CNN-Based Audio Recognition in Open-set Domain

Hitham Jleed[1] and Martin Bouchard[1]

[1]*School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada*
h.jleed@ieee.org

Abstract
This paper presents a method that uses convolutional neural networks for audio recognition in an open-set scenario. The audio sounds in an open-set scenario are usually out of the training data distribution, which necessitates a model that can recognize the known classes while rejecting the unknown ones. We propose a convolutional approach for recognizing audio events, that can effectively address open-set recognition by adding inclusion probabilities of extreme value machines. Extensive experiments conducted showed that our proposed method outrivals representative existing methods under the open-set regime.

Keywords
Audio Recognition, Open-set Recognition, Convolutional Neural Networks

1. Introduction

The audio recognition process requires sufficient statistical information taken from previously observed data representations [1]. Traditional recognition systems rely on the hypothesis that all of the data for testing and training comes from the same database, often with an equivalent distribution. This type of recognition is called closed-set recognition. Several algorithms, including Convolutional Neural Networks (CNNs), have had a lot of success with machine learning applications, due to their simplicity, good performance, and probabilistic interpretation. However, in open-set recognition applications, the label set often expands as new classes occur during the test phase, in which the robustness of these methods can drastically weaken. To solve this problem, the boundaries around known classes must be well-defined, so the system can either reject the instance as a new class or label it as a member of one of the existing classes. In this paper, we tackle the problem of open-set audio recognition by utilizing the extreme value machine (EVM) with a convolutional neural network for robust audio recognition. In this paper, Section II summarizes the related work, section III illustrates the architecture, section IV discusses the evaluation metrics, and section V demonstrates the experimental results.

2. Related Work

CNNs with the SoftMax activation function have been used in closed-set recognition tasks and demonstrated an outstanding performance in many applications in literature, such as speech sound recognition [2], audio source identification [3], and environmental audio recognition [4]. However, traditional closed-set recognition methods have no way of rejecting data from previously unknown classifications. The perception of open-set recognition has attracted more interest in image recognition and computer vision researchers for both deep and shadow classifiers. Some non-deep efforts have been investigated in [5] for open-set image recognition. The work was then expanded in [6] by proposing the 1-vs-set machine approach to improve the robustness of image recognition. A Weibull-calibrated SVM was introduced in [7] for open-set image recognition. It was built for minimizing the empirical error and open space risk. An open-set algorithm called PSR-SVM was proposed in [8] to compute the posterior probability distribution for all classifier outputs, using a confidence measurement to determine whether a certain event belongs to a specific group of predefined events or not. Similarly, in radar image recognition, an automatic target recognition was published in [9], where open-set recognition was used on high-scale resolution in radar images. It formulated an automated target images recognition. For the deep open-set problem, a deep CNN for face recognition produced some promising results in [10]. Gutoski et al [11] introduced a human action recognition system using a 3D CNN that rejects inputs belonging to unknown classes. A deep CNN for environmental sound recognition was proposed in [12]

but it did not perform an open-set recognition well. In this paper, we adapt the EVM and meta-recognition [13] in the SoftMax activation layer function. This method measures the sample signal probability and detects a potential novel class.

3. Methodology

The proposed CNN architecture includes several convolutional layers [14]. The convolutional layers extract higher-level features for the final classification. We used 2D CNNs since they can capture the spatiotemporal information of the signal [15]. The output of the last convolutional layer is compressed into a 1D vector after the series of convolutional layers. The automatically generated feature vector is the result of this phase.

3.1. Preprocessing

Each audio signal is composed of different frequencies and different energy amplitudes, with quick variations within a short time. There is a need to define and represent audio signals such that a robust recognition system can be built. We have chosen the log-Mel spectrogram since it has proven to be suitable to model the human auditory system and is used in many speech and audio recognition tasks [16]. The Mel spectrogram is a spectrogram that converts frequencies to the Mel scale. It is computed by using a set of overlapping triangle filters to ascertain the energy of each spectral band. Audio features are obtained by computing 64 log-Mel bins with a window length of 1024 and a hop size of 500 samples at a 44.1 kHz sampling rate.

3.2. Network Architectures

The architecture block of the CNN is composed of convolution layers and pooling layers. A convolution layer applies filters to the input then takes the inner product and adds the bias. Each filter has its own bias and weights. A pooling layer reduces the dimensionality of the subsequent layers. It is applied to each convolution feature map independently. Please refer to [2] for a more detailed description. The input features (2D Mel spectrogram array) are organized to be fed into the CNN algorithm, each representing a small window of input audio signal for training or testing. Rectified Linear Unit (*ReLU)* activation functions are used in each convolutional layer, which imposes nonlinearity on the feature maps.

When we apply more convolution and pooling techniques to feature maps at higher levels, their resolution decreases. Before feeding the features to the output layer, they must be integrated across all frequency bands. On top of the last CNN layer, fully connected hidden layers are formed. The SoftMax is the output layer used as an activation function to predict probability over the class labels. Each number in the softmax function's output is inferred as the probability of belonging to each class. However, for open-set recognition, SoftMax cannot work well, so we propose to replace it with the EVM to determine the probability of the output for each class.

For closed-set recognition, let us assume the known classes $\{C_1, C_2, ..., C_N\}$, where Nk is the number of known classes. The final layer has the same size as the number of known classes. We denote the representation of this final network layer as $y = f(x)$, where f denotes the network as a function. When an audio data point x arrives, the SoftMax function to label this sound is defined as follows:

$$p(C_i \mid x, x \in N) = SoftMax_i(y) = \frac{\exp(x_i)}{\sum_j^N \exp(x_j)} \tag{1}$$

The SoftMax function assigns a certain probability to each training class by computing the maximum SoftMax probability, which is suitable for the optimization of the deep network in the closed-set recognition. In open-set settings, we need to consider $x \notin N$, where the class C_{N+1} corresponds to a novel class. The crucial step is to find a suitable value for thresholding between known and unknown classes. Some previous works such as [11] and [17] used test data distributions and thresholds values. In this work, we set a threshold by calibrating the activation vector with the inclusion probabilities of each

class, where the extreme-value theory indicates that the Weibull family of distributions is fit for this purpose [18]. To build a matched score distribution during training time, the distance between all training samples from a given class and its associated class mean μ is calculated using some distance functions, such as, Euclidean, hybrid, and cosine distance. Then, a Weibull distribution is equipped to the tail of the matched distribution. We used the *libmr* library [13] to compute the parameters in the Weibull distribution, whose values of hyperparameters were taken as suggested in [19].

4. Evaluation Metrics

We assess the effectiveness of our proposed algorithm by computing similarities after aligning the recognition outputs with a reference ground truth. The evaluation utilizes cross-validation, which allows evaluation of the accuracy of data that may not be part of the training dataset. Two fundamental assumptions have been used in the DCASE/ AASP challenges [20] to evaluate how individual audio sounds are classified:

- Segment-based evaluation: the system output and ground truth are compared for each segment length.
- Event-based evaluation: the system output is considered the same within all ranges (duration) of the event. This means that event labels in the recognition output will be compared to the ground truth events.

Let us consider a binary classification, where the labels consist only of positives or negatives. Based on true labels and predicted labels, we divide the metrics into four intermediate statistics: true-positive (*TP*), false-positive (*FP*), true-negative (*TN*), and false-negative (*FN*). A count is made for each category. Applying this to a multi-class problem, every single classifier that produces a "positive" or "negative" prediction can be "true" or "false" depending on the corresponding ground-truth label.

4.1. Recognition Accuracy

The recognition accuracy (*RA*) can be described as the ratio of the correctly labeled predictions to the whole pool:

$$RA = \frac{TP + TN}{TP + FP + FN + TN} \tag{2}$$

4.2. Precision And Recall

Precision (*Pr*) is the ratio of predicted positive samples that are calculated correctly (true) divided by all predicted positive samples, while recall (*Re*) is the fraction of predicted positive samples correctly detected from all ground truth positive samples (labels). For multi-class classification, there are two ways of computation: macro-averaging and micro-averaging [21].

$$Pr_{macro} = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i} \qquad Re_{macro} = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i} \tag{3}$$

$$Pr_{micro} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FP_i)} \qquad Re_{micro} = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N}(TP_i + FN_i)} \tag{4}$$

4.3. F1-measure

The F1-measure includes both precision and recall merged in a single score, which is computed as the harmonic mean between precision and recall. The F1-measure is computed as:

$$F1\text{-}measure = \frac{2PrRe}{Pr + Re} \tag{5}$$

4.4. Confusion Matrix

The confusion matrix $CM(i, j)$ summarizes the performance of multi-class recognition. It depicts the various ways in which the classification model gets confused when making predictions. Each column of the matrix represents a real class, whereas each row represents a predicted class. The diagonal of this matrix $(i = j)$ reveals the correct prediction.

5. Experiments

Extensive experiments were conducted using Python. A Keras [22] implementation of CNNs was used, with TensorFlow [23] as the backend. First, we carried out closed-set recognition experiments where the audio dataset is separated into training and testing datasets.

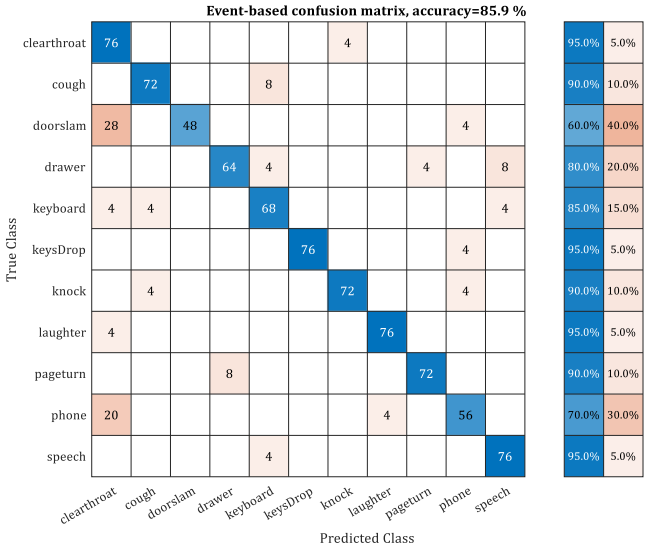

Figure 1: Confusion matrix of event-based recognition

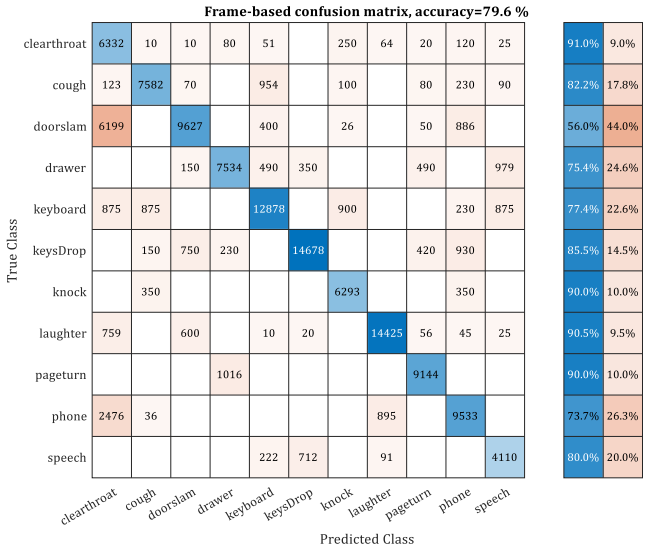

Figure 2: Confusion matrix of frame-based recognition

To model the classifiers, we applied the 5-folding cross-validation technique where a total of 80% is used as the training dataset, while the remaining 20% of the data is used for testing. The experiment is conducted on the DCASE2016 dataset. This dataset consists of audio recorded in everyday life, which includes 11 sound classes that were recorded in an office environment: clearing throat, coughing, speech, drawer, keyboard, keys drop, knock, laughter, page-turning, phone ringing, and a door slam.

5.1. Closed-set Recognition

The classification output is evaluated to be correct or not according to the ground truth. We did not perform comparisons with other algorithms in this part, because the experiments in a closed-set regime aim to evaluate the ability of the algorithm to differentiate among recognized classes. The comparison will be conducted later in the open-set recognition part. As can be noticed from Fig. 1, the event-based confusion matrix discloses that most of the classes have been recognized very well except door-slam and phone-ring classes, whose accuracies were 60% and 70 %, respectively. Fig. 2 shows the confusion matrix after applying frame-based recognition. The right column reveals the percentage accuracy of each class. Most of the classes have been recognized correctly, and some misclassification can be also observed that is because the similarities among these classes are high. The sound class has a great impact on the results, as expected. For example, the door-slam class was the hardest class to recognize, probably because of the short length of such sounds.

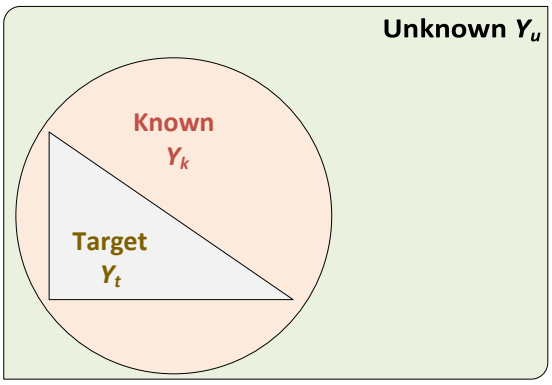

Figure 3: Venn diagram of acoustic classes.

5.2. Open-set Recognition

The experiments in this section were performed to recognize audio sounds where the testing set also includes classes that may not be part of the training dataset. These experiments measure the capability to discriminate known classes from novel classes and to discriminate known classes from one another. The level of openness for a classification task can be defined as:

$$Openness = 1 - \sqrt{\frac{2 \times |Y_t|}{|Y_k| + |Y_u|}} \tag{6}$$

Where the subscripts k, t and u are for the known, target, and unknown label sets, as defined in Fig. 3. The testing dataset is defined as $Y_{test} = Y_k \cup Y_t \cup Y_u$, while the training dataset is the combination $Y_{train} = Y_t \cup Y_k$, and the unknown classes are its complement $Y_u = \{ y \mid y \notin Y_{train} \text{ and } y \in Y_{test} \}$. If we set $Y_k = Y_t$ these yields $Y_{train} = Y_k$.

We used varying degrees of openness and followed k-fold cross-validation to obtain robust evaluation metrics. The experiments were performed by generating different amounts of openness. Our experiments were conducted for several evaluations in which we examined how well our proposed algorithm performs in comparison to other representative algorithms: W-SVM [6], IOmSVM [8], OSNN [24], and OSmIL [25].

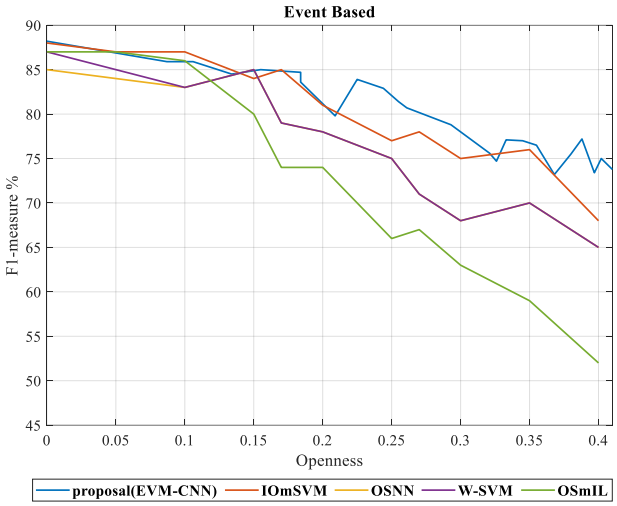

Figure 4: F1-measure as a function of openness for open-set recognition. Results computed for event-based metrics.

The parameters of all previous algorithms were set according to the corresponding paper. To ensure a fair comparison, all of the algorithms were run on the same dataset and the same distribution of classes.

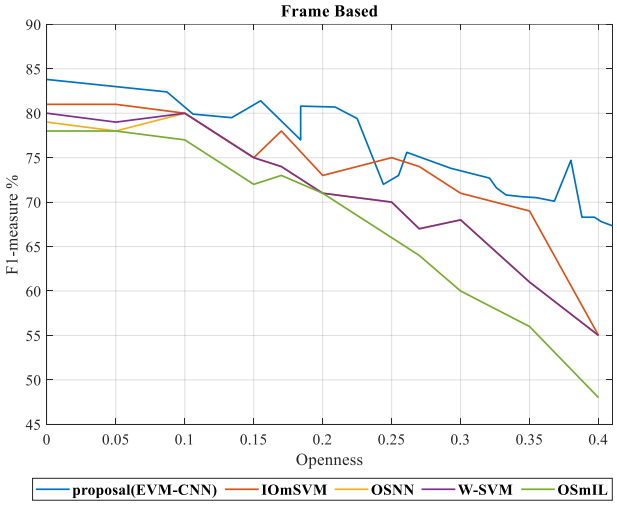

Figure 5: F1-measure as a function of openness for open-set recognition. Results computed for frame-based metrics.

The experimental results are depicted in Fig.4 for event-based recognition and in Fig.5 for frame-based recognition. It is clear that there are considerable differences in the performance among the different methods. Looking at these figures, all the methods, in general, suffer from a performance decrease if the openness increases. However, our proposed algorithm performs relatively well compared to the other methods, in terms of determining the novel classes and discriminating among the known classes. The OSmIL algorithm had the worst performance. As expected, the performance of event-based measures outperforms the performance of frame-based measurements.

6. Conclusion

In this work, we presented a CNN network architecture that is efficient for robust audio open-set recognition. Extensive testing was done to distinguish between known and unknown audio classes. Our proposed method overall outperformed representative previous work across a wide range of openness levels. For further work, more research should be done to see how well the proposed CNN performs on

large real-world audio datasets. Experiments and algorithmic modifications for incremental learning should also be performed.

7. References

[1] S. Marsland, *Machine Learning : An Algorithmic Perspective, Second Edition*. Chapman and Hall/CRC, 2014. doi: 10.1201/b17476.

[2] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEEACM Trans. Audio Speech Lang. Process.*, vol. 22, no. 10, pp. 1533–1545, 2014.

[3] E. M. Grais, H. Wierstorf, D. Ward, and M. D. Plumbley, "Multi-resolution fully convolutional neural networks for monaural audio source separation," in *International Conference on Latent Variable Analysis and Signal Separation*, 2018, pp. 340–350.

[4] W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very Deep Convolutional Neural Networks for Raw Waveforms," *ArXiv161000087 Cs*, Oct. 2016, Accessed: Feb. 28, 2022. [Online]. Available: http://arxiv.org/abs/1610.00087

[5] W. J. Scheirer, A. D. R. Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, Jul. 2013, doi: 10.1109/TPAMI.2012.256.

[6] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability models for open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2317–2324, Nov. 2014, doi: 10.1109/TPAMI.2014.2321392.

[7] L. P. Jain, W. J. Scheirer, and T. E. Boult, "Multi-class open set recognition using probability of inclusion," in *Computer Vision – ECCV 2014*, vol. 8691, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 393–409. doi: 10.1007/978-3-319-10578-9_26.

[8] H. Jleed and M. Bouchard, "Open set audio recognition for multi-class classification with rejection," *IEEE Access*, vol. 8, pp. 146523–146534, 2020, doi: 10.1109/ACCESS.2020.3015227.

[9] J. D. Roos and A. K. Shaw, "Probabilistic SVM for open set automatic target recognition on high range resolution radar data," Anaheim, California, United States, May 2017, p. 102020B. doi: 10.1117/12.2262840.

[10] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," 2015.

[11] M. Gutoski, A. E. Lazzaretti, and H. S. Lopes, "Deep metric learning for open-set human action recognition in videos," *Neural Comput. Appl.*, Jun. 2020, doi: 10.1007/s00521-020-05009-z.

[12] Z. Mushtaq and S.-F. Su, "Environmental sound classification using a regularized deep convolutional neural network with data augmentation," *Appl. Acoust.*, vol. 167, p. 107389, Oct. 2020, doi: 10.1016/j.apacoust.2020.107389.

[13] W. J. Scheirer, A. Rocha, R. J. Micheals, and T. E. Boult, "Meta-Recognition: The Theory and Practice of Recognition Score Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1689–1695, Aug. 2011, doi: 10.1109/TPAMI.2011.54.

[14] N. K. Manaswi, *Deep Learning with Applications Using Python*. Berkeley, CA: Apress, 2018. doi: 10.1007/978-1-4842-3516-4.

[15] S. Hershey *et al.*, "CNN Architectures for Large-Scale Audio Classification," *ArXiv160909430 Cs Stat*, Jan. 2017, Accessed: Mar. 01, 2022. [Online]. Available: http://arxiv.org/abs/1609.09430

[16] H. Meng, T. Yan, F. Yuan, and H. Wei, "Speech Emotion Recognition From 3D Log-Mel Spectrograms With Deep Learning Network," *IEEE Access*, vol. 7, pp. 125868–125881, 2019, doi: 10.1109/ACCESS.2019.2938007.

[17] H. Oliveira, C. Silva, G. L. S. Machado, K. Nogueira, and J. A. dos Santos, "Fully convolutional open set segmentation," *Mach. Learn.*, Jul. 2021, doi: 10.1007/s10994-021-06027-1.

[18] E. M. Rudd, L. P. Jain, W. J. Scheirer, and T. E. Boult, "The Extreme value machine," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 762–768, Mar. 2018, doi: 10.1109/TPAMI.2017.2707495.

[19] A. Bendale and T. E. Boult, “Towards open set deep networks,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 1563–1572. doi: 10.1109/CVPR.2016.173.
[20] A. Diment, T. Heittola, and T. Virtanen, “Sound event detection for office live and office synthetic AASP challenge,” *Proc IEEE AASP Chall. Detect. Classif Acoust Scenes Events WASPAA*, 2013, Accessed: Nov. 11, 2016. [Online]. Available: http://c4dm.eecs.qmul.ac.uk/sceneseventschallenge/abstracts/OL/DHV.pdf
[21] K. Zhang, H. Su, and Y. Dou, “Beyond AP: a new evaluation index for multiclass classification task accuracy,” *Appl. Intell.*, vol. 51, no. 10, pp. 7166–7176, Oct. 2021, doi: 10.1007/s10489-021-02223-7.
[22] J. Moolayil, *Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python*. Berkeley, CA: Apress, 2019. doi: 10.1007/978-1-4842-4240-7.
[23]M. Abadi *et al.*, “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” *ArXiv Prepr. ArXiv160304467*, 2016.
[24] P. R. Mendes Júnior *et al.*, “Nearest neighbors distance ratio open-set classifier,” *Mach. Learn.*, vol. 106, no. 3, pp. 359–386, Mar. 2017, doi: 10.1007/s10994-016-5610-8.
[25] S. Dang, Z. Cao, Z. Cui, Y. Pi, and N. Liu, “Open set incremental learning for automatic target recognition,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4445–4456, Jul. 2019, doi: 10.1109/TGRS.2019.2891266.