
Raw Waveform-based Audio Classification Using Sample-level CNN Architectures

Jongpil Lee

Graduate School of Culture Technology
KAIST
richter@kaist.ac.kr

Taejun Kim

School of Electrical and Computer Engineering
University of Seoul
ktj7147@uos.ac.kr

Jiyoung Park

Graduate School of Culture Technology
KAIST
jypark527@kaist.ac.kr

Juhan Nam

Graduate School of Culture Technology
KAIST
juhannam@kaist.ac.kr

Abstract

Music, speech, and acoustic scene sound are often handled separately in the audio domain because of their different signal characteristics. However, as the image domain grows rapidly by versatile image classification models, it is necessary to study extensible classification models in the audio domain as well. In this study, we approach this problem using two types of sample-level deep convolutional neural networks that take raw waveforms as input and uses filters with small granularity. One is a basic model that consists of convolution and pooling layers. The other is an improved model that additionally has residual connections, squeeze-and-excitation modules and multi-level concatenation. We show that the sample-level models reach state-of-the-art performance levels for the three different categories of sound. Also, we visualize the filters along layers and compare the characteristics of learned filters.

1 Introduction

Broadly speaking, audio classification tasks are divided into three sub-domains including music classification, speech recognition (particularly for the acoustic model), and acoustic scene classification. However, input audio features and models for each sub-domain task are usually different due to the different signal characteristics.

Recent advances in deep learning have encouraged a single audio classification model to be applied to many cross-domain tasks. For example, Adavanne et. al. used a Convolutional Recurrent Neural Network (CRNN) model for sound event detection [1], bird audio classification [2] and music emotion recognition [3]. However, depending on the task, the majority of the audio classification models use different sub-optimal settings of time-frequency representation as input in terms of filter-bank type and size, time-frequency resolution and magnitude compression. This in turn influences model architecture, for example, the choices of convolutional layer (1D or 2D) and filter shape [4].

This issue can be solved by a waveform-based model that directly takes raw input signals. Recently, Dieleman and Schrauwen used raw waveforms as input of CNN models for music auto-tagging task [5]. Sainath et. al. used Convolutional Long short-term memory Deep Neural Network (CLDNN) for speech recognition [6]. Dai et. al. used Deep Convolutional Neural Networks (DCNN) with residual connections for environmental sound recognition [7]. All of them used frame-level filters (typically several hundred samples long) in the first convolutional layer which were carefully configured to

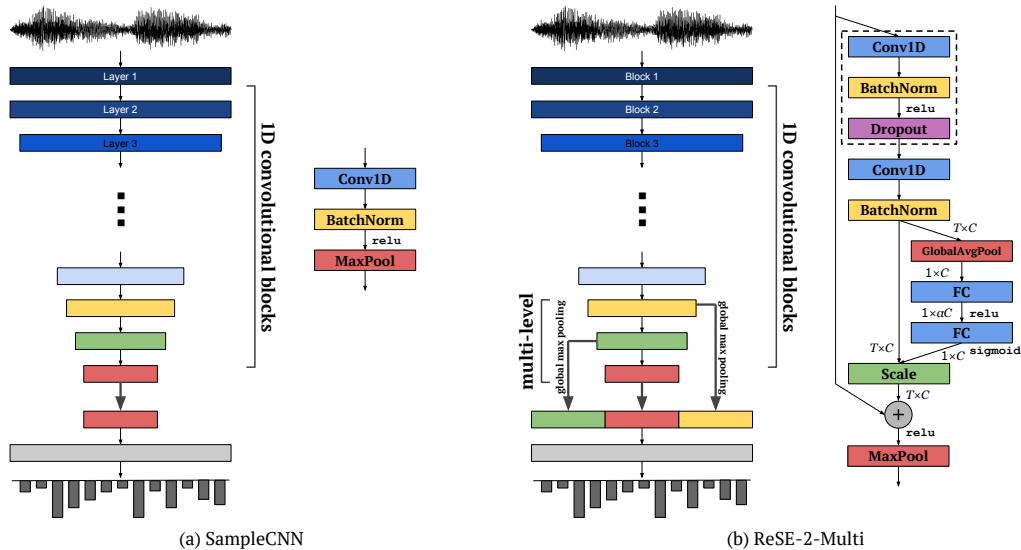


Figure 1: SampleCNN and ReSE-2-Multi models.

handle the target task. In this frame-level raw waveform input, however, the filters in the bottom layer should learn all possible phase variation of (pseudo-)periodic waveforms which are likely to be prevalent in audio signals. This has impeded the use of raw waveform as input over spectrogram-based representations where the phase variation within a frame (i.e. time shift of periodic waveforms) is removed by taking the magnitude only.

The phase invariance is analogous to translation invariance in the image domain. Considering that the filter size is typically small in the image domain, even 3×3 in the VGG model [8], we investigated the possibility of stacking very small-size filters from the bottom layer of DCNN for raw audio waveforms with max-pooling layers. The results on music auto-tagging [9] and sound event detection [10] showed that the VGG-style 1-D CNN models are highly effective. We term this model as **SampleCNN**.

We enhanced the SampleCNN model by adding residual connections, squeeze-and-excitation modules and multi-level feature concatenation for music auto-tagging [11]. The residual connection makes gradient propagation more fluent, allowing training deeper networks [12]. The Squeeze-and-Excitation (SE) module recalibrates filter-wise feature responses [13]. The multi-level feature concatenation takes different abstraction levels of classification labels into account [14]. We term this model as **ReSE-2-Multi**.

In this study, we show that the sample-level CNN models are effective for three datasets from different audio domains. Furthermore, we visualize hierarchically learned filters for each dataset in the waveform-based model to explain how they process sound differently.

2 Models

Figure 1 shows the structures of the two sample-level models. 2 or 3 sample-size 1D filters and poolings are used in all convolutional layers. In SampleCNN, convolutional layer, batch normalization layer and max-pooling layer are stacked as shown in Figure 1 (a). The detailed description can be found in [9].

In ReSE-2-Multi, we add a residual connection and an SE module onto the SampleCNN building block as shown in Figure 1 (b). The SE path recalibrates feature maps through two operations. One is squeeze operation that aggregates a global temporal information into filter-wise statistics using global average pooling. The operation reduces the temporal dimensionality (T) to one. The other is excitation operation that adaptively recalibrates each filter map using the filter-wise statistics from the squeeze operation and a simple gating mechanism. The gating consists of two fully-connected (FC)

Table 1: Description of the three datasets, models and results.

	Music	Speech	Scene sound
Dataset	MagnaTagATune (MTAT) [15]	Speech Commands Dataset [16, 17] (TensorFlow Speech Recognition Challenge)	DCASE 2017 Task 4 [18] (subtask A)
Task	Music auto-tagging	Speech command recognition	Acoustic scene tagging
# of classes	50 tags	10 commands + "silence" + "unknown" (31 classes in training / 12 classes in testing)	17 sound events
Labels	Multi-label	Multi-class	Multi-label
Sampling rate	22,050Hz	16,000Hz	44,100Hz
Dataset split (train/valid/test)	15,244 / 1529 / 4332	57,929 / 6798 / 30% of 158,538 (public leaderboard test setting)	45,313 / 5859 / 488 (development set)
Duration	29 seconds	1 second	10 seconds
Description	Collected using the TagATune game and music from Magnatune.	Single-word speaking commands, rather than conversational sentences	Subset of AudioSet [19], YouTube clips focusing on vehicle and warning sounds.
Model	(resampled to 16,000Hz)		(resampled to 16,000Hz)
Input size	39,366 samples, 2.46 sec	16,000 samples, 1 sec	19,683 samples, 1.23 sec
# of segments	12 segments per clip	1 segments per clip	9 segments per clip
# of blocks	9 blocks	8 blocks	8 blocks
Results	AUC	Accuracy	F-score (instance-based)
SampleCNN	0.9033	84%	38.9%
ReSE-2+Multi	0.9091	86%	45.1%
State-of-the-art	0.9113 [11]	88% (as of Nov 29, 2017) [17]	57.7% [20]

layers that compute nonlinear interactions among filters. Then, the original outputs from the basic block are rescaled by filter-wise multiplication with the sigmoid activation of the second FC layer of the SE path. We also added residual connections to train a deeper model. The digit in the model name, ReSE-2-Multi, indicates the number of convolution layers in one building block. Finally, we concatenate three hidden layers to take account of different levels of abstraction.

3 Datasets and Results

We validate the effectiveness of the proposed models on music auto-tagging, speech command recognition and acoustic scene tagging. The details about the datasets for the tasks are summarized in Table 1. Note that we resampled all audio samples to 16,000Hz in order to verify how extensible the models are for the three sub-domain of audio tasks in the same condition. However, we configured the input size of the models for each dataset to commonly used size in each domain. Then, we set the number of building blocks according to the input size of the models. We averaged the prediction score for all segments of one clip in testing phase if the input size of the model is shorter than the duration of the clip.

Table 1 compares the results from the two sample-level CNN models. The performances are reported using commonly used evaluation metrics for each task. We also compare them to state-of-the-art performance on each dataset. In general, the ReSE-2-Multi model shows close performance to the state-of-the-art results except for the DCASE task. However, in [20], they used data balancing, ensemble networks and auto thresholding techniques. Without those techniques, they report that their CRNN model achieved 42.0% F-score value which is lower than our result with ReSE-2-Multi. Also, for the music auto-tagging task on MTAT, the state-of-the-art result was achieved by the ReSE-2-Multi model. In this case, the performance degradation is seen to be caused by downsampling to 16,000Hz.

4 Filter Visualizations

Visualizing the filters at each layer allows better understanding of representation learning in the hierarchical networks. Since both models yielded similar patterns of learned filters at each layer, we visualize them only for the sampleCNN model. Figure 2 shows the filters obtained by an activation maximization method [21]. To show the patterns more clearly, we visualized them as spectrum in the frequency domain and sorted them by the frequency at which the magnitude is maximum [9]. In this case, we set the size of the initial random noise to 729 ($= 3^6$) samples, so that the estimated

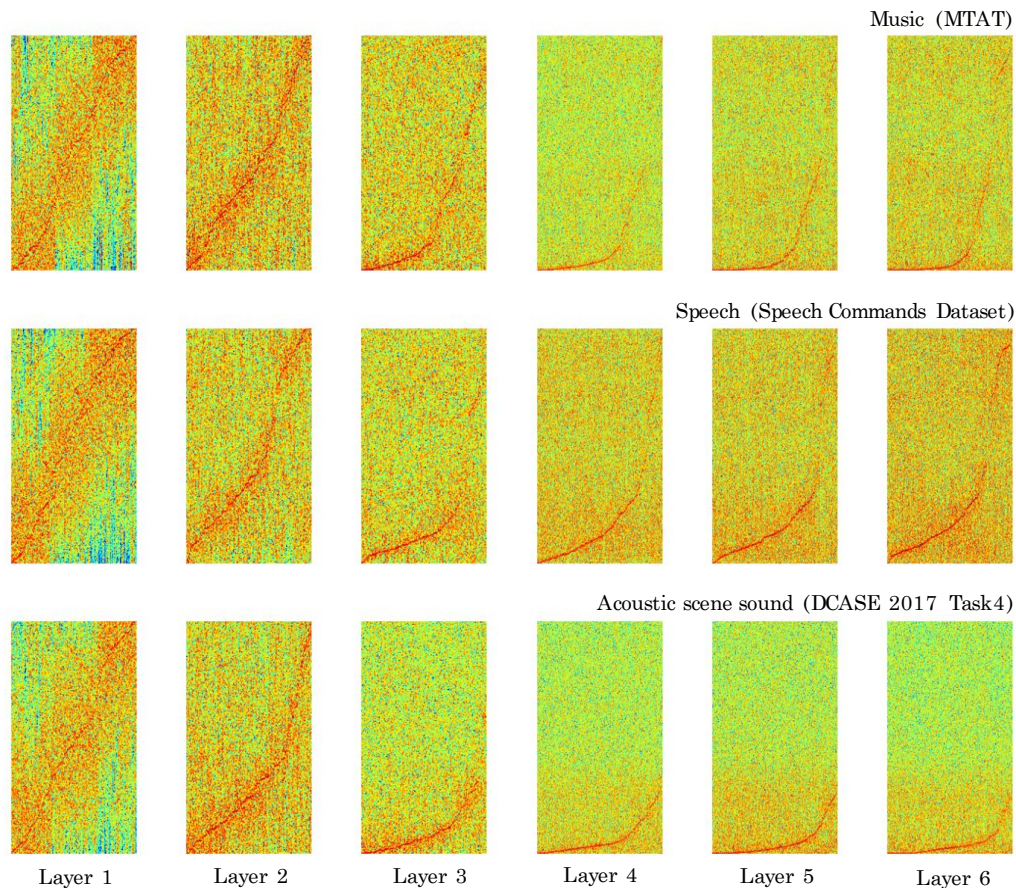


Figure 2: The spectrum of learned filter estimates for the three datasets in SampleCNN. They are sorted by the frequency at which the magnitude is maximum. The x-axis represents the index of the filters and the y-axis represents the frequency (ranging from 0 to 8000Hz for all figures). The visualizations were obtained using a gradient ascent method that finds the input waveform that maximizes the activation of a filter at each layer.

filters have typical frame-sized shape which will make the spectrum clearer. Also, for the first 6 layers we only used 3-sized filters and sub-sampling layers. Thus, the temporal dimension of the 6th layer output becomes one in this configuration. For other layers, we averaged remaining temporal dimension so as to make a single activation loss value. Finally, we conducted log-based magnitude compression on the spectrum.

From the figure, we can first find that they are sensitive to log-scaled frequency along the layers, such as mel-frequency spectrogram that is widely used in audio classification tasks. Second, when comparing acoustic scene sound with the other domains, the learned filters tend to have more low-frequency concentration and less complex patterns. This is probably because the DCASE task 4 dataset is made up of simple traffic and warning sounds. Finally, between music and speech, we can observe that more filters explain low-frequency content in music than speech.

5 Conclusions

We presented the two sample-level CNN models that directly take raw waveforms as input and have filters with small granularity. We evaluated them on three audio classification tasks. The results show the possibility that they can be applied to different audio domains as a true end-to-end model. As future work, we will investigate more filter visualization techniques to have better understanding of the models.

References

- [1] Sharath Adavanne and Tuomas Virtanen. Sound event detection using weakly labeled dataset with stacked convolutional and recurrent neural network. In *DCASE Workshop*, 2017.
- [2] Sharath Adavanne, Konstantinos Drossos, Emre Çakır, and Tuomas Virtanen. Stacked convolutional and recurrent neural networks for bird audio detection. In *EUSIPCO*, 2017.
- [3] Miroslav Malik, Sharath Adavanne, Konstantinos Drossos, Tuomas Virtanen, Dasa Ticha, and Roman Jarina. Stacked convolutional and recurrent neural networks for music emotion recognition. In *Sound and Music Computing Conference (SMC)*, 2017.
- [4] Jordi Pons, Thomas Lidy, and Xavier Serra. Experimenting with musically motivated convolutional neural networks. In *Content-Based Multimedia Indexing (CBMI) Workshop*, pages 1–6, 2016.
- [5] Sander Dieleman and Benjamin Schrauwen. End-to-end learning for music audio. In *IEEE ICASSP*, pages 6964–6968, 2014.
- [6] Tara N Sainath, Ron J Weiss, Andrew Senior, Kevin W Wilson, and Oriol Vinyals. Learning the speech front-end with raw waveform cldnns. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.
- [7] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das. Very deep convolutional neural networks for raw waveforms. In *IEEE ICASSP*, pages 421–425, 2017.
- [8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.
- [9] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam. Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. In *Sound and Music Computing Conference (SMC)*, 2017.
- [10] Jongpil Lee, Jiyoung Park, Sangeun Kum, Youngho Jeong, and Juhan Nam. Combining multi-scale features using sample-level deep convolutional neural networks for weakly supervised sound event detection. In *DCASE Workshop*, 2017.
- [11] Taejun Kim, Jongpil Lee, and Juhan Nam. Sample-level cnn architectures for music auto-tagging using raw waveforms. *arXiv preprint arXiv:1710.10451*, 2017.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE CVPR*, pages 770–778, 2016.
- [13] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507*, 2017.
- [14] Jongpil Lee and Juhan Nam. Multi-level and multi-scale feature aggregation using pre-trained convolutional neural networks for music auto-tagging. *IEEE Signal Processing Letters*, 24(8):1208–1212, 2017.
- [15] Edith Law, Kris West, Michael I Mandel, Mert Bay, and J Stephen Downie. Evaluation of algorithms using games: The case of music tagging. In *ISMIR*, pages 387–392, 2009.
- [16] Pete Warden. Speech commands: A public dataset for single-word speech recognition. *Dataset available from http://download.tensorflow.org/data/speech_commands_v0.01.tar.gz*, 2017.
- [17] TensorFlow Speech Recognition Challenge. <https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/leaderboard>.
- [18] Annamaria Mesaros, Toni Heittola, Aleksandr Diment, Benjamin Elizalde, Ankit Shah, Emmanuel Vincent, Bhiksha Raj, and Tuomas Virtanen. Dcase 2017 challenge setup: tasks, datasets and baseline system. In *DCASE Workshop*, 2017.
- [19] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *IEEE ICASSP*, 2017.
- [20] Yong Xu, Qiuqiang Kong, Wenwu Wang, and Mark D Plumbley. Surrey-cvssp system for dcase2017 challenge task4. *DCASE Tech. Rep.*, 2017.
- [21] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341:3, 2009.