange and yellow arrows, respectively). **(TRA)** The first part of the network exploits the feature extractor of IMG to compute spatial features for each frame, which are packed in a 2D–array. Then, a positional embedding enforces time sequencing and prepares the intermediate–level data for the sequential analysis performed by the Transformer (self–attention as in Mousavi et al. (2020)).

residual connections when there is not any stride in the convolutions. We use a global average pooling strategy after the feature extractor. A scheme of the architecture is provided in the second box of Figure 3.

2.3.3 Image time series–based Transformer (TRA)

Image time series–based approaches are required to account for both the spatial and the temporal variability into the input data. Deep sequence models such as LSTM (Long-Short Term Memory) or GRU (Gated Recurrent Unit) have been successfully used in geosciences to exploit the sequential behaviour of the data (Bergen et al., 2019; Wang et al., 2017), as well as Transformers, which have overcome the former becoming the reference methods in the state–of–art (Vaswani et al., 2017; Münchmeyer et al., 2021; Mousavi et al., 2020). We tested both the LSTM and the Transformer approaches and we chose the latter, whose complexity is justified by its better ability to constrain the spatio–temporal evolution.

Here we design a relatively simple model to validate to consider both spatial and temporal features jointly, which can serve as a baseline to add more complexity in the future. We first use a feature extractor to compress the input data dimensionality to obtain a reduced representation. We use the same architecture of the IMG feature extractor and we distribute it in time, *i.e.*, we use the same feature extractor for each frame of the image time series. As a result, we obtain a feature vector for each frame of the image time series. Afterwards, we stack all the feature vectors in one matrix to be exploited by the Transformer layer, as shown in the third box of fig 3. Since the self–attention is, in general, order agnostic, we apply a Positional Embedding layer to ensure that the relative position of the frame information is correctly enforced (Chollet, 2021). We chose not to have a fixed mapping, therefore the embedding weights are learnt during the training phase. After the embedding layer, we use a Transformer equipped with additive self–attention, as in (Mousavi et al., 2020). For simplicity, we use only one global self–attention. According to our preliminary tests, the performance is not considerably increasing when adding a second level of attention, possibly because our model is still too simple to benefit from a hierarchical attention structure. After the self–attention, we apply another dropout (dropout rate 0.5) layer (cf. section 2.3.1) followed by a one dimensional Global Max Pooling. As a final remark, we train the model by enforcing the feature extractor to evolve from weights already learnt by IMG. Therefore, we apply a sort of fine–tuning which may be beneficial for the self–attention to reach some acceptable parameter configurations in the early stage of the training already. The TRA architecture is presented in the third box of Figure 3.

2.4 Implementation and training details

We enforced the mean squared error (squared $L2$ norm) as loss function, *i.e.*, the objective function which is minimized during the training, defined as follows:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{N}\frac{1}{d}\sum_{i=1}^{N}\sum_{j=1}^{d}(y_{i,j} - \hat{y}_{i,j})^2 \tag{6}$$

where $\mathbf{y} \in \mathbb{R}^{N\times d}$ and $\hat{\mathbf{y}} \in \mathbb{R}^{N\times d}$ represent the ground truth and the predicted output, respectively, with N being the number of observations and d the number of dimensions. Notably, $d = 3$, being latitude, longitude and magnitude the output variables. Hence, the loss function jointly minimizes the error on both position and magnitude. Since the ranges of the output variables are not comparable, they are first scaled in $(0, 1)$. Thanks to this transformation, the high–range variables do not prevail on the others, possibly masking small variations on low–magnitude variables. As a result, the loss minimization turns out to be more regular and effective.

Table 1. Quantitative results of the tested methods.

Model	Position error (km)	Magnitude error (°)
TS	156.46 ± 116.94	0.26 ± 0.20
IMG	133.07 ± 146.97	0.18 ± 0.18
TRA	$\mathbf{105.44 \pm 128.84}$	$\mathbf{0.13 \pm 0.15}$

All the three models have been provided with a last fully–connected layer with three outputs and a linear activation function (linear combination). Since the output variables are uniformly distributed, such an activation function would not squash the predictions in the boundaries of the output range, possibly making the model more flexible when predicting patterns laying outside of the ranges used in the training process. Thereafter, we enforce a dropout regularization (Srivastava et al., 2014) in this final layer (dropout rate 0.5) at training time, which helps prevent the models from overfitting the training data, in addition to the dropout regularization which may already be enforced throughout the previous layers.

We performed the training of the three models by adopting a mini–batch stochastic gradient learning (Bottou et al., 2018) with a batch size of 128 samples and the ADAM method (Kingma & Ba, 2014) for the optimization. The learning rate was chosen according to a grid–search optimization and the best value was found at 0.001. We initialize all the network weights with an orthogonal initializer (Saxe et al., 2013) for TS and with a uniform Xavier initializer (Glorot & Bengio, 2010) for IMG and TRA.

We employ twenty thousand synthetic samples that we divide it into training, validation and test sets with proportions of 60%, 20% and 20% respectively. We used the training and validation sets for the training phase. When the loss on the validation set is not decreasing anymore in a certain number of training steps, the training is terminated and the model's weights are loaded with the ones associated to the best loss value. Moreover, the validation set has been employed to tune the hyper–parameters of the models (such as the learning rate, the best architecture, etc.) in order to prevent any overfitting. The test set is used for the final inference and for the performance analysis.

The code was implemented in Python using the Tensorflow (Abadi et al., 2016) library as well as the higher–level package Keras (Chollet et al., 2015). The training was run on NVIDIA Tesla V100 Graphics Processing Units (GPUs).

3 Results on synthetic data and discussion

We first evaluate the performance of the three models on a synthetic test set, independent of the training and validation ones. In order to concretely compare the three methods, the synthetic and real data sets under consideration are the same for all the models and differ only in their input representation.

Table 1 shows quantitative results in terms of average error and standard deviation for the three models with respect to the synthetic test set.

The position error is assumed as the Euclidean distance and is computed for each sample as:

$$E_p^i = \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2} \tag{7}$$

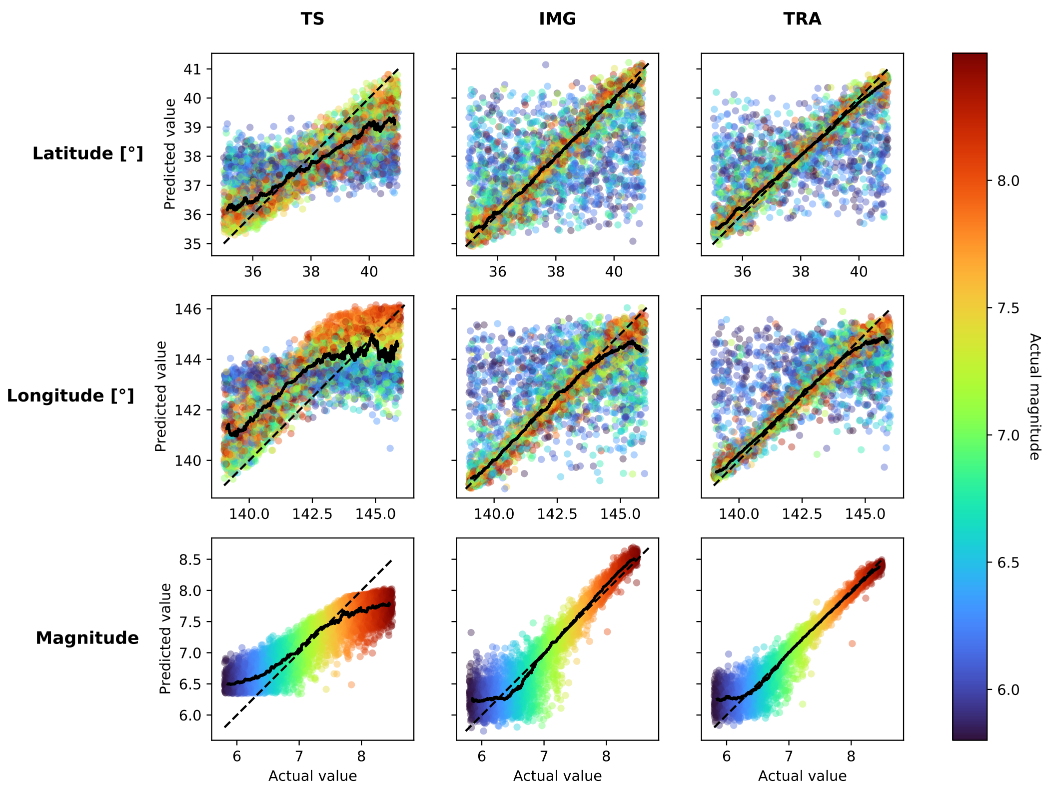

Figure 4. Comparison of the performance of the tested models at inference time. TS, IMG and TRA models are shown on columns respectively. For each row, latitude, longitude and magnitude predictions are reported, respectively. Each point of the scatter plots represents a test sample, whose magnitude is indicated by the colorbar, and it is illustrated as a function of both its actual and predicted value. Black dashed lines represent the ideal prediction, while solid black lines represent the rolling median.

where x_i and y_i represent the actual longitude and latitude and $\hat{x}_i$ and $\hat{y}_i$ the predicted longitude and latitude, respectively. We adopt a Mean Absolute Error (MAE) for the magnitude, which is computed for each sample as:

$$E_m^i = |m_i - \hat{m}_i| \tag{8}$$

where m_i and $\hat{m}_i$ are the actual and predicted magnitude, respectively. Then, the total position and magnitude errors are computed by averaging E_p^i and E_m^i.

The quantitative results evidence that the TRA method outperforms the other two, in terms of average error, both in position (105.44) and in magnitude (0.13), with a lower standard deviation in position (128.84) with respect to the IMG method (146.97) and slightly higher with respect to the TS method (116.94). We may expect TRA to have also a lower standard deviation, yet it depends on many factors which can be related to the type of data used, as well as randomness in the training.

3.1 Analysis of the performance

Figure 4 shows the prediction of the three models on the synthetic test set color–coded by the actual magnitude of the test events. Indeed, the performance of all the models depends on the magnitude, which is closely related to the Signal-to-Noise ratio (SNR). As we can observe in the third row, low magnitudes tend to be overestimated by all models, likely because there is an intrinsic resolution threshold preventing the models from achieving good performance when the SNR is not sufficiently high. For the lower magnitude events (blue points), also the localization ability is poor, as the predictions of the three models do not follow, in general, the ideal prediction line. This behaviour may thus be linked to an intrinsic limitation of data information.

The solid black lines in the plots show the rolling median on the scatter plot computed on 150 samples. They give the general trend of the predictions. At first order, they can help in individuating a tentative magnitude threshold value, that is the value under which the magnitude prediction is significantly degraded. We can derive the following resolution limits: $M_w^{TS} = 7$, $M_w^{IMG} = 6.3$, $M_w^{TRA} = 6.2$. Those quantities have to be taken as a general indication.

The trend of the magnitude prediction for TS deviates from the ideal prediction line both for small and for large magnitudes, presenting a median saturation around M_w 6.5 and M_w 7.8, respectively (cf. black solid lines). The saturation for high magnitudes could be due to the employed network architecture as well as to specific features associated to the type of data. The magnitude prediction for the image–based methods, *i.e.*, IMG and TRA, better adhere to the ideal prediction line, with a progressively smaller error variance at larger magnitudes, in line with the SNR improvement. As for the magnitude resolution, TRA is the method associated with smaller error variances and with a better median trend.

From the latitude and longitude prediction, *i.e.*, the localization performance, we can observe that the models do not treat similarly the low and high magnitudes. Notably, for magnitudes smaller than the SNR limit, TS assigns them an average position (*i.e.*, near 38 for the latitude and 142 for the longitude). This behaviour is clearly indicated by the horizontally–clustered blue points. This pattern is indeed coherent with the choice of the quadratic loss function used to train the model. In fact, at first order, the best guess is represented by the mean value of the output range subject to the posterior distribution (Haykin, 2008; Moon & Stirling, 2000). We can derive that, when the SNR is below a certain resolution threshold, the model associates low–magnitude events to average coordinates, which likely minimize the average error. For higher magnitudes, the TS latitude predictions are more clustered around the ideal prediction line, although a tendency towards the mean values is still present, while TS predicts the longitude of high–magnitude events either in the proximity of the GNSS network (longitudes less than ~ 142) or in far field (longitudes higher than ~ 144). Conversely, image–based methods characterize low–magnitude events as having a random position in the region of interest (cf. scattered blue points), while being able to precisely constrain higher magnitude events, with predictions tightly clustered around the ideal line. Moreover, the median prediction lines for IMG and TRA are more stable and significantly bends only in correspondence of far field events (longitudes near 145°), which is physically consistent.

3.2 Spatial variability of the location error

Figure 5 shows the location error as a function of the ground truth spatial coordinates. The plot has been computed by interpolating the location error for each test data sample onto a grid, corresponding to the area of interest. This smoothed heatmap indicates the amount and the distribution of location errors all over the tested region, for different magnitude ranges. This type of representation can help to assess the physical consistency of the tested models, as well as revealing systematic biases in the error

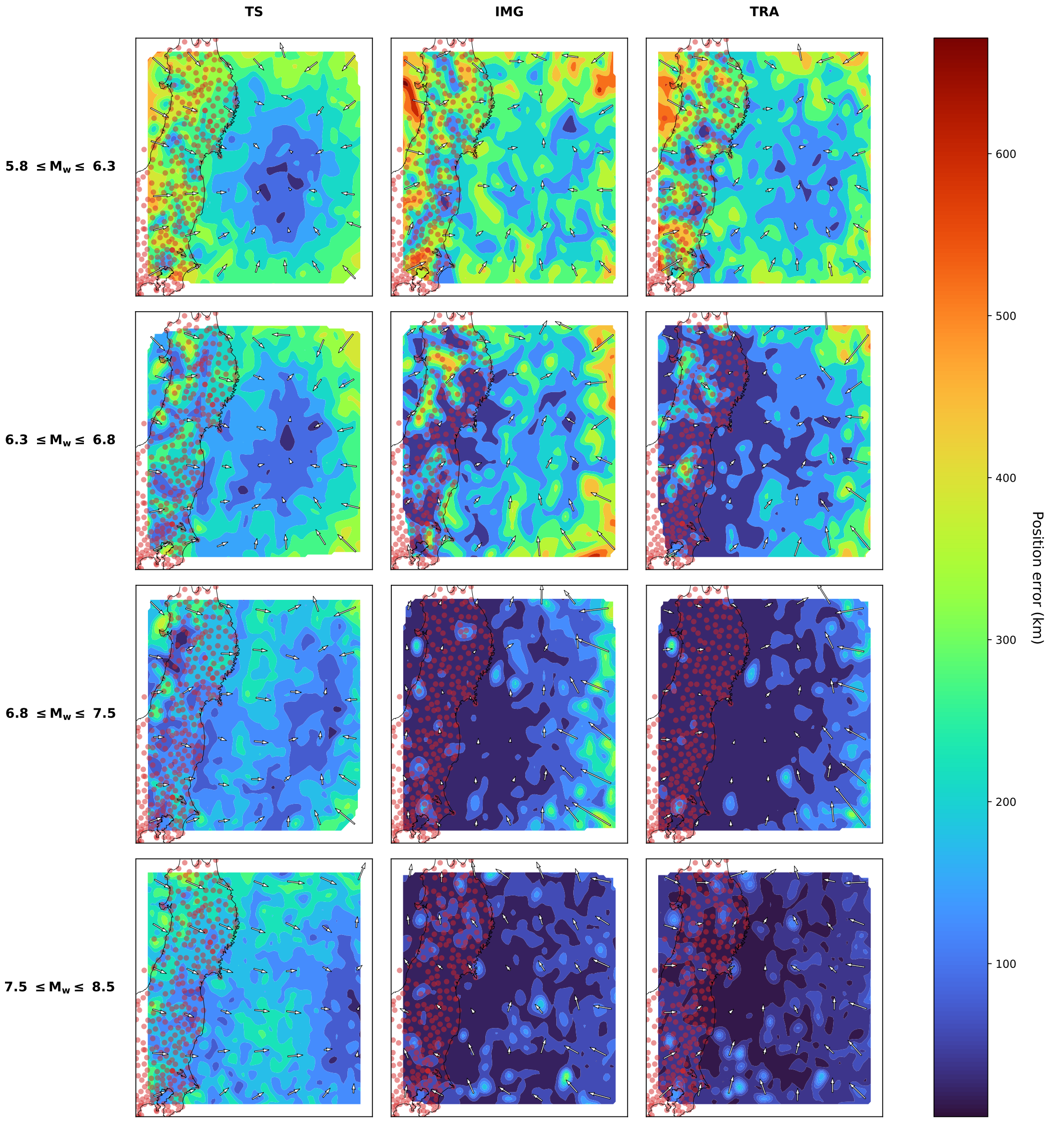

Figure 5. Comparison of the location error of the tested models, reported in the columns. Each subplot shows the location error associated to the test samples, interpolated on a grid which corresponding spatial coordinates are indicated along the axes. Magenta data points represent the position of GNSS stations in Japan. The heatmap depicts the distribution of the error in position committed by the tested models, for different magnitude ranges, in rows. Arrows show the average direction of position error for patches of 1 × 1 arc degree. The arrows have the same scale throughout all the subplots, making a comparison be possible among different models.

pattern, which can be more evident for specific magnitude ranges. Moreover, it is useful to compare and discuss how the error distribution of certain events can be influenced by their relative position with respect to the GNSS network.

From the heatmaps of the first two lines, corresponding to the magnitude ranges $(5.8, 6.3)$ and $(6.3, 6.8)$, we can see how the three methods handle the characterization of low–magnitude events (cf. 3). We can remark that TRA is able to better resolve small magnitude events in near field (*i.e.*, in proximity of the GNSS network). By increasing the magnitude range, the error amplitude of IMG and TRA are decreasing, affecting only the points which are far from the network (on the east side). For high magnitudes, TS tend to localize most events in far field, seemingly not taking advantage of the strong GNSS signal in the near field.

The error pattern for image–based methods is, therefore, more physically consistent. The most reasonable explanation is that image–based models can better capture the spatial information by extracting spatial features which are essential for the characterization. As a general comment, we do not see any clear bias and the error patterns exhibit a correct behaviour, since, as the magnitude increases, highest errors are pushed towards the far field. We notice that, for low magnitudes, the maximum error associated to the TS is about 200 km less than the other models, as its bias correctly minimizes the average error, yet without providing any discriminant ability to the model. By increasing the magnitude, errors become smaller and smaller, with the events contributing to the largest errors being distributed on the east (offshore) side, in favor of TRA, which is associated with the most reasonable error pattern.

3.3 Influence of the distance from the GNSS network on the predictions

Figure 6 helps us in analyzing the dependency of errors to the relative position with respect to the GNSS network. Each scatter plot represents the error as a function of the distance to the nearest GNSS station. Such a distance is computed from the coordinate of a hypocenter as the 3D Euclidean norm, in order not to take into account the Earth curvature. This kind of representation is effective in revealing patterns of the position and magnitude errors as function of both distance, on the x axis, and magnitude, in color code.

In order to better summarize and understand this behaviour, we identified three regions, according to the relative distance to the nearest station: being d the distance to the nearest station, we will refer to near, intermediate and far field when $d \leq 0.5$, $0.5 \leq d \leq 3$ and $d \geq 3$ arc degrees, respectively (see dashed lines in Figure 6). The dashed lines correspond to the median for several magnitude ranges (cf. Figure 5).

For the TS model, we can see in the first row a non negligible presence of errors due to high magnitude events both in near field and intermediate field, while image–based methods being able to correctly locate a larger number of high and even low magnitude events. Looking at the magnitude estimation (second row), we can observe for TS a cluster of errors corresponding to very high magnitude events in near field in the upper part (average error $M_w = 0.8$), and a second cluster of errors associated to lower–intermediate magnitudes affecting all the region. Conversely, image–based methods are more accurate in the magnitude estimation, with a less biased error pattern: the median curves of errors increase with the distance, both for the magnitude and the position estimation. Moreover, since the depth has been taken into account when computing the distance to the nearest GNSS station, we also find that the underestimation of large magnitudes committed by TS (cf. Figure 4) is affecting very shallow and near events, leading to the conclusion that image–based data representation can bring more exploitable information about the deformation field. Therefore, more low–magnitude events are captured.

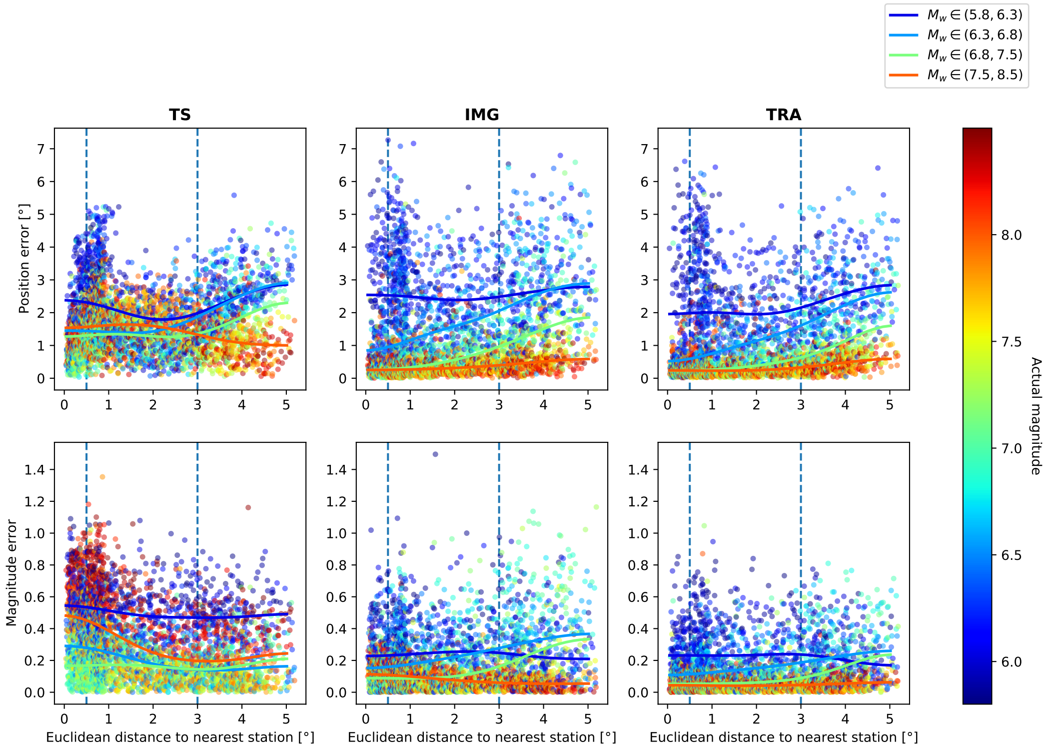

Figure 6. Comparison of errors as a function of the distance to the nearest GNSS station. The deep learning models are shown in columns, while the rows indicate position and magnitude errors, respectively. Each scatter plots depicts errors as a function of the Euclidean distance to the nearest GNSS station, expressed in arc degrees. Each data point, representing the position error and the absolute magnitude error between the test samples and the model predictions, is color coded by the actual magnitude of the event. Solid lines represent the median of subsets of the data points, filtered by magnitude ranges as indicated in the legend in the top right. Vertical dashed lines discriminate among near, intermediate and far field, respectively.

Table 2. Magnitude thresholds of TRA estimated against the synthetic test set.

	depth $\leq$ 30 km	30 km $<$ depth $\leq$ 60 km	60 km $<$depth $\leq$ 100 km
near field	6	6.2	6.5
interm. field	6.8	6.8	7
far field	7.5	7.5	7.8

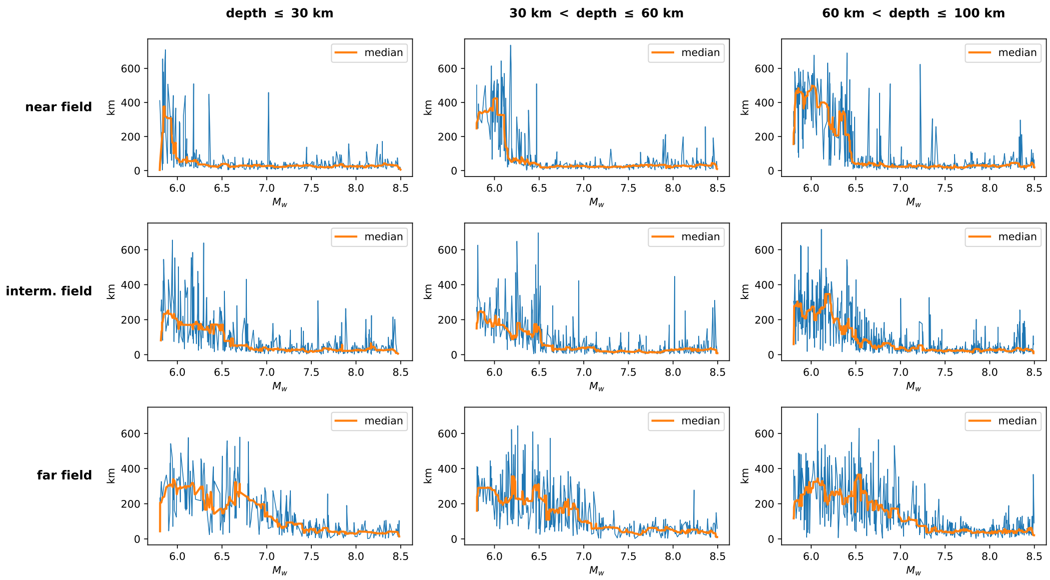

Figure 7. Position error, computed for each test sample, as a function of the magnitude (x axis), the depth range (columns) and the distance range (cf. Figure 6) with respect to the GNSS network (rows) for TRA. The orange solid line represents the result of a median smoothing by employing a kernel size of 15 points.

3.4 Magnitude threshold estimation from TRA localization error

By positioning this deep learning pipeline in an operational framework, is it interesting to ask whether a characterization coming from a learning model is reliable. Indeed, we are interested in the confidence of the model subject to the physical constraints that come from the tectonic context under consideration, notably the magnitude (SNR), the distance and the depth, as seen in the previous sections.

Figure 7 shows the position error for the TRA method, computed for each test sample, as a function of the magnitude, with each subplot corresponding to a different range of hypocenter–station distances and hypocentral depths. The general idea is to get an estimation of the magnitude threshold for different settings, *i.e.*, for different values of depth and distance to the GNSS network. This can serve as a prior probability associated to each new event that we are willing to characterize, such that we can assess, *a priori*, if the deep learning model will be able to characterize it with an acceptable precision. To keep it simple, in this study we will not estimate any probability but assign a hard threshold (*characterizable*, *non–characterizable*).

As discussed in previous sections, as the depth increases, the magnitude detection limit also increases. For events having a depth $d \leq 30$ km, we can set a magnitude thresh-

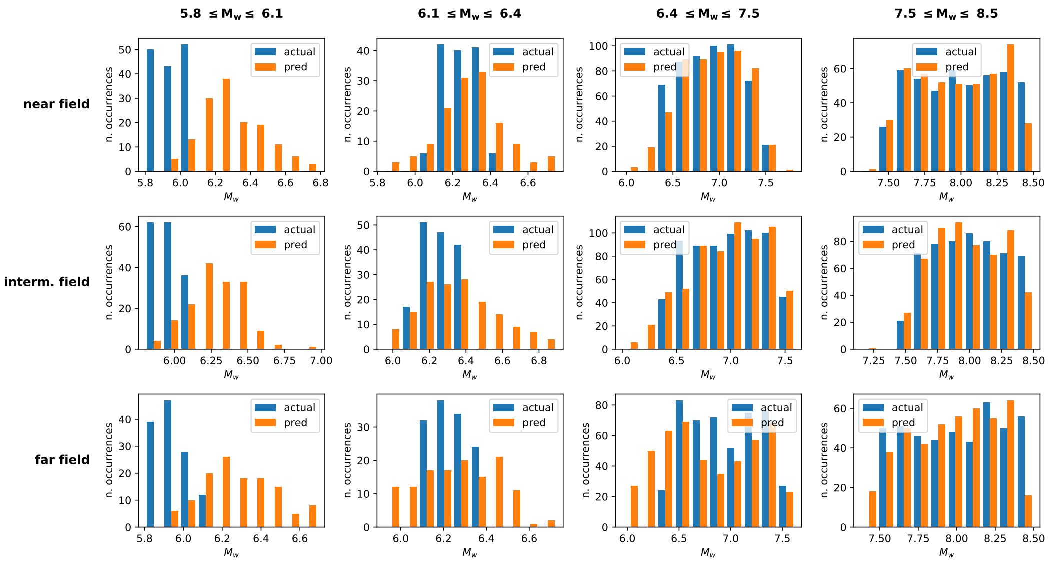

Figure 8. Histograms of the predicted magnitude (orange bars) and ground truth synthetic magnitude (blue bars) as a function of the distance range (cf. Figure 6) with respect to the GNSS network (rows) and for different magnitude ranges (columns), for the TRA model.

old at M_w 6, by selecting a limit where there is an evident discontinuity and where the error is reasonably low with respect to the general trend. As for intermediate and far field, it is harder to find a clear discontinuity, as the interplay between magnitude, distance and depth is generally nonlinear, yet a general tendency can be still observed. The estimated thresholds will be M_w 7 for the intermediate and far fields. We should also consider that, to parity of depth range, the relative distance between the event and the GNSS network strongly affects the probability of correct retrieval, making the magnitude threshold larger and larger. This poses some limitations in the characterization of deep and far offshore events, with only large magnitude earthquakes being characterizable in those conditions. A summary of the chosen magnitude thresholds for TRA can be found in table 2.

3.5 Trade–off between depth and magnitude in the TRA magnitude resolution

Figure 8 shows the histograms of the estimated (using TRA) and the real magnitude as a function of the distance ranges (same ranges than in Figure 6). This plot helps us further understand in which conditions the model predictions are reliable, by studying their statistics with respect to the magnitude posterior distribution. For sufficiently high magnitudes (third and fourth column), the conditional posterior of the predicted and the ground truth magnitude are in good accordance. This may be an indication that the method has indeed learnt how to correctly characterize the high magnitudes based on input image time series (see also Figure S1 in the Supporting Information). However, the predicted magnitude distributions for lower magnitudes are far from the ground truth values and seem to be drawn from a Gaussian distribution. We performed a statistical normality test by following the approach of (R. B. D'Agostino, 1971; R. D'Agostino & Pearson, 1973), confirming that the predicted magnitudes for TRA in low–SNR conditions are following a normal distribution. The same observation is found for IMG, while the TS prediction is more difficult to interpret (cf. figures S2 to S7).

The deep learning models are not provided with any prior on the depth, therefore they cannot resolve the ambiguities coming from the interaction between magnitude, position and depth. Hence, by adding a prior knowledge on the depth, it may be possible to better resolve the magnitude, improving also the localization performance.

For the first two magnitude ranges (cf. first and second columns of Figure 8), the distributions are centered around $M_w = 6.3$. Since their standard deviation is $\sigma = 0.2$, the 99.7% of the realizations will fall within $\pm 3\ \sigma$, *i.e.*, in the range $(5.8, 7)$. In fact, $M_w = 7$ is seemingly a threshold value beyond which the magnitude resolution ability of TRA is exceptionally high (cf. Figure S1). Therefore, the method is resolving intermediate–low magnitude by drawing predictions according to a normal probability distribution centered in the middle of the uncertainty range. Furthermore, some residuals of the Gaussian tail are visible in the third column, likely corresponding to deformation fields associated to high–depth events, which thus have been wrongly characterized as $M_w < 6.4$ events.

4 Application to real GNSS data

4.1 Data processing

The data selection for real events in Japan has been conducted as follows. The F-Net catalog from NIED (cf. https://www.fnet.bosai.go.jp) has been exploited and events ranging from 1998 to 2021 have been selected according to the studied range of characteristics (epicentral position, hypocentral depth, magnitude, see section 2.2.1) for a total of 85 events. Magnitudes have been allowed to exceed the 8.5 limit in order to further test the models on high-magnitude events, even though it's out of the training range. Since GNSS data is daily sampled, if more than one event is recorded in the same day, only the maximum magnitude event is kept. All events in 2011 have been removed except the Tohoku event (11 March 2011). Indeed, the earthquake and subsequent tsunami damaged several GPS stations, and the time series of the remaining ones are dominated by a strong post-seismic relaxation effect making GNSS time series difficult to interpolate and interpret on an automated manner.

Two GNSS data sets have been collected: the data processed in double difference at ISTerre (Institut des Sciences de la Terre) that range from 1998 to 2019 (Marill et al., 2021; gnss products, 2019) and the data processed in PPP at NGL (Nevada Geodetic Laboratory) (Blewitt et al., 2018). that range from 2009 to 2021. We performed outlier detection and removal by processing the data with the *hampel filter* (Pearson et al., 2016) with a window length $n = 3$. Thereafter, we extracted, for each date in the seismic catalog, a window of 100 days, centered onto the coseismic offset (cf. section 2.2). We considered a 100–day stack of time series as valid if at least 60% of the stations are present (*i.e.*, ~ 180) and if at least the 70% of the median number of data points in the 100–day window (*i.e.*, 70) is not undefined (*i.e.*, less than 30% of data gaps). The remaining data gaps are filled as follows. After centering the time window on the coseismic offset date, we compute the linear trend in the first and the second half. Thanks to this procedure, an approximation is provided for the small data gaps and also a first order reconstruction of the coseismic offset when that information may be missing. Finally, the data is detrended, *i.e.*, the linear trend is subtracted for every 100–day stack.

After the previous processing, the ISTerre/DD and the NGL/PPP data sets contain 70 and 52 labelled time series. We used the magnitude thresholds obtained for TRA (cf. table 2) to differentiate the theoretically characterizable events from the rest, as shown in Figure 9, that is if magnitude, depth and position of the events are such that they satisfy those experimentally–derived relationships. We found 8 and 5 characterizable events for ISTerre/DD and NGL/PPP data sets, respectively. The data is further rearranged

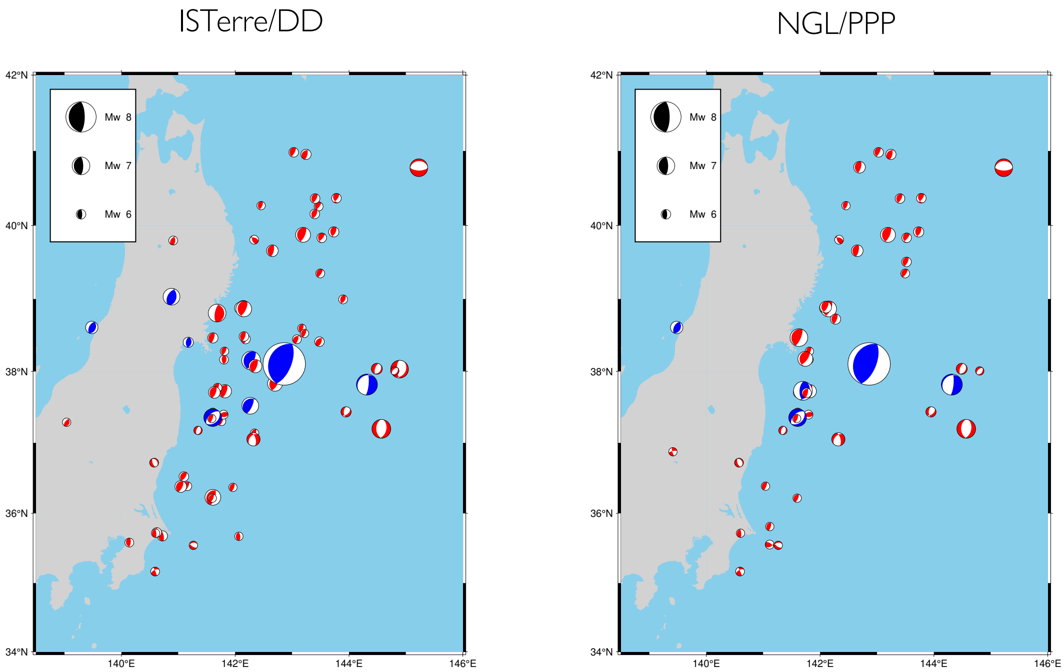

Figure 9. Seismic catalog associated to the ISTerre/DD and NGL/PPP data sets, respectively. ISTerre/DD data set contains 114 events ranging from 1998 to 2019, while NGL/PPP set contains 94 events ranging from 2009 to 2021. Focal mechanisms are depicted for each event and their size is proportional to the magnitude, according to the legend at the top left. Blue–colored focal mechanisms indicate the characterizable events according to table 2.

Table 3. Quantitative results of the tested methods on the characterizable events belonging to the real data sets.

	ISTerre/DD		NGL/PPP	
Model	Position error (km)	Magnitude error (°)	Position error (km)	Magnitude error (°)
TS	1422.53 ± 2634.99	11.57 ± 24.52	1536.11 ± 2553.39	12.53 ± 23.99
IMG	$\mathbf{87.98 \pm 78.49}$	$\mathbf{0.26 \pm 0.20}$	143.52 ± 79.44	0.48 ± 0.44
TRA	140.08 ± 150.79	0.36 ± 0.31	$\mathbf{126.45 \pm 87.61}$	$\mathbf{0.43 \pm 0.21}$

into differential images and image time series and the performance of the three deep learning methods are evaluated.

4.2 Results and discussion

The quantitative results are shown in table 3, while Figure 10 shows the performance of the tested methods on the two real data sets. The displacement fields associated to all the characterizable events in the ISTerre/DD dataset are represented in Figure 11.

The performance of the image–based models is more accurate than the TS model on both data sets, in line with the results obtained on synthetic data (cf. section 3). This is probably linked to the presence of a huge amount of data gaps and missing stations,

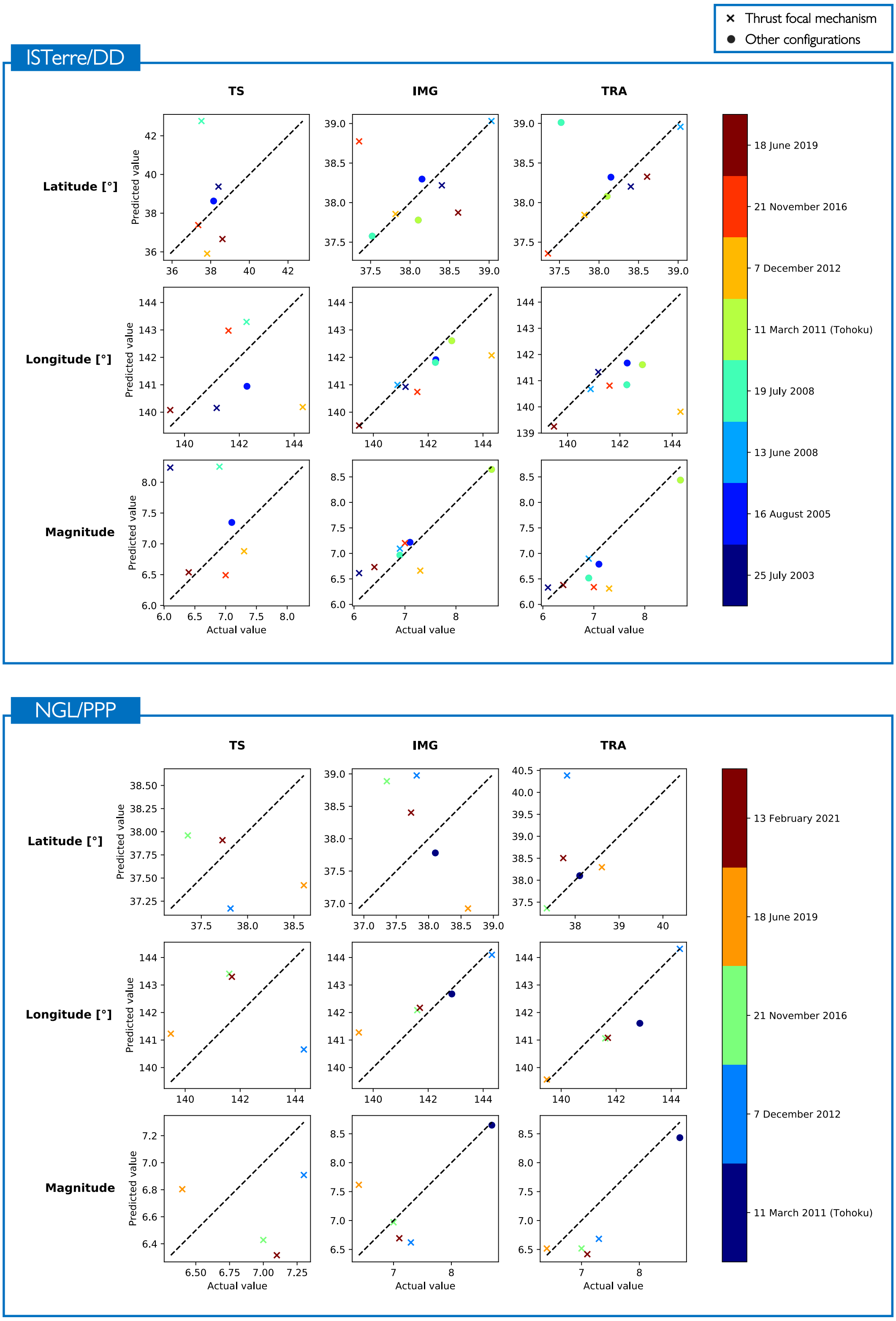

Figure 10. Performance plots on real data from ISTerre/DD and NGL/PPP data sets. Each subplot shows the *real vs predicted* comparison for the estimated parameters (Latitude, longitude and magnitude in each row) for each of the three methods (TS, IMG and TRA in each columns). For each scatter plot, circles represent mean predictions associated to events having thrust focal mechanism, with crosses indicating any other focal mechanism. The solid dashed line shows the line of perfect predictions. The data points are color–coded according to the time of occurrence. The predictions of the TS model associated to the events that occurred the 13 June 2008 and the 11 March 2011 (Tohoku) are not visible as their predictions are located outside of the plot limits (*i.e.*, outliers, cf. section 4.2).

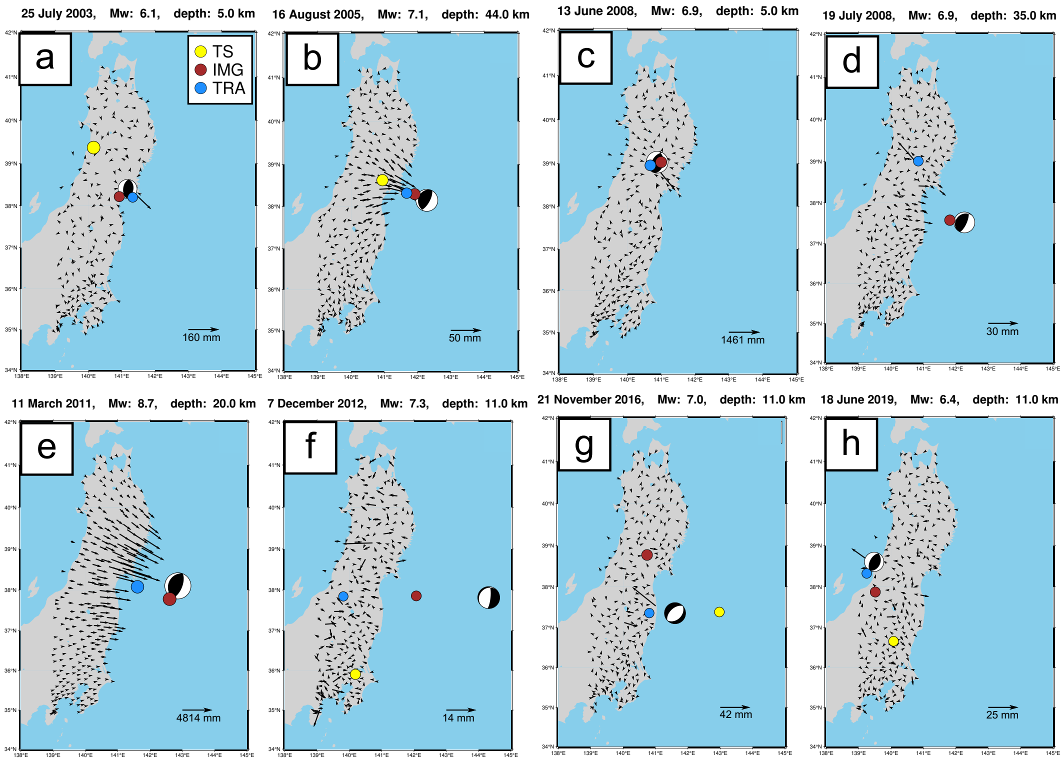

Figure 11. Displacement fields associated to the eight events of the ISTerre/DD data set. The deformation fields have been computed by subtracting the deformation at day $t_c + 1$ and $t_c - 1$. In each subplot, the focal mechanism from the NIED catalog is shown as well as the magnitude and depth (in each title) with the yellow, brown and blue points representing the predictions for TS, IMG and TRA, respectively. The predictions of the TS model are not visible in the plot for the events occurred the 13 June 2008, the 19 July 2008 and the 11 March 2011 (Tohoku), as they have been located outside of the figure bounds (cf. section 4.2).

which worsen the resemblance between synthetic and real data, thus deteriorating the performance of TS. As a result, image–based models can better deal with data gaps thanks to the spatial interpolation. Hence, the amount and continuity of the data plays an essential role on the final prediction accuracy, which is indeed mitigated by the image and image time series representations. It is also worth to notice that all the models have a larger error associated to the NGL/PPP dataset, probably because of the Precise Point Positioning solution, which is slightly noisier with respect to the DD approach. Since the noise in the training samples is obtained from DD solution time series, it is not surprising that the model may have a lower performance better on data obtained from a PPP solution. Therefore, this constitutes a possible limitation of the method, which could be overcome by applying a fine–tuning to improve the results on the PPP data set. For these reasons, we will focus on the ISTerre/DD data set henceforward.

The events in Figure 10 have been marked with a different symbol if their rupture has a thrust focal mechanism ($\phi_S = 200 \pm 40°$, $\delta = 25 \pm 20°$, $\lambda = 90 \pm 45°$). Differentiating thrust and non–thrust events is interesting to assess if the shape of the associated deformation field plays a key role in the characterization performed by image–based models, given that the model was trained on thrust events only. Indeed, the results shown in Figure 10 seem to suggest that the shape of the deformation field (*e.g.*, cf. outliers having strike–slip focal mechanism) is not a relevant feature in the characterization of the location and the magnitude, since the predictions of the image–based models do not seem to depend on the nature of the focal mechanism, which indeed would be a key information when inverting for the focal mechanism itself. Hence, the amplitude of the deformation and the SNR (linked to the intensity of the interpolated image) are likely the most informative characteristics to retrieve the epicenter and the magnitude of the earthquake, especially in the coastal stations, which often register the highest displacement values.

Interestingly, IMG and TRA models seem to be complementary on some events, as shown in figures 11 (d), (g) and (h). The TRA model is unable to separate the source of deformation in the 19 July 2008 event (Figure 11 (d)) because of a persistent outlier in the displacement field, whose influence is better mitigated by the differential approach used for IMG (cf. Figure S9 – S11). On the contrary, TRA can effectively retrieve the 21 November 2016 event (11 (g)), likely thanks to the spatio–temporal approach (cf. Figure S12 – S17), while IMG is not well performing. This seems to suggest that the two different image–based data representations carry some particular characteristics coming from the network geometry and the spatio–temporal variability of the data.

As a further comment, we notice that the outlier displacement value north of the epicenter of the 19 July 2008 event (cf. Figure 11 (d)) is actually an artifact introduced by the linear interpolation performed on the time series in presence of a large data gap (cf. Figure S8). Therefore, either a more efficient method should be set up for the missing data interpolation, or artifacts should be taken into account in the training data base. Accounting for the data gaps is not a trivial task and future developments should focus on this aspect, since, as we saw, the larger the data gaps, the harder is the characterization.

Finally, it is worth to mention the performance of the models on the Tohoku event (11 March 2011, $M_w = 8.7$, according to the NIED solution used in the paper), which is estimated as a $M_w \sim 8.6$ and $M_w \sim 8.5$ event by IMG and TRA, respectively, with the IMG model correctly exceeding the training upper bound (M_w 8.5) on a pattern which has never been presented to the network during the training phase. Yet, it should be noted that this result should be taken carefully as the actual magnitude of the event is 9.1 (Lay, 2018).

5 Conclusions

We studied and developed an end–to–end framework for the seismic source characterization with GNSS data. We constructed three deep learning methods associated with three data representations: time series, differential images and image time series. We train our methods on synthetic data generated to be subduction events compliant with actual events occurring in the Japan subduction zone. We tested the methods both on synthetic and real GNSS data, and we studied the performance and the sensitivity of the three methods, evidencing their strengths and their limits.

Image–based methods outperform time series–based methods, possibly because their associated data representations better exploit the topology of the GNSS network. The wavelength of the deformation is seemingly better constrained with images with respect to time series, the longitudinal extent of the deformation being more difficult to characterize by means of the temporal evolution only. Results on synthetic data clearly evidence a detection threshold associated to GNSS data, which is associated to the SNR, and also dependent on the depth and position of events. This allows us to partition the output space by identifying regions in which the source characterization can be performed with confidence.

Performance on real data sets is consistent with the results obtained on synthetic data and shows accurate and reliable results. Image–based methods outperform the time–series based approach in both the real data sets, with image–time series and the TRA model showing that the spatio–temporal approach proposed is crucial in resolving the location and magnitude of most of the real events. However, the noise characterization needs to be improved, in order to better account for outliers in GNSS time series, data gaps and, possibly, common modes. By improving the simulation of the realistic noise, we can produce more and more real–looking synthetic data, possibly having better results on the characterization and a lower SNR threshold. Nonetheless, the results on real data are promising and could potentially lead to an effective analysis of the slow deformation, which would benefit from the present work as well as from the potential refinements that we have listed before.

Acknowledgments

All rough data used in this manuscript are available through doi:10.17178/GNSS.products.Japan (ISTerre/DD) and http://geodesy.unr.edu (NGL). This work has been supported by ERC CoG 865963 DEEP-trigger. Most of the computations presented in this paper were performed using the GRICAD infrastructure (https://gricad.univ-grenoble-alpes.fr), which is supported by Grenoble research communities. All the computations needed to build images and image time series, as well as map plots, have been performed thanks to the GMT software (and its Python wrapper, pyGMT) (Wessel et al., 2019). Thanks to Juliette Cresseaux for the support and suggestions.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... others (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Aki, K., & Richards, P. G. (2002). *Quantitative seismology.*

Anantrasirichai, N., Biggs, J., Albino, F., & Bull, D. (2019). A deep learning approach to detecting volcano deformation from satellite imagery using synthetic datasets. *Remote Sensing of Environment*, *230*, 111179.

Bergen, K. J., Johnson, P. A., Maarten, V., & Beroza, G. C. (2019). Machine learning for data-driven discovery in solid earth geoscience. *Science*, *363*(6433).

Bevis, M., & Brown, A. (2014). Trajectory models and reference frames for crustal motion geodesy. *Journal of Geodesy*, *88*(3), 283–311.

Blewitt, G., Hammond, W., & Kreemer, C. (2018). Harnessing the GPS Data Explosion for Interdisciplinary Science. *Eos*, *99*. doi: 10.1029/2018EO104623

Bock, Y., & Melgar, D. (2016). Physical applications of gps geodesy: A review. *Reports on Progress in Physics*, *79*(10), 106801.

Bottou, L., Curtis, F. E., & Nocedal, J. (2018). Optimization methods for large-scale machine learning. *Siam Review*, *60*(2), 223–311.

Bürgmann, R. (2018). The geophysics, geology and mechanics of slow fault slip. *Earth and Planetary Science Letters*, *495*, 112–134.

Chollet, F. (2021). *Deep learning with python*. Simon and Schuster.

Chollet, F., et al. (2015). *Keras.* GitHub. Retrieved from `https://github.com/fchollet/keras`

D'Agostino, R., & Pearson, E. S. (1973). Tests for departure from normality. empirical results for the distributions of b_2 and $\sqrt{b}$. *Biometrika*, *60*(3), 613–622.

D'Agostino, R. B. (1971). An omnibus test of normality for moderate and large size samples. *Biometrika*, *58*(2), 341–348.

Dong, D., Fang, P., Bock, Y., Cheng, M., & Miyazaki, S. (2002). Anatomy of apparent seasonal variations from gps-derived site position time series. *Journal of Geophysical Research: Solid Earth*, *107*(B4), ETG–9.

Donoso, F., Moreno, M., Ortega-Culaciati, F., Bedford, J., & Benavente, R. (2021). Automatic detection of slow slip events using the picca: Application to chilean gnss data. *Frontiers in Earth Science*, *9*.

Dragert, H., Wang, K., & James, T. S. (2001). A silent slip event on the deeper cascadia subduction interface. *Science*, *292*(5521), 1525–1528.

Frank, W. B., & Brodsky, E. E. (2019). Daily measurement of slow slip from low-frequency earthquakes is consistent with ordinary earthquake scaling. *Science advances*, *5*(10), eaaw9386.

Fukuda, J. (2018). Variability of the space-time evolution of slow slip events off the boso peninsula, central japan, from 1996 to 2014. *Journal of Geophysical Research: Solid Earth*, *123*(1), 732–760.

Gardonio, B., Campillo, M., Marsan, D., Lecointre, A., Bouchon, M., & Letort, J. (2019). Seismic Activity Preceding the 2011 Mw9.0 Tohoku Earthquake, Japan, Analyzed With Multidimensional Template Matching. *Journal of Geophysical Research: Solid Earth*, *124*, 6815–6831. Retrieved 2022-01-11, from `https://onlinelibrary.wiley.com/doi/abs/10.1029/2018JB016751` doi: 10.1029/2018JB016751

Gardonio, B., Marsan, D., Bouchon, M., Socquet, A., Jara, J., Sun, Q., ... Campillo, M. (2018). Revisiting Slow Slip Events Occurrence in Boso Peninsula, Japan, Combining GPS Data and Repeating Earthquakes Analysis. *Journal of Geophysical Research: Solid Earth*, *123*, 1502–1515. Retrieved 2018-06-13, from `http://doi.wiley.com/10.1002/2017JB014469` doi: 10.1002/2017JB014469

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feed-forward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256).

gnss products. (2019). Gnss position solutions in japan. *"CNRS, OSUG, ISTERRE"*. doi: 10.17178/GNSS.products.Japan

Gomberg, J., Wech, A., Creager, K., Obara, K., & Agnew, D. (2016). Reconsidering earthquake scaling. *Geophysical Research Letters*, *43*(12), 6243–6251.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Gutenberg, B. (1956). The energy of earthquakes. *Quarterly Journal of the Geological Society*, *112*(1-4), 1–14.

Hanks, T. C., & Kanamori, H. (1979). A moment magnitude scale. *Journal of Geophysical Research: Solid Earth*, *84*(B5), 2348–2350.

Haykin, S. S. (2008). *Adaptive filter theory.* Pearson Education India.

He, B., Wei, M., Watts, D. R., & Shen, Y. (2020). Detecting slow slip events from seafloor pressure data using machine learning. *Geophysical Research Letters*, *47*(11), e2020GL087579.

Hirose, H., Matsuzawa, T., Kimura, T., & Kimura, H. (2014). The boso slow slip events in 2007 and 2011 as a driving process for the accompanying earthquake swarm. *Geophysical Research Letters*, *41*(8), 2778–2785.

Hirose, H., & Obara, K. (2005). Repeating short-and long-term slow slip events with deep tremor activity around the bungo channel region, southwest japan. *Earth, planets and space*, *57*(10), 961–972.

Hulbert, C., Rouet-Leduc, B., Johnson, P. A., Ren, C. X., Rivière, J., Bolton, D. C., & Marone, C. (2019). Similarity of fast and slow earthquakes illuminated by machine learning. *Nature Geoscience*, *12*(1), 69–74.

Hulbert, C., Rouet-Leduc, B., Jolivet, R., & Johnson, P. A. (2020). An exponential build-up in seismic energy suggests a months-long nucleation of slow slip in cascadia. *Nature communications*, *11*(1), 1–8.

Ide, S., Beroza, G. C., Shelly, D. R., & Uchide, T. (2007). A scaling law for slow earthquakes. *Nature*, *447*(7140), 76–79.

Ito, Y., Hino, R., Kido, M., Fujimoto, H., Osada, Y., Inazu, D., . . . Ashi, J. (2013). Episodic slow slip events in the Japan subduction zone before the 2011 Tohoku-Oki earthquake. *Tectonophysics*, *600*, 14–26. Retrieved 2019-01-08, from https://linkinghub.elsevier.com/retrieve/pii/S0040195112004994 doi: 10.1016/j.tecto.2012.08.022

Itoh, Y., Aoki, Y., & Fukuda, J. (2022). Imaging evolution of cascadia slow-slip event using high-rate gps. *Scientific reports*, *12*(1), 1–12.

Ji, K. H., & Herring, T. A. (2013). A method for detecting transient signals in gps position time-series: smoothing and principal component analysis. *Geophysical Journal International*, *193*(1), 171–186.

Khoshmanesh, M., Shirzaei, M., & Uchida, N. (2020). Deep slow-slip events promote seismicity in northeastern Japan megathrust. *Earth and Planetary Science Letters*, *540*, 116261. Retrieved 2022-02-04, from https://linkinghub.elsevier.com/retrieve/pii/S0012821X20302041 doi: 10.1016/j.epsl.2020.116261

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kong, Q., Trugman, D. T., Ross, Z. E., Bianco, M. J., Meade, B. J., & Gerstoft, P. (2019). Machine learning in seismology: Turning data into insights. *Seismological Research Letters*, *90*(1), 3–14.

Lay, T. (2018). A review of the rupture characteristics of the 2011 tohoku-oki mw 9.1 earthquake. *Tectonophysics*, *733*, 4–36.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Lowry, A. R., Larson, K. M., Kostoglodov, V., & Bilham, R. (2001). Transient fault slip in guerrero, southern mexico. *Geophysical Research Letters*, *28*(19), 3753–3756.

Mao, A., Harrison, C. G., & Dixon, T. H. (1999). Noise in gps coordinate time series. *Journal of Geophysical Research: Solid Earth*, *104*(B2), 2797–2816.

Marill, L., Marsan, D., Socquet, A., Radiguet, M., Cotte, N., & Rousset, B. (2021). Fourteen-year acceleration along the japan trench. *Journal of Geophysical Research: Solid Earth*, *126*(11), e2020JB021226.

Marsan, D., Reverso, T., Helmstetter, A., & Enescu, B. (2013). Slow slip and aseismic deformation episodes associated with the subducting Pacific plate offshore Japan, revealed by changes in seismicity. *Journal of Geophysical Research: Solid Earth*, *118*, 4900–4909. Retrieved 2022-01-11, from http://doi.wiley.com/10.1002/jgrb.50323 doi: 10.1002/jgrb.50323

Michel, S., Gualandi, A., & Avouac, J.-P. (2019). Interseismic coupling and slow slip events on the cascadia megathrust. *Pure and Applied Geophysics*, *176*(9), 3867–3891.

Moon, T. K., & Stirling, W. C. (2000). *Mathematical methods and algorithms for signal processing* (No. 621.39: 51 MON).

Mousavi, S. M., & Beroza, G. C. (2020). A machine-learning approach for earthquake magnitude estimation. *Geophysical Research Letters*, *47*(1), e2019GL085976.

Mousavi, S. M., Ellsworth, W. L., Zhu, W., Chuang, L. Y., & Beroza, G. C. (2020). Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking. *Nature communications*, *11*(1), 1–12.

Münchmeyer, J., Bindi, D., Leser, U., & Tilmann, F. (2021). Earthquake magnitude and location estimation from real time seismic waveforms with a transformer network. *Geophysical Journal International*, *226*(2), 1086–1104.

Münchmeyer, J., Bindi, D., Sippl, C., Leser, U., & Tilmann, F. (2020). Low uncertainty multifeature magnitude estimation with 3-d corrections and boosting tree regression: application to north chile. *Geophysical Journal International*, *220*(1), 142–159.

Nishikawa, T., Matsuzawa, T., Ohta, K., Uchida, N., Nishimura, T., & Ide, S. (2019). The slow earthquake spectrum in the Japan Trench illuminated by the S-net seafloor observatories. *Science*, *365*(6455), 808–813. Retrieved 2019-10-08, from http://www.sciencemag.org/lookup/doi/10.1126/science.aax5618 doi: 10.1126/science.aax5618

Nishimura, T. (2014). Short-term slow slip events along the Ryukyu Trench, southwestern Japan, observed by continuous GNSS. *Progress in Earth and Planetary Science*, *1*, 22. Retrieved 2019-10-08, from http://progearthplanetsci.springeropen.com/articles/10.1186/s40645-014-0022-5 doi: 10.1186/s40645-014-0022-5

Nishimura, T. (2021). Slow Slip Events in the Kanto and Tokai Regions of Central Japan Detected Using Global Navigation Satellite System Data During 1994–2020. *Geochemistry, Geophysics, Geosystems*, *22*, e2020GC009329. Retrieved 2021-08-20, from https://onlinelibrary.wiley.com/doi/10.1029/2020GC009329 doi: 10.1029/2020GC009329

Okada, Y. (1985). Surface deformation due to shear and tensile faults in a halfspace. *Bulletin of the seismological society of America*, *75*(4), 1135–1154.

Okada, Y., Nishimura, T., Tabei, T., Matsushima, T., & Hirose, H. (2022). Development of a detection method for short-term slow slip events using gnss data and its application to the nankai subduction zone. *Earth, Planets and Space*, *74*(1), 1–18.

Ozawa, S., Murakami, M., Kaidzu, M., Tada, T., Sagiya, T., Hatanaka, Y., ... Nishimura, T. (2002). Detection and monitoring of ongoing aseismic slip in the tokai region, central japan. *Science*, *298*(5595), 1009–1012.

Ozawa, S., Murakami, M., & Tada, T. (2001). Time-dependent inversion study of the slow thrust event in the nankai trough subduction zone, southwestern japan. *Journal of Geophysical Research: Solid Earth*, *106*(B1), 787–802.

Ozawa, S., Yarai, H., Imakiire, T., & Tobita, M. (2013). Spatial and temporal evolution of the long-term slow slip in the bungo channel, japan. *Earth, Planets and Space*, *65*(2), 67–73.

Pearson, R. K., Neuvo, Y., Astola, J., & Gabbouj, M. (2016). Generalized hampel filters. *EURASIP Journal on Advances in Signal Processing*, *2016*(1), 1–18.

Prichard, D., & Theiler, J. (1994). Generating surrogate data for time series with several simultaneously measured variables. *Physical review letters*, *73*(7), 951.

Radiguet, M., Cotton, F., Vergnolle, M., Campillo, M., Valette, B., Kostoglodov, V., & Cotte, N. (2011). Spatial and temporal evolution of a long term slow slip event: the 2006 guerrero slow slip event. *Geophysical Journal International*, *184* (2), 816–828.

Radiguet, M., Perfettini, H., Cotte, N., Gualandi, A., Valette, B., Kostoglodov, V., … Campillo, M. (2016). Triggering of the 2014 mw7. 3 papanoa earthquake by a slow slip event in guerrero, mexico. *Nature Geoscience*, *9* (11), 829–833.

Rogers, G., & Dragert, H. (2003). Episodic tremor and slip on the cascadia subduction zone: The chatter of silent slip. *Science*, *300* (5627), 1942–1943.

Ross, Z. E., Yue, Y., Meier, M.-A., Hauksson, E., & Heaton, T. H. (2019). Phaselink: A deep learning approach to seismic phase association. *Journal of Geophysical Research: Solid Earth*, *124* (1), 856–869.

Rouet-Leduc, B., Hulbert, C., & Johnson, P. A. (2019). Continuous chatter of the cascadia subduction zone revealed by machine learning. *Nature Geoscience*, *12* (1), 75–79.

Rouet-Leduc, B., Hulbert, C., McBrearty, I. W., & Johnson, P. A. (2020). Probing slow earthquakes with deep learning. *Geophysical research letters*, *47* (4), e2019GL085870.

Rouet-Leduc, B., Jolivet, R., Dalaison, M., Johnson, P. A., & Hulbert, C. (2021). Autonomous extraction of millimeter-scale deformation in insar time series using deep learning. *Nature communications*, *12* (1), 1–11.

Rousset, B., Campillo, M., Lasserre, C., Frank, W. B., Cotte, N., Walpersdorf, A., … Kostoglodov, V. (2017). A geodetic matched filter search for slow slip with application to the mexico subduction zone. *Journal of Geophysical Research: Solid Earth*, *122* (12), 10–498.

Saad, O. M., Hafez, A. G., & Soliman, M. S. (2020). Deep learning approach for earthquake parameters classification in earthquake early warning system. *IEEE Geoscience and Remote Sensing Letters*, *18* (7), 1293–1297.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4510–4520).

Saxe, A. M., McClelland, J. L., & Ganguli, S. (2013). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*.

Schreiber, T., & Schmitz, A. (1996). Improved surrogate data for nonlinearity tests. *Physical review letters*, *77* (4), 635.

Seydoux, L., Balestriero, R., Poli, P., Hoop, M. d., Campillo, M., & Baraniuk, R. (2020). Clustering earthquake signals and background noises in continuous seismic data with unsupervised deep learning. *Nature communications*, *11* (1), 1–12.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Smith, W., & Wessel, P. (1990). Gridding with continuous curvature splines in tension. *Geophysics*, *55* (3), 293–305.

Socquet, A., Valdes, J. P., Jara, J., Cotton, F., Walpersdorf, A., Cotte, N., … Norabuena, E. (2017). An 8 month slow slip event triggers progressive nucleation of the 2014 chile megathrust. *Geophysical Research Letters*, *44* (9), 4046–4053.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15* (1), 1929–1958.

Takagi, R., Uchida, N., & Obara, K. (2019). Along-strike variation and migration of long-term slow slip events in the western nankai subduction zone, japan. *Journal of Geophysical Research: Solid Earth*, *124* (4), 3853–3880.

Uchida, N., Asano, Y., & Hasegawa, A. (2016). Acceleration of regional plate subduction beneath Kanto, Japan, after the 2011 Tohoku-oki earthquake. *Geophysical Research Letters*, *43*, 9002–9008. Retrieved 2019-10-09, from `http://doi.wiley.com/10.1002/2016GL070298` doi: 10.1002/2016GL070298 van den Ende, M. P., & Ampuero, J.-P. (2020). Automated seismic source characterization using deep graph neural networks. *Geophysical Research Letters*, *47*(17), e2020GL088690.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Wallace, L. M. (2020). Slow slip events in new zealand. *Annual Review of Earth and Planetary Sciences*, *48*, 175–203.

Wallace, L. M., & Beavan, J. (2010). Diverse slow slip behavior at the hikurangi subduction margin, new zealand. *Journal of Geophysical Research: Solid Earth*, *115*(B12).

Wallace, L. M., Webb, S. C., Ito, Y., Mochizuki, K., Hino, R., Henrys, S., ... Sheehan, A. F. (2016). Slow slip near the trench at the hikurangi subduction zone, new zealand. *Science*, *352*(6286), 701–704.

Wang, Q., Guo, Y., Yu, L., & Li, P. (2017). Earthquake prediction based on spatio-temporal data mining: an lstm network approach. *IEEE Transactions on Emerging Topics in Computing*, *8*(1), 148–158.

Wessel, P., Luis, J., Uieda, L., Scharroo, R., Wobbe, F., Smith, W., & Tian, D. (2019). The generic mapping tools version 6. *Geochemistry, Geophysics, Geosystems*, *20*(11), 5556–5564.

Williams, S. D., Bock, Y., Fang, P., Jamason, P., Nikolaidis, R. M., Prawirodirdjo, L., ... Johnson, D. J. (2004). Error analysis of continuous gps position time series. *Journal of Geophysical Research: Solid Earth*, *109*(B3).

Zhang, J., Bock, Y., Johnson, H., Fang, P., Williams, S., Genrich, J., ... Behr, J. (1997). Southern california permanent gps geodetic array: Error analysis of daily position estimates and site velocities. *Journal of geophysical research: solid earth*, *102*(B8), 18035–18055.

Zhang, X., Zhang, M., & Tian, X. (2021). Real-time earthquake early warning with deep learning: Application to the 2016 m 6.0 central apennines, italy earthquake. *Geophysical Research Letters*, *48*(5), 2020GL089394.

Zhu, W., & Beroza, G. C. (2019). Phasenet: a deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, *216*(1), 261–273.

Zhu, W., Mousavi, S. M., & Beroza, G. C. (2019). Seismic signal denoising and decomposition using deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(11), 9476–9488.