Multifeature Fusion-Based Earthquake Event Classification Using Transfer Learning

Gwantae Kim, Bonhwa Ku, and Hanseok Ko, *Senior Member, IEEE*

***Abstract*— This letter proposes a multifeature fusion model using deep convolution neural networks and transfer learning approach for earthquake event classification. There are several feature representations for seismic analysis, such as the time domain, the frequency domain, and the time–frequency domain. To successfully classify various earthquake events, we propose a novel model that combines these features hierarchically. In addition, we apply a transfer learning to mitigate overfitting problem of deep learning model while achieving high classification performance. To evaluate our approach, we conduct experiments with the Korean peninsula earthquake database from 2016 to 2018 and a large earthquake database on the Circum-Pacific belt in 2019. The experimental results show that the proposed method outperforms over the compared state-of-the-art methods.**

***Index Terms*— Convolution neural network (CNN), deep learning, earthquake event classification, multifeature fusion, transfer learning.**

I. Introduction

EARTHQUAKE events classification is an essential task to analyze the presence of earthquake events, thereby preventing possible earthquake disasters by an early warning. In most previous approaches, only moderate and large earthquakes were considered while microearthquakes with low magnitude were often labeled as plain noise [1]. Although macroearthquake induces greater structural damages and human casualties, microearthquake also causes hazards in some regions where effective countermeasures are not well prepared. In addition, macroearthquakes tend to occur after a series of microearthquakes taking place. Thus, it is important to detect and classify not only macroseismic events but also microseismic events. Moreover, since earthquakes exhibit different seismic activity trends depending on the magnitude, distinguishing the type of earthquakes (e.g., macroearthquake, microearthquake, and noise) accurately becomes important to performing earthquake analysis.

Traditional approaches of seismic event detection and classification have relied on analyzing the phase arrival time, which is based on detecting the onset of a characteristic function. Short-time average over long-time average (STA/LTA) [2], which is a commonly used earthquake event detection method, used amplitude and time derivative of a single seismic trace to generate the characteristic function. While STA/LTA is fast and effective for large earthquake detection, it is sensitive to making errors in low signal-to-noise ratio (SNR) environments and requires to set the trigger thresholds with heuristic design. Other methods employed high-order statistics, such as kurtosis or skewness [3], [4]. These methods have the potential of producing phase arrival times in low SNR environments but fail to capture the relative amplitudes of seismic events at different frequency bands [5]. The autocorrelation [6] and cross-correlation methods [7] are based on waveform similarity between predetected templates and upcoming waveform. Although these methods are effective in finding commonly occurring earthquakes, it is hard to detect rare and unusual earthquakes. Recently, the fingerprint and similarity thresholding (FAST) [8] method, which uses similarity search through locality-sensitive hashing, has shown extracting generalized features to improve the efficiency of the similarity search.

In recent years, the quantity of seismic data has been increased immensely by the seismological research groups. In order to process and analyze these massive amounts of data, machine learning approaches are rapidly evolving in the earthquake research fields. Shallow artificial neural network (ANN)-based methods [9] have marked as the beginning of neural network approaches for seismic signal detection. The ANN model, which typically consists of fully connected (FC) layers, requires careful model settings to reduce the number of parameters and prevent overfitting.

With the evolution of advanced neural network architecture, there have been several attempts to solve seismic signal processing problems using advanced model structures, such as convolution neural networks (CNNs) [1], [10]–[13], recurrent neural networks (RNNs) [14], and generative adversarial networks (GANs) [15]. In [1], a 1-D CNN model was proposed using a three-channel raw seismic waveform to solve earthquake detection problem. Although the model is designed for detecting not only large earthquakes but also low-magnitude earthquakes, the difference between macroearthquakes and microearthquakes was not considered. In [13], an adaptive high-pass filter model using CNN was proposed to reduce the seismic noise. It determines the cutoff frequency for independent waveform, but the information in the low-frequency band is lost because of applying a high-pass filter. The supervised model using spectral image [12] has been shown that seismic events have significant features in the time–frequency (TF) domain, but the frequency band of the running spectra is restricted to less than 10 Hz.

In this letter, we investigate the potential utility of 2-D representations for seismic event classification and propose an effective CNN structure for feature extraction and aggregation. First, we analyze and compare various 2-D representations of the seismic waveform and find effective features for CNN-based earthquake event classification. Second, we propose a CNN model for feature extraction from the

Manuscript received November 26, 2019; revised March 9, 2020; accepted May 4, 2020. This work was supported by Meteorological/Earthquake See-At Technology Development Research under Grant KMI2018-09610. *(Corresponding author: Hanseok Ko.)*

The authors are with the School of Electrical Engineering, Korea University, Seoul 02841, South Korea (e-mail: gtkim@ispl.korea.ac.kr; bhku@ispl.korea.ac.kr; hsko@korea.ac.kr).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2020.2993302

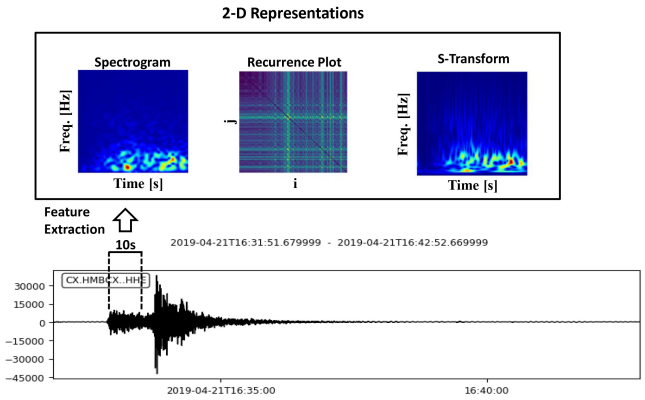

Fig. 1. Depiction of the 2-D feature extraction.

2-D representation. The proposed model focuses on the usage of mid-level features and high-level features of CNN using global average pooling (GAP) and skip-connection to the last layer. Finally, we design an aggregation model that applies transfer learning. The envisioned transfer learning-based method has the advantages in which it trains feature extractor separately for each type of representation and avoids overfitting when training a huge model.

To evaluate the performance, we use the earthquakes and seismic noise signals observed on the Korean peninsula from 2016 to 2018 and large earthquakes in 2019 that occurred on or near the Circum-Pacific belt.

II. 2-D Representations

We use three kinds of 2-D representation: spectrogram, recurrence plot (RP), and Stockwell transform, as shown in Fig. 1. The seismic waveform is truncated to 10 s, and the cropped signal is transformed into the 2-D feature space. The left representation is the spectrogram, which is a result of the short-time Fourier transform (STFT), and the right is the Stockwell transform [16]. Both are TF-domain features that include frequency information over time. The middle representation is RP [17] that represents the correlation between two vectors cropped at different times.

A. Spectrogram

Spectrogram is one of the most frequently used 2-D representations for the time-series data analysis. It contains of the magnitude and phase information of local sections of the time-series signal as it changes over time. The STFT of the time-series signal $\boldsymbol{x}(t)$ is

$$\mathrm{STFT}(\tau, f) = \int_{-\infty}^{\infty} x(t) g(\tau - t) e^{-j2\pi f t} dt \tag{1}$$

where τ and f denote the time and frequency axis of spectrogram, and $g(*)$ is a window function. Since the TF-domain features can improve seismic event classification by better characterizing seismic signals [14], [18], we selected spectrogram as 2-D representation.

B. RP

RP [17] is a graphical feature for measuring time constancy of a dynamic system. Assuming that a time-series signal is $\boldsymbol{x}$ and choosing its d-dimensional vector representation as $\boldsymbol{x}_i = [x(i), x(i+1), \ldots, x(i+d)]$ and $\boldsymbol{x}_j$, then the elements of RP are

$$\boldsymbol{RP}_{ij} = H\left(\epsilon - ||\boldsymbol{x}_i - \boldsymbol{x}_j||\right) \tag{2}$$

where $H(*)$ is heaviside step function and ϵ is a threshold. In (2), we use only the distance term $||\boldsymbol{x}_i - \boldsymbol{x}_j||$ for minimizing heuristic design and preserving information. Thus, RP becomes

$$\boldsymbol{RP}_{ij} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||. \tag{3}$$

We adopt RP approach as 2-D representation with a premise that it can capture the characterizing features over time, which are different and distinguishable from noise.

C. Stockwell Transform

The Stockwell transform, also known as "S transform," is a special case of STFT [19] that uses a Gaussian window function. The S transform is defined by the following equation:

$$S(\tau, f) = \int_{-\infty}^{\infty} x(t) \frac{|f|}{\sqrt{2\pi}} e^{\frac{(\tau - t)^2 f^2}{2}} e^{-j2\pi f t} dt. \tag{4}$$

The S transform affects the frequency resolution that depends on the Gaussian window width. As the variance of the Gaussian window is the inverse of f, it can provide better frequency resolution than a spectrogram near the frequency band of f. Therefore, we employ the S transform to improve the frequency resolution of a spectrogram.

III. Proposed Model

The proposed CNN structure for earthquake event classification is presented in Fig. 2. The model is trained in two stages. In the first stage, the classification models for each feature are trained separately. After the models are converged, the outputs of the CNN layers are used to inputs of the second stage. In the second stage, the CNN-based aggregation model is trained by combining features from the first stage. To avoid the overfitting on the training data set [20], the parameters of the feature extraction model, which is trained in the first stage, are not updated by the loss in the second stage. The output of the structure is the estimated label of the classification target. We use a rectified linear unit (ReLU) activation function for the convolution layer and FC layer. Every FC layer takes a dropout [21] with 0.3 probability.

A. Preprocessing

Since each seismic waveform has different bias over the other, we perform a centering operation as in (5) to the raw signal that has the effect of geometrically moving the center of data to the origin

$$\boldsymbol{y}_j = \boldsymbol{x}_j - \frac{1}{L} \sum_{i=1}^{L} x_i \tag{5}$$

where $\boldsymbol{x}_j$ is the jth element of raw waveform $\boldsymbol{x}$, $\boldsymbol{y}_j$ is the jth element of centered waveform $\boldsymbol{y}$, and L is the dimension of $\boldsymbol{x}$. In addition, we normalize the 2-D representations in the range [0–1] to prevent the local minimum problem.

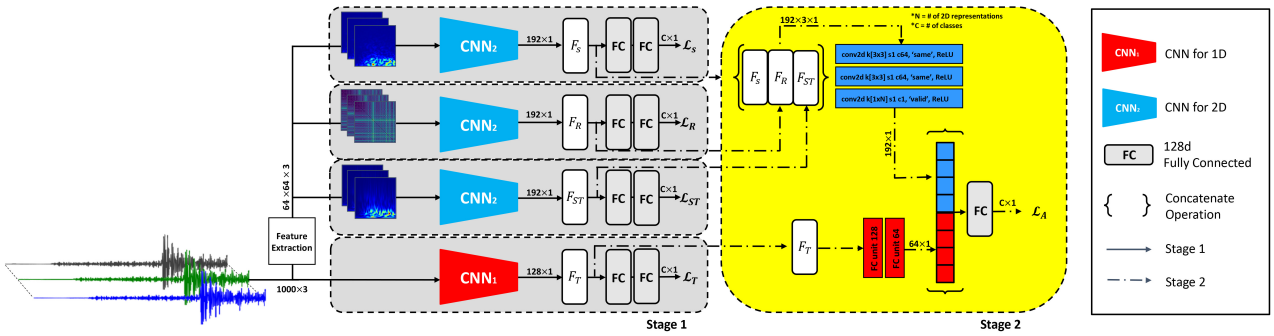

Fig. 2. Proposed training model architecture with pretraining and aggregation.

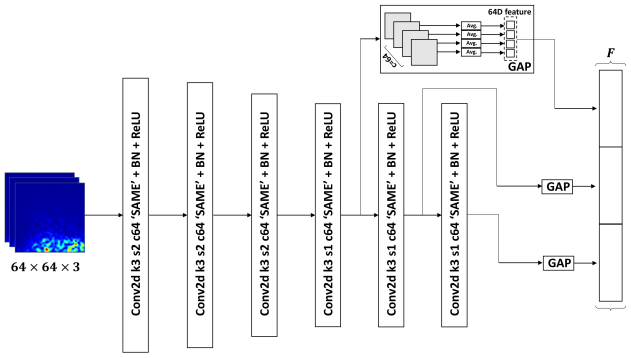

Fig. 3. Proposed CNN structure for 2-D representation in stage 1.

B. CNN Structure for the Single Feature

We essentially train two models: raw waveform and 2-D representation. The CNN structure for raw waveform uses the model proposed by Perol *et al.* [1]. The input is 1000×3 seismic signal, and the dimension of the output vector F_T is 128-D, which is a flattened feature of the last CNN layer. The proposed model, which is shown in Fig. 3, is used for training $64 \times 64 \times 3$.2-D representations. The first three convolution layers have 3×3 kernel, 64 channel, and stride 2, and the next three layers have the same kernel and channel size, but the stride is 1. Every convolution layer contains batch normalization [22] layer. The projections through each CNN layer have a different signification. Low-level features are extracted in shallow layers, and high-level features are extracted in deep layers. To focus on not only high-level features but also mid-level features, the outputs from the batch normalization layer of the last three layers are summarized by GAP [23] and concatenated. The classifier for single-feature consists of two FC layers with 128 units.

C. CNN Structure for Aggregation

We propose an aggregation method using deep neural networks (DNNs). We employ a transfer learning approach to train the fusion model because it is difficult to train the whole model at once. In the second stage, the features F_S, F_R, F_{ST}, and F_T, which are extracted from the pretrained CNN in the first stage, are used as inputs of the envisioned aggregation model. F_S, F_R, and F_{ST}, which are 192×1 vector, are concatenated along the second axis. This merging trick is designed to improve generalization performance. The concatenated matrix becomes the input of the CNN that consists of two convolution layers with 3×3 kernels, 64 channel, and stride 1 and one convolution layer with 3×3 kernels, 1 channel, stride 1, and no zero padding. F_T is 128×1 vector that is passed through two subsequent FC layers with 128 and 64 units, respectively. The outputs of each layer are then concatenated, which then passes through an FC layer with 128 units and softmax function, whereby an estimate of the class label is made.

TABLE I
NUMBER OF DATA

Events	LE	ME	N
train	7168	39900	60000
test	2004	6216	30000

D. Training

In the first stage, each feature extraction network is trained to minimize each of the cross-entropy loss functions L_S, L_R, L_{ST}, and L_T, respectively. In the second stage, the parameters in the aggregation network are updated to minimize the cross-entropy loss L_A. The cross-entropy loss is defined as

$$\text{Loss} = -\sum_{c=1}^{C} y_c \log(\hat{y}_c) \tag{6}$$

where C is the number of classes, y_c is the true label, and $\hat{y}_c$ is the estimated outputs. We use Adam optimizer with a learning rate of 0.001 to train the model. We employ L2 regularization to improve generalization performance.

IV. EXPERIMENTS

In this section, we investigate the performance of the proposed methods. We perform two experiments to evaluate the proposed models. First, we compare the 2-D CNN models for feature extraction in two aspects: the number of parameters and classification performance. Second, to verify the capability of the multifeature fusion model, we compare the single-feature model with that of the aggregated model. The experiments are performed with two classification targets: large earthquake versus noise and large earthquake versus microearthquake versus noise.

A. Data Set

The seismic raw waveform was collected from January 2016 to July 2018 at the observatory stations

TABLE II
EXPERIMENT 1 RESULTS

LE vs. N	#params	ACC	TPR			FPR		
VGG-19	148.0M	99.77	97.95			0.11		
ResNet-50	23.5M	99.82	97.36			**0.01**		
Flatten	0.7M	99.85	98.25			0.04		
GAP(proposed)	**0.2M**	**99.88**	**98.70**			0.04		
LE vs. ME vs N	#params	ACC	TPR(LE)	TPR(ME)	TPR(N)	FPR(LE)	FPR(ME)	FPR(N)
VGG-19	148.0M	97.44	64.42	96.81	99.78	0.73	2.43	11.08
ResNet-50	23.5M	99.33	**97.16**	97.28	99.90	0.55	**0.28**	2.75
Flatten	0.7M	99.28	95.61	97.73	99.84	0.52	0.42	2.79
GAP(proposed)	**0.2M**	**99.42**	94.91	**98.18**	**99.98**	**0.33**	0.34	**2.62**

TABLE III
EXPERIMENT 2 RESULTS

LE vs. N	#params	ACC	TPR			FPR		
Pero et al.	0.04M	99.20	96.36			0.61		
Spectrogram	0.2M	99.88	98.70			0.04		
Recurrence Plot	0.2M	99.71	97.06			0.12		
S Transform	0.2M	99.77	97.36			0.08		
Aggregation(ResNet)	1.12M	99.90	98.75			**0.01**		
Aggregation(proposed)	1.04M	**99.91**	**99.20**			0.04		
LE vs. ME vs. N	#params	ACC	TPR(LE)	TPR(ME)	TPR(N)	FPR(LE)	FPR(ME)	FPR(N)
Pero et al.	0.04M	94.54	90.87	93.47	95.01	5.26	5.25	7.17
Spectrogram	0.2M	99.42	94.91	98.18	**99.98**	0.33	0.34	2.62
Recurrence Plot	0.2M	98.07	92.52	94.14	99.26	1.62	1.17	6.25
S Transform	0.2M	99.00	94.21	97.38	99.66	0.73	0.68	3.39
Aggregation(ResNet)	1.12M	99.49	**96.71**	98.20	99.94	0.36	**0.26**	2.17
Aggregation(proposed)	1.04M	**99.51**	96.01	**98.44**	99.97	**0.30**	0.28	**2.15**

in South Korea. We also collected large earthquake events from the Circum-Pacific belt that occurred in 2019. We used three sensor types: HH, EL, and HG. The sensor type HH is a broadband high-gain seismometer, EL is a short-period low-gain seismometer, and HG is a broadband gravimeter. Every sensor is recorded 100 Hz on three channels: E-oriented west-east, N-oriented north-west, and Z-oriented vertically. We cropped the signal by 10-s window starting 3 s prior to P-wave arrival time and by sliding the window three times by 1 s. The seismic events are divided into three types: large earthquakes (macroearthquakes with magnitude equal to or greater than 3.0), microearthquakes (small earthquakes with magnitude less than 2.0), and seismic noise. Table I indicates the size of each data set used for training and test. LE is the large earthquake, ME is the microearthquake, and N is the noise.

We used the magnitude spectrogram that has a 500-ms-long Tukey window function, 50% overlap, and nfft $= 256$. The vector dimension d of RP is 10. All 2-D representations are normalized in the range [0–1] and resized into 64×64 images.

B. Performance Measures

The classification performance is evaluated in terms of accuracy, true positive rate (TPR), and false positive rate (FPR). TPR means the potential of event detection, and FPR is related to false alarm. These measures are defined as

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{7}$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{8}$$

$$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} \tag{9}$$

where TP is the number of true positives, TN is the number of true negative, FP is the number of false positive, and FN is the number of false negatives.

C. Results

Initially, we assessed the computational cost and classification performance of the feature extraction model using a spectrogram. The results are presented in Table II. VGG-19 [24] and ResNet-50 [25] are the 2-D CNN structures that achieve good results in image classification problem. The Flatten model has the same CNN layers of the proposed CNN model, but the output of the last CNN layer is transformed into a vector representation using flattening instead of skip-connection and GAP. The GAP model is the proposed CNN model shown in Fig. 2. In the two-class classification, the proposed model showed better performance results than the VGG-19 model and the Flatten model with fewer parameters. The ResNet-50 model yields 0.03% better FPR results than others. In the three-class case, TPR, FPR for noise, and accuracy of the proposed model are at its competitive advantage, but the event classification performance is not better than that of ResNet-50. Consequently, the proposed CNN model achieves similar performance than the traditional image classification model with fewer parameters.

TABLE IV
AGGREGATION RESULTS

LE vs.ME vs.N	ACC	TPR(LE)	FPR(N)
S+R+ST+T	**99.51**	96.01	**2.15**
S+R+T	99.45	95.91	2.42
S+ST+T	99.46	**96.51**	2.26
R+ST+T	99.26	95.21	3.14
S+R+ST	99.51	95.96	2.22
S+R	99.42	95.80	2.63
S+ST	99.46	96.16	2.42
R+ST	99.27	93.71	2.92
S+T	99.37	94.41	2.83
R+T	98.54	93.81	5.78
ST+T	99.13	94.66	3.24

Second, we compare and analyze the performance of the proposed CNN model in 2-D representations. The experiments are summarized in Table III. As the results indicate, the model using the spectrogram provides better performance than RP and S transform. Especially, the spectrogram model is shown superior to that of the state-of-the-art model [1] by 1% in two-class classification and 4.7% in three-class classification on the average.

Finally, we evaluate the performance of the proposed aggregation model. As shown by the results in Table III, the aggregation model shows better performance than that of the single-feature-based models. The accuracy of the aggregation model is shown to achieve 0.9% higher than that of the spectrogram model, which has the best performance over the single model. Furthermore, we explore an effective aggregation model structure. In the Aggregation (ResNet) model, the first two CNN layers of the aggregation model are replaced by residual blocks. In this case, the number of parameters is increased, but the accuracy is not improved. In Table IV, we compare the performance of the several combinations of the employed features, where S is the spectrogram, R is RP, ST is the S transform, and T is the time signal feature. Every fusion model outperforms the single-feature-based model except S + T. The performance of the proposed model (S + R + ST + T) is equal or better than others. Consequently, the proposed fusion model using 2-D representations improves classification performance.

V. CONCLUSION

In this letter, we proposed a CNN structure for feature extraction with skip-connection and GAP and an aggregation model for multifeature-based earthquake event classification. We assessed various forms of 2-D representations of the seismic waveform for performance comparison. We also proposed DNNs architecture for feature extraction and aggregation. The proposed CNN model for feature extraction, which employed skip-connection and GAP, considered not only the mid-level feature but also the high-level feature of the architecture. The aggregation model combined several types of possible representation to improve estimation results. The experimental results indicated that the spectrogram, RP, and S transform are all valid 2-D representations for earthquake event classification. Moreover, the proposed CNN model with fewer parameters yielded the same or better performance than the competing state-of-the-art models. To conclude, our proposed earthquake event classification model yielded substantially high classification performance, making it a great candidate for deploying it to an earthquake early warning system.

REFERENCES

[1] T. Perol, M. Gharbi, and M. Denolle, "Convolutional neural network for earthquake detection and location," *Sci. Adv.*, vol. 4, no. 2, Feb. 2018, Art. no. e1700578.

[2] R. V. Allen, "Automatic earthquake recognition and timing from single traces," *Bull. Seismol. Soc. Amer.*, vol. 68, no. 5, pp. 1521–1532, 1978.

[3] C. D. Saragiotis, L. J. Hadjileontiadis, and S. M. Panas, "PAI-S/K: A robust automatic seismic p phase arrival identification scheme," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 6, pp. 1395–1404, Jun. 2002.

[4] C. Baillard, W. C. Crawford, V. Ballu, C. Hibert, and A. Mangeney, "An automatic kurtosis-based P- and S-phase picker designed for local seismic networks," *Bull. Seismol. Soc. Amer.*, vol. 104, no. 1, pp. 394–409, 2013.

[5] A. Reynen and P. Audet, "Supervised machine learning on a network scale: Application to seismic event classification and detection," *Geophys. J. Int.*, vol. 210, no. 3, pp. 1394–1409, Sep. 2017.

[6] J. R. Brown, G. C. Beroza, and D. R. Shelly, "An autocorrelation method to detect low frequency earthquakes within tremor," *Geophys. Res. Lett.*, vol. 35, no. 16, 2008.

[7] S. J. Gibbons and F. Ringdal, "The detection of low magnitude seismic events using array-based waveform correlation," *Geophys. J. Int.*, vol. 165, no. 1, pp. 149–166, Apr. 2006.

[8] C. E. Yoon, O. O'Reilly, K. J. Bergen, and G. C. Beroza, "Earthquake detection through computationally efficient similarity search," *Sci. Adv.*, vol. 1, no. 11, Dec. 2015, Art. no. e1501057.

[9] J. Wang and T.-L. Teng, "Artificial neural network-based seismic detector," *Bull. Seismol. Soc. Amer.*, vol. 85, no. 1, pp. 308–319, 1995.

[10] Y. Wu, Y. Lin, Z. Zhou, D. C. Bolton, J. Liu, and P. Johnson, "DeepDetect: A cascaded region-based densely connected network for seismic event detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 62–75, Jan. 2019.

[11] W. Zhu and G. C. Beroza, "PhaseNet: A deep-neural-network-based seismic arrival time picking method," *Geophys. J. Int.*, vol. 216, no. 1, pp. 261–273, Oct. 2018.

[12] M. Nakano, D. Sugiyama, T. Hori, T. Kuwatani, and S. Tsuboi, "Discrimination of seismic signals from earthquakes and tectonic tremor by applying a convolutional neural network to running spectral images," *Seismol. Res. Lett.*, vol. 90, no. 2A, pp. 530–538, Mar. 2019.

[13] Z. Zhang, Y. Lin, Z. Zhou, and T. Chen, "Adaptive filtering for event recognition from noisy signal: An application to earthquake detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 3327–3331.

[14] S. M. Mousavi, W. Zhu, Y. Sheng, and G. C. Beroza, "CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection," *Sci. Rep.*, vol. 9, no. 1, pp. 1–14, Dec. 2019.

[15] Z. Li, M.-A. Meier, E. Hauksson, Z. Zhan, and J. Andrews, "Machine learning seismic wave discrimination: Application to earthquake early warning," *Geophys. Res. Lett.*, vol. 45, no. 10, pp. 4773–4779, May 2018.

[16] R. G. Stockwell, L. Mansinha, and R. P. Lowe, "Localization of the complex spectrum: The S transform," *IEEE Trans. Signal Process.*, vol. 44, no. 4, pp. 998–1001, Apr. 1996.

[17] J. Eckmann *et al.*, "Recurrence plots of dynamical systems," *World Sci. Series Nonlinear Sci. Series A*, vol. 16, pp. 441–446, Sep. 1995.

[18] S. M. Mousavi, S. P. Horton, C. A. Langston, and B. Samei, "Seismic features and automatic discrimination of deep and shallow induced-microearthquakes using neural network and logistic regression," *Geophys. J. Int.*, vol. 207, no. 1, pp. 29–46, Oct. 2016.

[19] Y.-H. Wang *et al.*, "The tutorial: S transform," Graduate Inst. Commun. Eng., Nat. Taiwan Univ., Taipei, Taiwan, Tech. Rep., 2010.

[20] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167

[23] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*. [Online]. Available: http://arxiv.org/abs/1312.4400

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.