Array-based convolutional neural networks for automatic earthquake detection and 4D localization

Heather Shen[1]* **and Yang Shen**[1]

[1]Graduate School of Oceanography, University of Rhode Island, Narragansett, RI, United States

Key Points:

- Convolutional neural network models based on seismic arrays automatically and accurately detect and localize earthquakes
- Application to continuous seismic data in Hawai'i detects and localizes 6 times as many earthquakes as the published catalog.
- This is the first deep learning model that can automatically provide an earthquake catalog from the continuous data of a seismic network.

*As a volunteer research assistant

Corresponding author: Yang Shen, yshen@uri.edu

Abstract

The growing amounts of seismic data necessitates efficient and effective methods to monitor earthquakes. Current methods are computationally expensive, ineffective under noisy environments, or labor-intensive. We leverage advances in machine learning to propose an improved solution – a convolutional neural network that uses array data to seamlessly detect and localize events. When testing this methodology with events at Hawai'i, we achieve 99.4% accuracy and predict hypocenter locations within a few kilometers of the U.S. Geological Survey catalog. We demonstrate that training with relocated earthquakes reduces localization errors significantly. We outline several ways to improve the model, including enhanced data augmentation and use of relocated offshore earthquakes recorded by ocean bottom seismometers. Application to continuous records shows that our algorithm detects 6 times as many earthquakes as the published catalog. Due to the enhanced detection sensitivity, localization granularity, and minimal computation costs, our solution is valuable, particularly for real-time earthquake monitoring.

Plain Language Summary

Earthquake catalogs provide baseline information about the movement of the crust and related geological hazards. Yet, catalogs are usually incomplete and fail to log smaller earthquakes undetected by seismic networks. Here, we present a new deep learning model that is computationally efficient and can seamlessly detect and locate earthquakes from continuous seismic records. When the new model is applied to Hawai'i, it yields 6 times as many earthquakes as the published catalog, promising a more complete catalog that will help improve understanding of seismic and volcanic processes.

1 Introduction

Recent advances in instrumentation have provided an exponential increase in seismic data. Yet, detecting and localizing earthquakes at scale remains expensive and inefficient. Traditional earthquake detection methods used by many seismic network operators (e.g. Allen, 1982; Withers et al., 1998) do not perform well for small earthquakes in noisy environments. In addition, network operations often involve human review of earthquake arrivals and time picks as well as iterative tuning of hypocenter estimates. To improve detection, methods based on waveform similarity (matched filter or template matching) have been developed and widely applied (e.g. Caffagni et al., 2016; Gibbons & Ringdal, 2006). Such efforts have led to a great increase in the detection of small earth-

quakes, yielding rich details that enable the next generation of analyses of earthuquakes and faults (e.g. Ross et al., 2019). These methods are, however, computationally expensive and limited; detection only works for earthquakes that share similar waveforms and thus likely have the same source regions and mechanisms of the template events.

In the past few years, convolutional neural networks (CNNs) have been adapted for earthquake detection and location. One common feature shared by CNN approaches is that once the model is trained, it is far more computationally efficient than the waveform-similarity-based approach (Gibbons & Ringdal, 2006; Yoon et al., 2015) when it is applied to new data, an advantage important for seismic network operations, particularly during periods of intense seismic activities. Perol et al. (2018) introduced a CNN model for earthquake detection and localization based on waveforms at individual stations. The localization was limited to a few subregions. Lomax et al. (2019) and Mousavi and Beroza (2020) developed CNN models for rapid earthquake characterization using single-station waveforms. Dokht et al. (2019) extended the CNN earthquake detection in the time-frequency domain. Other studies (e.g. Ross et al., 2018; W. Zhu & Beroza, 2018; L. Zhu et al., 2019; J. Wang et al., 2019) focused on seismic phase detection and picking of arrival times, which were then used in traditional travel time-based localization. Kriegerowski et al. (2019) and Zhang et al. (2020) showed it is possible to use CNNs to locate earthquakes without the intermediate step of phase picking; however, the former depended on manually chosen arrival times at a reference station and the latter assumed that seismic events had already been detected. Taking a different approach, Van den Ende and Ampuero (2020) used Graph Neural Networks with multi-station waveforms to locate earthquakes and estimate magnitude. Though they too applied it only to existing catalog events. Here we present a framework based on recent advances in deep learning for seamless, automatic detection and 4D localization of earthquakes without the intermediate steps of phase detection and picking, phase association, travel time calculation, and inversion. Our approach builds upon previous work by using a network of seismic stations to first identify if an earthquake has occurred, and if so, estimate the latitude, longitude, depth, and origin time of the event.

Specifically, we propose a two-stage seismic-array-based, convolutional neural network (ArrayConvNet) model where 1) earthquake detection becomes a supervised classification problem and 2) earthquake localization becomes a supervised regression problem. We train and test on data from 55 seismic stations on the Island of Hawai'i – our solution not only detects earthquakes in the United States Geological Survey (USGS) catalog, but also uncovers 6 times more earthquakes missing from the catalog. Once an earthquake is detected, our model can locate an earthquake's hypocenter to within 3-

4 km of the catalog. To the best of our knowledge, this is the first deep learning model that can automatically provide an earthquake catalog from the continuous data of a seismic network. Finally, we outline several steps that can be taken to greatly reduce the model localization errors, making it a viable solution to improve the efficiency and accuracy of seismic monitoring at much lower computational and human costs.

2 Data

The Hawai'i Island, USA is one of the most seismically and volcanically active regions in the world, a fact that was heightened by the 2018 eruption of Kīlauea Volcano (Neal et al., 2019). The Hawaiian Volcano Observatory (HVO), USGS operates a permanent seismic network (HV) on the island, providing the earthquake information and waveform data needed for this study. We use 55 seismic stations on Hawai'i Island (Figure S1). Among them, 33 have three-component (north, east, and vertical) seismometers while the rest have single, vertical-component seismometers.

Both the earthquake waveforms and noise segments are downloaded from the Incorporated Research Institutions for Seismology (IRIS) Data Management Center (DMC). For each earthquake, we select a 50-second window so that the event origin time is randomly between 1 - 10 seconds from the trace start time. The time difference between the trace start time and the event origin time, along with the catalog hypocentral location (latitude, longitude and depth), is used to train the localization part of the model (see Method). The noise segments are chosen between the USGS reported earthquakes and are 10 to 50 seconds before the origin time of an earthquake. We visually inspect the noise windows to minimize the possible presence of unreported earthquakes in the noise segments. Because the Hawaiian seismic networks have a variety of sensors, we remove instrument response from the traces and transfer them to velocity seismograms to minimize the effects of different instrument sensitivities to ground motion. The earthquake waveforms and noise are filtered between 3 to 20 Hz and downsampled to an uniform sampling rate of 50 samples per second on all channels. The frequency range is chosen for optimum earthquake signal-to-noise ratios based on visual inspection of earthquake waveforms over a wide range of frequencies as well as previous studies of the Hawaiian earthquake characteristics (e.g. Matoza et al., 2014). All traces are normalized individually before they are used as the inputs for the CNN model. For stations with missing records or that do not have three channels, we zero-fill the missing channels.

Our ArrayConvNet model has two stages: one for event detection and one for event localization (see Method). Each stage is trained on distinct training and test data sets.

2.1 Detection

We use 1843 analyst-reviewed earthquakes with magnitude (ml or md) ranging from 0.1 to 5.28 in the 2017 USGS catalog and 1905 noise segments. The number of earthquakes is comparable to that in Perol et al. (2018).

We explored several ways of arranging the input trace data for the detection model and chose the following approach based on the robustness of the results when the model is applied on unseen, continuous data: for both earthquakes and noise segments, we sort the 55 station traces in order of the time of the vertical component's largest amplitude and take the absolute of the traces so all values are between 0 and 1. Therefore, for an earthquake, we see a clear propagation of earthquake arrivals through the seismic network in a visually easily recognizable pattern (Figure S2). The general pattern is consistent from earthquake to earthquake, regardless of the earthquake location and magnitude (and thus signal-to-noise ratio), as the wave always propagates from the lower left to the upper right in the maximum-amplitude-sorted waveform images (Figure S2). For each sorted station, the cross-station features – the information learned by the convolutional kernel, which often has a small size – are local in the time-and-trace-number space in a well-defined trend. In contrast, the unsorted waveforms, arranged alphabetically by station names, do not have an easy-to-follow pattern from event to event. Depending on the source-receiver geometry, a station that has an early earthquake arrival for one event may have a late arrival for the next event. The cross-station features are highly variable and may span the entire time-and-trace-number space, thus requiring a deep and large network to capture. While the sorted and unsorted waveforms do not show substantial differences in terms of model precision, recall, and the receiver operating characteristic (ROC) curve (Figure S3), they yield significantly different numbers of detections when the resulting model is applied to continuous data, indicating differences in the robustness of detection of small earthquakes in noisy data; the unsorted waveforms result in higher number of false detection from visual inspection of the corresponding seismic traces. Intuitively, the sorted waveforms have simpler, lower-order features, which require less complicated neural networks and thus less training data to achieve robust models.

Each input event is labeled with a "0" or an "1" to indicate whether it is a noise or an earthquake event, respectively. This now transforms detection into a well-understood classification problem.

2.2 Localization

For the localization part of the model, we use the earthquake locations and origin times from the same 1843 earthquakes in the 2017 USGS catalog. We augment these original waveforms by performing 7 cuts of 50-s-long waveforms for the same earthquake, with each cut starting at a random time between 1-10 s before the event origin time. The seven cuts of each earthquake have the same hypocenter (latitude, longitude, depth) but different offsets between the trace start time and the origin time. The total number of earthquakes used to train and test the localization part is thus 12,901. Such data augmentation is commonly used in deep learning (J. Wang & Perez, 2017) and, in our case, helps to train the model to better localize the event origin time (see more in Discussion), which is crucial when the model is applied to continuous data.

Given the different units and scales for the hypocenter and origin time, we normalize the latitude, longitude, depth, and time values so that they are all comparable in magnitude (within -1 to 1). For the hypocenter, we subtract a reference location (latitude 19.5° , longitude -155.5° , depth 0 km) from the catalog location and then divide the depth by 50; for the time difference between the trace start time and the origin time, we divide by 10. Therefore, we avoid the situation where one variable (e.g., depth) dominates the loss function.

Unlike the inputs for the detection part of the model, the input traces for localization are arranged alphabetically by station names. This is necessary as localization requires that the station geometry remains a constant. We now may treat localization as a supervised regression problem.

3 Method

3.1 Network Architecture

Traces for each event are arranged as a three-dimensional (3D) tensor $Z(c, s, t)$. The depths of Z for $c \in \{1, 2, 3\}$ correspond to three channels of seismic records, the rows for $s \in \{1, ..., 55\}$ represent various stations, and $t \in \{1, \ldots, 2500\}$ represents the time index of trace values. Inputs are then processed in a feed-forward stack of three convolutional layers, followed by two fully connected layers that in the detection model, output class scores and in the localization model, output latitude, longitude, depth, and time offset between the trace start time and earthquake origin time (Figure 1).

After each convolutional layer, we use a rectified linear unit (ReLU) layer to apply an element-wise activation function and then a max pooling layer to perform a down-

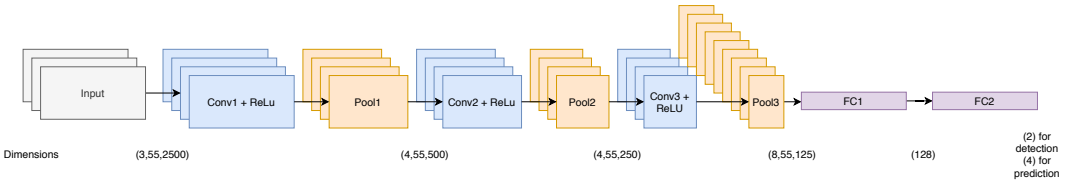

Figure 1. Architecture of array-based CNN model. Conv and ReLu stand for the convolutional layer and rectified linear unit layer, while Pool represents max pooling. FC1 and FC2 are the two fully connected layers. Numbers within the parenthesis represent the dimensions of the input or output data at the various stages.

sampling operation and decrease the number of parameters. Convolutions are also zero-padded to maintain input shape.

The kernel of the first convolutional layer has a dimension of width of 9 and height of 1. The kernels in the second and third layers have the same dimension of width of 3 and height of 5. The motivation behind the 1D filter in the first layer is to isolate learning of temporal features among the three input channels of each station, as in Kriegerowski et al. (2019), while the 2D filters in later layers are designed to extract cross-station information. Pooling after the first convolutional layer has a size of (1,5) with a stride of (1,5), while pooling after the second and third convolutional layers has a size of (1,2) with a stride of (1,2). Thus pooling in our model is designed primarily to downsample in the time dimension.

We note that our number of convolutional layers (3) and the number of channels in each layer (4, 4, and 8) are substantially smaller than in previous studies (e.g., 8 convolutional layers with 32 channels each in Perol et al., 2018). To determine the optimal network architecture, we explored a range of the number of convolutional layers (2-5), number of channels (2-32), and number of features/neurons of the first fully connected layer (64-1024). Our guiding principle in selecting the optimum models is to find the smallest network that yields better or comparable results in detection precision. Fewer than 3 convolutional layers and smaller than 4 channels per convolutional layer yield lower precision, as the model may be too simple to capture the full complexity of the data. Greater than or equal to 4 convolutional layers, larger than 8 channels, and larger than 128 neurons in the first fully connected layer yield detection precision comparable to that of our preferred network, with the training loss far below (in most cases more than an order of magnitude smaller than) the test loss, which suggests overfitting.

3.2 Training the Network

The two parts of the CNN model can be trained separately and then connected for examining continuous data. For detection, we optimize the network parameters by minimizing a cross-entropy loss function. This measures the average discrepancy between our predicted distribution and the true class probability distribution in the training set and is proven for standard classification problems (e.g. Perol et al., 2018). For localization, we optimize the parameters by minimizing a mean-squared error loss between our predicted and given location and time values.

Given our training data set, we are able to minimize our loss functions using a batch approach. We use a typical 75-25% split for the training and test data sets, respectively. At each training step, we feed a batch of 32 inputs to the network, evaluate the expected loss on the batch, and update the network parameters accordingly using backpropagation. We cycle through all training data in batches as an epoch, and after each epoch, we calculate the loss for both the training and test data sets. This is repeated until the loss stops decreasing significantly (80 epochs for both detection and localization, Figure 2).

For optimization, we used the AdamW algorithm (Loshchilov & Hutter, 2017), which builds on the well-known Adam algorithm (Kingma & Ba, 2014) but separates the weight decay from the learning rate. The result of this distinction is that the weight decay and learning rate can be optimized separately, and has been proven to substantially improve generalization performance. For detection, we use the default learning rate, 2e−5; for localization, given the increase in training data due to augmentation, we use a larger learning rate of 5e−5.

3.3 Computational Implementation

We implemented our ArrayConvNet model in Pytorch (Paszke et al., 2019) and performed all model training, testing, and application to continuous data on an iMac with a 3.8 GHz 8-core Intel Core i7 CPU and 128 GB memory. Model training and testing in 80 epochs took about 1.3 and 5.7 hours for the detection and localization parts of the model, respectively. Application of the model to 31-day continuous seismic data took about 5.5 hours.

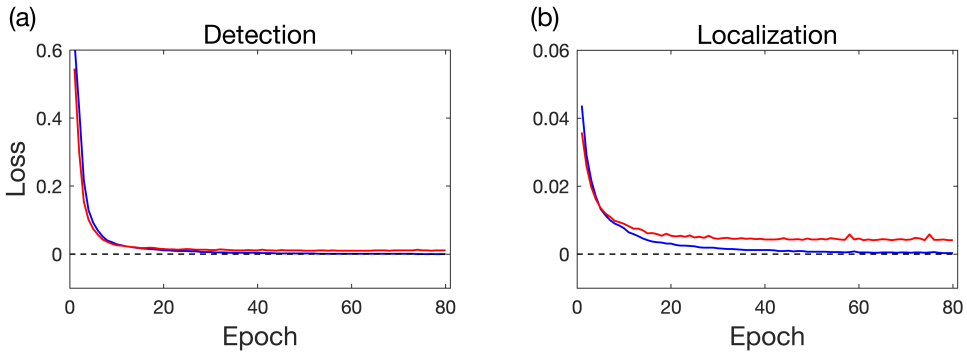

Figure 2. Training (blue line) and test (red line) losses as the function of epochs for the detection and localization parts of the model. It takes less than 20 epochs for both detection and localization for the loss to decrease rapidly and for the test loss to approach a small and relatively stable value. The dashed line marks the zero loss.

4 Results

4.1 Detection

Within 20 epochs, both the training and test losses decrease rapidly and the test loss remains small and relatively stable as the number of epochs increases (Figure 2). For comparison, Perol et al. (2018) used 32,000 epochs to train their model. In Dokht et al. (2019), it took over 10,000 epochs for the earthquake detection learning to approach an asymptotic and stable flat line. We attribute the rapid learning of our model to, at least partially, the relative simplicity of the network architecture.

Our detection accuracy on the test data, defined as the percentage of events that are correctly classified as an earthquake or noise, is 99.4% at 0.5 classification (probability) threshold. Between 0.5 and 0.7 classification threshold for earthquakes, the precision is 99.6% while recall is 99.2-99.0% (Figure S3). Above 0.7 classification threshold, the precision is 100% while recall is 99-98%, suggesting that above this detection threshold ArrayConvNet does not label any noise as earthquakes, at least in the test data, and rarely mis-classifies earthquakes as noise. For comparison, the precision and recall reported by Perol et al. (2018) are 94.8% and 100%, respectively, and those by Dokht et al. (2019) are 99.6% and 99.9%, respectively.

4.2 Localization

Similar to the detection part of the model, the training and test losses of the localization part of the model decrease rapidly within 20 epochs. While the training loss continues to decrease towards zero with increasing epochs, the test loss remains flatlined (Figure 2), suggesting that the network has enough neurons or complexity to fit the training data nearly completely, but uncertainty or random noise in the data keeps the test loss at a certain level; more epochs or a larger network likely would not improve the fit of the test data.

Overall, our model is able to predict the location of an earthquake in the test data within -0.08 $\pm$ 4.5 km in the north-south direction, 0.07 $\pm$ 4.1 km in the east-west direction and -0.02 $\pm$ 3.5 km in depth (Figure 3 and Figure S4). The values following the $\pm$ sign (and hereinafter) represent one standard deviation. The difference between the predicted and catalog origin times is -0.06 $\pm$ 0.81 s.

Some of the location and origin time differences may be attributed to errors in the USGS catalog. Synthetic tests by Zhang et al. (2020) show that adding a location error to the catalog location results in their CNN model prediction error of a similar size. Lin et al. (2014) relocated earthquakes with magnitude greater than 1.0 between 1992

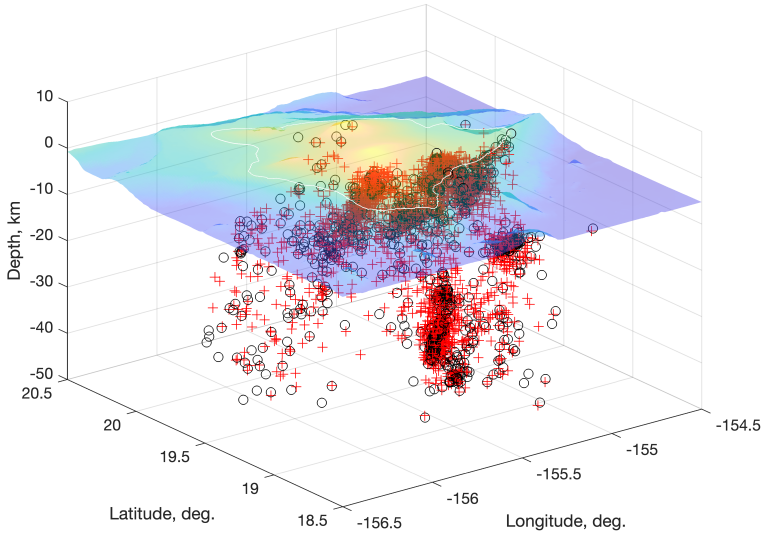

Figure 3. Hypocenter locations of earthquakes from the USGS catalog (circles) and model predictions (red crosses) in a three-dimensional view looking from the southwest direction. Clusters of earthquakes in the catalog and model predictions are clearly visible. The topography and bathymetry of the Island are shown as a semi-transparent surface.

and 2009, using their 3D velocity model and source-specific station term corrections. Their resulting catalog thus represents a subset of the HVO events with the best location quality (Lin et al., 2014). Comparing the earthquake locations in Lin et al. (2014) with the USGS catalog locations, we find a lateral location offset of 1.1 ± 1.8 km and a depth offset of 1.0 ± 2.1 km. So a significant portion of the hypocenter location differences between our model predictions and the USGS catalog may stem from errors in the data used to train the model (see more in Discussion).

5 Application to Continuous Hawaiian Seismic Data

Earthquake catalogs usually represent a subset of earthquakes that occurred, with detection and localization limited by signal-to-noise ratios in seismic records, number of detected stations, and other factors. The USGS catalog for Hawai‘i is no exception. So while our ArrayConvNet performs well for the test data set (Figure S3), further tests on continuous data, combined with expert reviews of the results, are required to evaluate its true efficacy.

For seismic network operators generating earthquake catalogs, one may wish to minimize false detection by using a higher confidence threshold (Ross et al., 2018). Here we follow this approach, using a probability threshold of 0.95 (95% confidence) in the following discussion unless otherwise stated. Based on the precision and recall characteristics (Figure S3), the model should rarely mis-classify earthquakes as noise, and almost never identify noise as an earthquake at this confidence level.

We input continuous seismic data from the same 55 seismic stations in Hawai‘i, which are unseen in the development of our CNN model and preprocessed in the same way as the data used to train the models. The model runs through the data in 50-s-long moving windows at 3-s increments. When the detection stage of the model finds that the probability of an earthquake is above a specified confidence threshold, we determine the exact 50-s window by choosing the one that has the highest detection probability. We then feed the window to the localization stage and calculate the event location. To be consistent with the localization training data, where traces start 1-10 s before the origin time, the declared event must also have a predicted origin time within 1-10 s after the start of the traces (Figure S5).

Using a continuous data stream from January 2018, our model detects and locates 1603 earthquakes, which is approximately 6.1 times the number reported in the USGS catalog. Figure S6 presents the number of earthquakes reported by USGS and our model detection, showing a weak correlation between the daily event numbers. Comparing the

USGS catalog and our model outputs on selected days, we find that most but not all of the catalog events are recovered by our model, consistent with the precision and recall characteristics at the 0.95 threshold (Figure S3). The missing ones are low magnitudes (<0.7) and have low numbers of reporting stations (<10) in the catalog. The events detected and localized by our model have a similar epicentral distribution as those of the USGS catalog events for January 2018 (Figure S7). As a measure of the sensitivity to the detection probability threshold, the model detects and locates 1915 and 1542 earthquakes with the probability thresholds of 0.68 and 0.997, respectively, or 7.3 and 5.9 times the number in the published USGS catalog.

6 Discussions

As with any supervised machine learning, the more accurate and greater the training data, the better the resulting model. In our case, the training data can be improved in several ways:

The first is to include the USGS catalog earthquakes from the many years of monitoring by HVO. A greater number of earthquakes plus a correspondingly large number of noise (visually inspected or automatically screened to minimize the presence of earthquakes in the noise segments) should further improve the accuracy and robustness of the model.

The second is to use relocated earthquakes with more accurate locations (e.g. Got & Okubo, 2003; Wolfe et al., 2004; Matoza et al., 2013; Lin et al., 2014; Shelly & Thelen, 2019). Lin and Okubo (2020) relocated over 48,000 earthquakes between July 2015 and August 2018. With the caveat that all the relocated events in Lin and Okubo (2020) are onshore, we found that using 1806 earthquakes in the year 2017 relocated by Lin and Okubo (2020) to train localization in the same way as we discussed above for the USGS catalog reduces the location difference between the model prediction and the catalog by 25-45% (from $\pm$ 4.5 to $\pm$ 2.4 km in the north-south direction, from $\pm$ 4.1 to $\pm$ 2.5 km in the east-west direction, and from $\pm$ 3.5 to $\pm$ 2.6 km in depth) and the origin time difference by 13% (from $\pm$ 0.8 to $\pm$ 0.7 s), demonstrating the effects of relocated catalogs with lower location errors. For offshore earthquakes, those located with additional ocean-bottom seismometer records (Anchieta et al., 2011; Merz et al., 2019) may see large improvements as the catalog based on the onshore HVO networks may contain higher errors. Relocation of earthquakes recorded by the ocean bottom seismic array deployed shortly after the 2018 Kīlauea eruption is currently underway (Wei et al., 2019). The results, together with relocated earthquakes onshore (Shelly & Thelen, 2019; Lin & Okubo, 2020), will be used to update our ArrayConvNet model.

The third is to use enhanced data augmentation. Due to limited computing resources, we have not explored the asymptotic limit of the number of cuts per earthquake in improving localization. Our tests show that using 7 cuts of the same earthquake with random offsets between the trace start time and event origin time improves the hypocenter depth from the case with no data augmentation by more than a factor of 2 (from $\pm$ 7.8 km to $\pm$ 3.5 km), and from the case with 3 cuts per earthquake ($\pm$ 4.7 km) by 26%. This form of data augmentation is clearly effective in improving localization of (origin) time and reducing its tradeoff with the location and the event depth in particular. Another computationally more expensive form of data augmentation is to generate realistic synthetic earthquake waveforms that may account for topography, 3D velocity heterogeneities, and attenuation (e.g. N. Wang et al., 2018). Such synthetic waveforms are Earth-model dependent, but have the advantage that the sources can be placed anywhere, filling the gaps of the catalog earthquake distribution.

Our model focuses on typical catalog earthquakes with short-period and high-frequency energy. However, there are volcanic and magmatic activities that generate long-period (LP) and very-long-period (VLP) seismicity with frequencies below the frequencies used in this study (e.g. Battaglia et al., 2003; Dawson & Chouet, 2014; Matoza et al., 2014; Wech et al., 2020). Because the frequencies of LP and VLP events overlap with microseism, broadening the frequency range to the LP and VLP frequencies may cause an overall decrease of trace signal-to-noise ratios. We suggest that LP and VLP events should be processed differently and modeled separately from the typical catalog earthquakes.

Beyond these improvements to the model, we suggest that this approach can be generalized for other areas. Although the limitation of this methodology is the size of the training set and number of stations, transfer learning may be applicable in this context. Starting from an existing, well-performing model, it is common to only retrain the last layers of the model and apply it to a different application. Utilizing transfer learning decreases the requirement of having thousands to millions of labeled earthquake events to orders of magnitude less, making our suggested methodology much more accessible. On the other hand, as we demonstrated with the Hawaiian data, training of ArrayConvNet with more or less stations and events requires only moderate computational resources that are accessible to nearly everyone. Thus, ArrayConvNet may be useful in other seismically active locations, where earthquake catalogs already exist. The unique potential values of ArrayConvNet are 1) its computational efficiency, which facilitates real-time seismic monitoring; 2) its sensitivity and robustness in detecting and localizing small earthquakes under noisy conditions, which may enable next generation of analyses of earthquakes and faults (Ross et al., 2019); and 3) its independence from template earthquakes

(as opposed to waveform similarity based methods), which allows it to uncover events with source locations and mechanisms that have not been cataloged before.

Acknowledgments

We thank Quoqing Lin for providing the relocated earthquakes used to assess the effects of catalog location errors. Special thanks to the staff at HVO for collecting the seismic data and providing the earthquake catalog used in this study. Y. Shen is supported by the U.S. NSF Grant No. 1949620. There is no financial conflicts of interests of any kind with respect to the results of this paper for any author.

The USGS earthquake catalog is obtained from (`https://earthquake.usgs.gov/earthquakes/search/`, last accessed March 23, 2020). The waveform data is available from the IRIS DMC (`https://ds.iris.edu/ds/nodes/dmc/`, last accessed March 2020).

Python codes and the trained models are available at `https://github.com/seismolab/ArrayConvNet`

References

Allen, R. (1982). Automatic phase pickers: Their present use and future prospects. *Bull. Seismol. Soc. Am*, *72*(6B), 225–242.

Anchieta, M., Wolfe, C., Pavlis, G., Vernon, F., Eakins, J., Solomon, S., ... Collins, J. (2011). Seismicity around the hawaiian islands recorded by the plume seismometer networks: Insight into faulting near Maui, Molokai, and Oahu. *Bull. Seismol. Soc. Am*, *101*(4), 1742–1758. doi: 10.1785/0120100271.

Battaglia, J., Got, J.-L., & Okubo, P. (2003). Location of long-period events below Kīlauea Volcano using seismic amplitudes and accurate relative relocation. *J. Geophys. Res*, *108*(B12), 2553. doi: 10.1029/2003JB002517.

Caffagni, E., Eaton, D., Jones, J., & Baan, M. (2016). Detection and analysis of microseismic events using a matched filtering algorithm (mfa). *Geophys. J. Int*, *206*(1), 644–658. doi: 10.1093/gji/ggw168

Dawson, P., & Chouet, B. (2014). Characterization of very-long-period seismicity accompanying summit activity at Kīlauea Volcano. *J. Volcanology and Geothermal Research*, *278-279*, 59–85. Retrieved from `http://dx.doi.org/10.1016/j.jvolgeores.2014.04.010`. doi: 10.1016/j.jvolgeores.2014.04.010.

Dokht, R., Kao, H., Visser, R., & Smith, B. (2019). Seismic event and phase detection using time-frequency representation and convolutional neural networks. *Seismol. Res. Lett*, *90*(2A), 481–490. doi: 10.1785/0220180308

Gibbons, S., & Ringdal, F. (2006). The detection of low magnitude seismic events using array-based waveform correlation. *Geophys. J. Int*, *165*(1), 149–166. doi: 10.1111/j.1365-246X.2006.02865.x

Got, J.-L., & Okubo, P. (2003). New insight into Kīlauea's volcano dynamics brought by large-scale relative relocation of microearthquakes. *J. Geophys. Res*, *108*(B7), 2337. doi: 10.1029/2002JB002060.

Kingma, D., & Ba, J. (2014). *Adam: A method for stochastic optimization.* Retrieved from `https://arxiv.org/abs/1412.6980.`

Kriegerowski, M., Petersen, G., Vasyura-Bathke, H., & Ohrnberger, M. (2019). A deep convolutional neural network for localization of clustered earthquake based on multistation full waveforms. *Seismol. Res. Lett.* doi: 10.1785/0220180320

Lin, G., & Okubo, P. (2020). Seismic evidence for a shallow detachment beneath Kīlauea's south flank during the 2018 activity. *Geophys. Res. Lett.* doi: 10 .1029/2020GL088003

Lin, G., Shearer, P., Matoza, R., Okubo, P., & Amelung, F. (2014). Three-dimensional seismic velocity structure of Mauna Loa and Kīlauea volcanoes in Hawai'i from local seismic tomography, tomography. *J. Geophys. Res. Solid Earth*, *119*, 4377–4392. doi: 10.1002/2013JB010820.

Lomax, A., Michelini, A., & Jozinović, D. (2019). An investigation of rapid earthquake characterization using single-station waveforms and a convolutional neural network. *Seismol. Res. Lett.*, *90*(2A), 517-529. doi: 10.1785/0220180311

Loshchilov, I., & Hutter, F. (2017). *Decoupled weight decay regularization.*

Matoza, R., Shearer, P., Lin, G., Wolfe, C., & Okubo, P. (2013). Systematic relocation of seismicity on Hawai'i Island from 1992 to 2009 using waveform cross correlation and cluster analysis. *J. Geophys. Res. Solid Earth*, *118*, 2275–2288. doi: 10.1002/jgrb.50189.

Matoza, R., Shearer, P., & Okubo, P. (2014). High-precision relocation of long-period events beneath the summit region of Kīlauea Volcano, Hawai'i, from 1986 to 2009. *Geophys. Res. Lett*, *41*, 3413–3421. doi: 10.1002/2014GL059819.

Merz, D., Caplan-Auerbach, J., & Thurber, C. (2019). Seismicity and velocity structure of Loihi submarine volcano and southeastern Hawai'i. *J. Geophys. Res. Solid Earth*, *124*(2). doi: 10.1029/2019JB018168.

Mousavi, S., & Beroza, G. (2020). Bayesian-deep-learning estimation of earthquake location from single-station observations. *IEEE Transactions on Geoscience and Remote Sensing*, 1-14. doi: 10.1109/TGRS.2020.2988770.

Neal, C. A., Brantley, S. R., Antolik, L., Babb, J. L., Burgess, M., Calles, K., ... Damby, D. (2019). The 2018 rift eruption and summit collapse of Kīlauea Volcano. *Science*, *363*(6425), 367–374. doi: 10.1126/science.aav7046

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alch, E. e Buc, & R. Garnett (Eds.), *Advances in neural information processing systems 32* (p. 8024–8035). Curran Associates, Inc. Retrieved from `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf` (Retrieved from)

Perol, T., Gharbi, M., & Denolle, M. (2018). Convolutional neural network for earthquake detection and location. *Science Advances*, *4*(2), 1700578. doi: 10.1126/sciadv.1700578

Ross, Z., Meier, M.-A., Hauksson, E., & Heaton, T. (2018). Generalized seismic phase detection with deep learning. *Bull. Seismol. Soc. Am*, *108*(5A), 2894–2901. doi: 10.1785/0120180080.

Ross, Z., Trugman, D., Hauksson, E., & Shearer, P. (2019). Searching for hidden earthquakes in southern California. *Science*, *364*, 767-771. doi: 10.1126/science.aaw6888

Shelly, D., & Thelen, W. (2019). Anatomy of a caldera collapse: Kīlauea 2018 summit seismicity sequence in high resolution. *Geophys. Res. Lett*, *46*, 14395–14,403. doi: 10.1029/2019GL085636.

Van den Ende, M., & Ampuero, J.-P. (2020). Automated seismic source characterization using deep graphic neural networks. *Geophys. Res. Lett*, *47*(17), 2020088690. Retrieved from `https://doi.org/10.1029/2020GL088690` doi: 10.1029/2020GL088690

Wang, J., & Perez, L. (2017). *The effectiveness of data augmentation in image classification using deep learning.* (arXiv:1712.04621v1 [cs.CV] 13 Dec 2017)

Wang, J., Xiao, Z., Liu, C., Zhao, D., & Yao, Z. (2019). Deep learning for picking seismic arrival times. *J. Geophys. Res. Solid Earth*, *124*(7), 6612–6624. doi: 10.1029/2019JB017536.

Wang, N., Li, J., Borisov, D., Gharti, H., Shen, Y., Zhang, W., & Savage, B. (2018). Modeling three-dimensional wave propagation in anelastic models with surface topography by the optimal strong stability preserving Runge-Kutta method. *J. Geophys. Res. Solid Earth*, *124*. Retrieved from `https://doi.org/10.1029/2018JB016175.` doi: 10.1029/2018JB016175.

Wech, A., Thelen, W., & Thomas, A. (2020). Deep long-period earthquakes gen-

erated by second boiling beneath Mauna Kea volcano. *Science*, *368*(6492), 775–779. doi: 10.1126/science.aba4798.

Wei, X., Shen, Y., Caplan-Auerbach, J., & Morgan, J. (2019, December). Seismicity of the Kīlauea Submarine South Flank Following the 2018 Eruption and Mw 6.9 Earthquake. In *Agu fall meeting abstracts* (Vol. 2019, p. V41A-02). Retrieved from `https://ui.adsabs.harvard.edu/abs/2019AGUFM.V41A..02W`

Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., & SusanMoore, J. T. (1998). A comparison of select trigger algorithms for automated global seismic phase and event detection. *Bull. Seismol. Soc. Am*, *88*, 95–106.

Wolfe, C., Okubo, P., Ekstrom, G., Nettles, M., & Shearer, P. (2004). Characteristics of deep (≥13 km) hawaiian earthquakes and hawaiian earthquakes west of 155.55w. *Geochem. Geophys. Geosyst*, *5*, 04006. doi: 10.1029/2003GC000618.

Yoon, C., O'Reilly, O., Bergen, K., & Beroza, G. (2015). Earthquake detection through computationally efficient similarity search. *Sci. Adv*, *1*, 1501057. doi: 10.1126/sciadv.1501057

Zhang, X., Zhang, J., Yuan, C., Liu, S., Chen, Z., & Li, W. (2020). Locating induced earthquakes with a network of seismic stations in Oklahoma via a deep learning method. *Sci. Reports*, *10*(1941). Retrieved from `https://doi.org/10.1038/s41598-020-58908-5.` doi: 10.1038/s41598-020-58908-5.

Zhu, L., Peng, Z., McClellan, J., Li, C., Yao, D., Li, Z., & Fang, L. (2019). Deep learning for seismic phase detection and picking in the aftershock zone of 2008 mw7.9 Wenchuan earthquake. *Physics Earth Planet. Int*, *293*, 106261. doi: 10.1016/j.pepi.2019.05.004.

Zhu, W., & Beroza, G. (2018). PhaseNet: a deep-neural network-based seismic arrival-time picking method. *Geophys. J. Int*, *216*(1), 261–273. doi: 10.1093/gji/ggy423

Figure1_model-architecture.pdf.

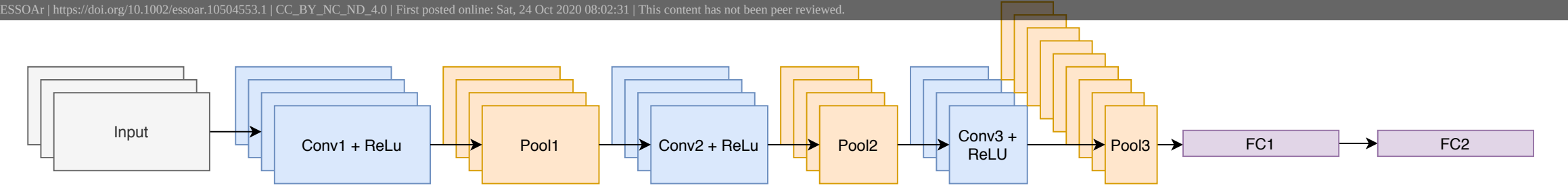

Figure2_loss_vs_epoch.png.

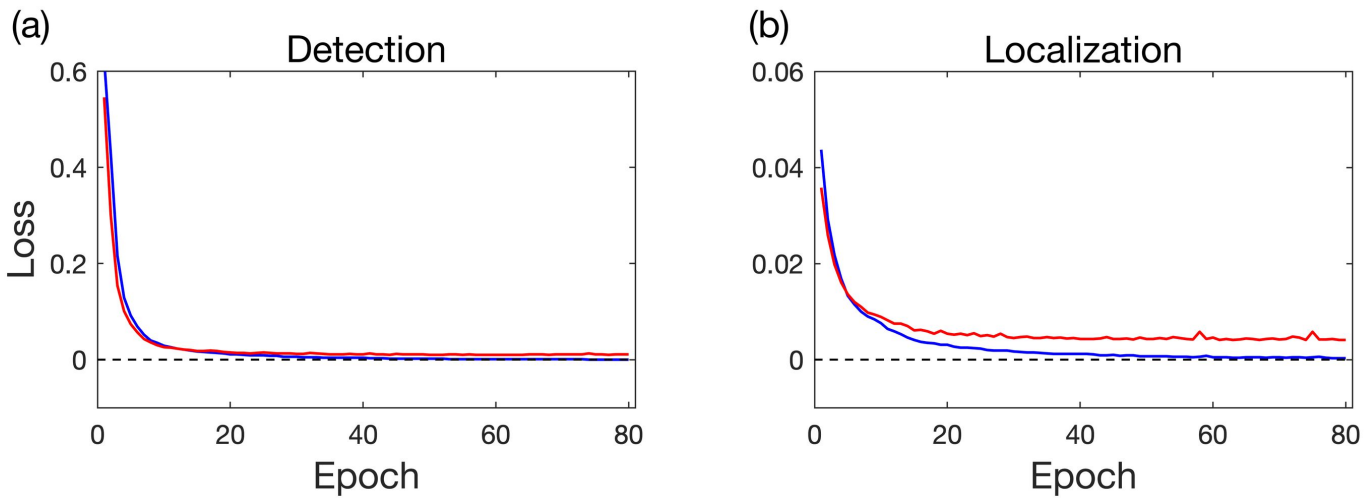

Figure3_hvo_vs_pre_3dview1SW_r600.png.