ACOUSTIC SCENE CLASSIFICATION BASED ON CONVOLUTIONAL NEURAL NETWORK

Lu Lu[1,3],Jiang Yuzhi[1,2],Zhang Huiyu[1],Yang Yuhong[1,3],Hu Ruimin[1,2],Ai Haojun[1,3],Tu Weiping[1],Huang Weiyi[1]*

[1] National Engineering Research Center for Multimedia Software,
Wuhan University, Wuhan 430072, Hubei, China;
[2] The Key Laboratory of Multimedia and Network Communication Engineering,
Wuhan University, Wuhan 430072, Hubei, China;
[3] Collaborative Innovation Center of Geospatial Technology, China.

ABSTRACT

In this paper, we present four approaches on Task1 Acoustic Scene Classification(ASC): a simple CNN with low time-complexity, a novel feature extraction, a feature fusion, and a finetune pre-training. First, we propose a simplified CNN architecture with only two convolutional layers to avoid overfitting. The model had a balance between higher accuracy and lower time-complexity. Second, we extract identifiable audio features by a data-driven spectrogram down-sampling. Third, we do feature fusion by combining data-driven features with Mel-Frequency spectrogram(MFS) as the network input. Fourth, we finetune pre-training model based on two semantic levels. All the four approaches improve classification accuracy, compared with baseline on the development set.

Index Terms— audio scene classification, feature extraction, convolutional neural networks, deep learning, late fusion

1. INTRODUCTION

In this report, we describe four methods for Task 1(ASC) in the DCASE-2017 challenge 1. We provide the performances of our methods on the openly accessible DCASE-2017 dataset[1]. In our challenge submissions, we follow 4 different approaches for audio scene classification. First, we devise a simpler Convolutional Neural Network (CNN) in balance of higher accuracy and lower time-complexity. Second, we propose a Across Scenes Frequency Standard Deviation based Spectrogram Image Feature (ASFSTD-SIF) features extraction scheme. Third, we do feature fusion by combining traditional MFS features with ASFSTD-SIF feature. Fourth, we finetune weights based on pre-training model. This report is organized as follows: Section 2 explains our proposed methods by four aspects, as illustrated in Fig. 1; Section 3 presents the results of ASC on the provided dataset and cross-validation splits.

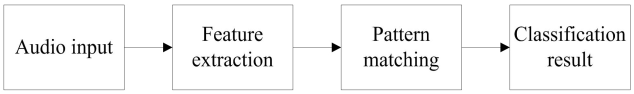

Figure 1: Acoustic scene classification system block diagram

2. PROPOSED METHOD

2.1. MFS and CNN

1-D Audio streams need to be converted into 2-D representations before feeding into the CNN. In this section, we choose the log-Mel spectrogram as 2-D feature inputs. First, the 10s-stereo recording are converted into mono, by down-mixing the right and left channels. Then we performed the fast Fourier transform (FFT) to 40ms frames with 50% overlap. The log Mel-spectrogram is calculated for all audio using 100 Mel bands with a maximum frequency of 22.05 kHz. Each 10s clip of the DCASE audio is split into 5 sub-clips of 100 frames (2 seconds) duration without overlapping. The input of CNN is 100×100 Mel-spectrogram. We subtract the mean from the data and divide them with the standard deviation, hence the data have a zero-mean and unit-variance. Test data is standardized by using the statistics from the training data. Thus, we have the Mel-spectrogram of each audio.

The feature extraction has been implemented in Python by using the librosa library.

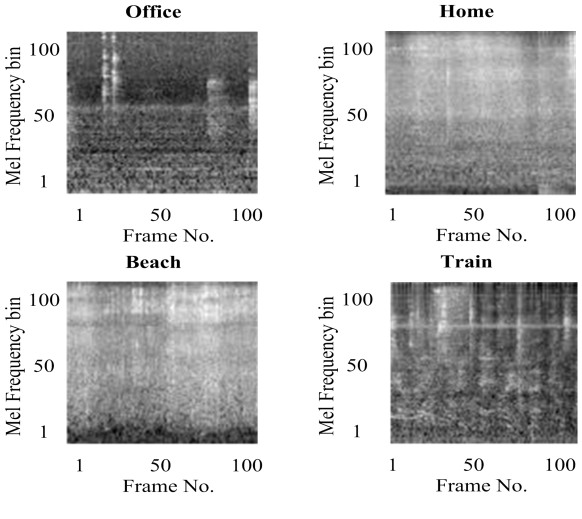

Figure 2: Samples of CNN input Mel-spectrogram

In Figure 2, we show some input samples of 100×100 log Mel-spectrogram. Mel-spectrograms present textures with horizontal and vertical stripes. The horizontal usually refers to acoustic backgrounds' spectrum, and the vertical refers to random occurrence of audio events.

In all the CNN-based submissions in DCASE2016, we find deeper CNN don't always get better performance. While in ASC, Eghbal-Zadeh et al.[2] achieved an accuracy of 83.3% by using a VGG-style CNN approach with 8 convolutional layers. Another VGG-style deep CNN with 8 convolutional layers with data augmentation obtained an accuracy of 84.6%[3]. And we also find the shallow CNN can achieve comparable performance than VGG-style deeper CNN. The CNN proposed by Valenti[4] with only 2 convolutional layers achieved an accuracy of 86.2 %. Hence, we assumed that the shallow CNN may be more effective in ASC task, due to the simplicity texture in audio spectrograms images. Shallow CNN can have achieved same or better accuracy with low time complexity.

Under this guideline， We formed a ConvNet architecture using only 2 convolution layers with max-pooling after each convolution layers. We increased the number of filters in convolution layer from 32 to 64 and added zero-padding before convolution layers to make full use of edges. Through the experiments, all convolutional layers are with RELU and a stride of 1. As for pooling strategy, all of them are 3×3 max-pooling with a stride 3, the dropout layer was preserved after each pooling layer with the rate of 0.25 except for the last global average pooling. overall architecture is presented in figure 3.

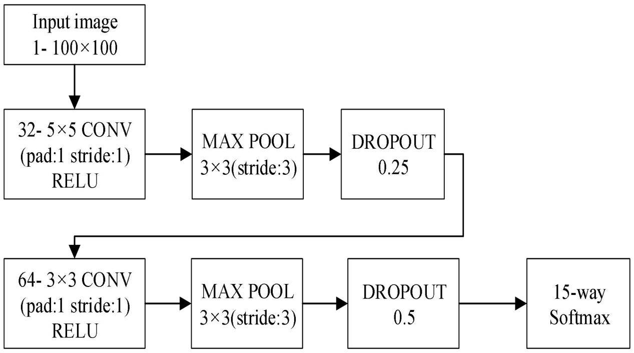

Figure 3: CNN architecture

2.2. Across Scenes Frequency Standard Deviation Based SIF

The acoustic scene contains a variety of sounds, the Mel spectrum represent spectral envelope information, but some burst sounds, such as bird calls with narrow spectrum peaks cannot be effectively expressed. Therefore, we propose a data-driven down-sampling approach for spectral graphs. The main idea is to calculate the logarithmic Magnitude Spectrum (LMS) of the original audio; then get the standard deviation (STD) of the energy at each frequency point for each scene category; finally, calculate the standard deviation between classes. The block-diagram of our ASFSTD-SIF is shown in Figure 4.

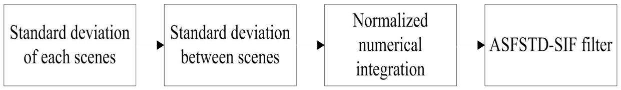

Figure 4: block-diagram of our ASFSTD-SIF

2.2.1. Calculate the standard deviation of the frequency domain

First, we calculate the normalized log spectrum features for all the audio files in the training set; then get intra-scene statistics. The number of scene categories is M=15, the whole spectral bin number is B (FFT inputs is 2048 points, so B=1025), the intra-scene standard deviation is computed as follows:

$$\sigma_{sj} = [\frac{1}{N}\sum_{i=1}^{N_j}\left(B_{ji}^{d} - \overline{B}^{d}\right)]^{\frac{1}{2}}, d = 1,2,..., B; j = 1,2,..., M \tag{1}$$

Where j is the index of the M-type scene, N_j is the number of samples in the j-th acoustic scene, B_{ji} is the logarithmic amplitude spectrum of the i-th sample in the jth scene, d is the subscript of the frequency. The set of training sets can be calculated as follows:

$$\overline{B}^{d} = \frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N_j} B_{ji}^{d}, d = 1,2,..., B \tag{2}$$

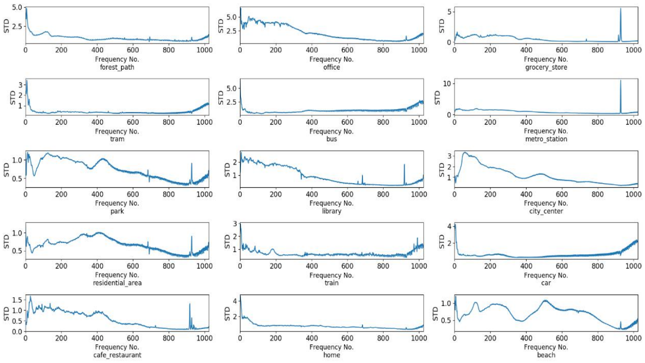

Figure 5: Standard deviation of 15 scenes in the frequency domain

2.2.2. Down-Sampling based on the standard deviation of the interclass frequency domain

We assume the bands with larger inter-scene standard deviation can effectively distinguish different scenes. Therefore, the corresponding resolution should be higher to be more distinguishable.

Using the formula (3), (4), the data of the training set is statistically calculated, and the overall frequency standard deviation (normalized to 0-1) is obtained, as shown in figure 6.

$$\sigma_{s} = [\frac{1}{M}\sum_{i=1}^{M}\left(\sigma_{sj}^{d} - \overline{\sigma}^{d}\right)]^{\frac{1}{2}} \tag{3}$$

$$\overline{\sigma}^{d} = \frac{1}{M}\sum_{j=1}^{M}\sigma_{sj}^{d} \tag{4}$$

We integrate the frequency standard deviation chart in the above figure, then we get the relationship between the area and the index of the band, the abscissa corresponds to the different frequency band, the vertical coordinate corresponds to the value of the integration between [0,1], as shown in Figure 7.

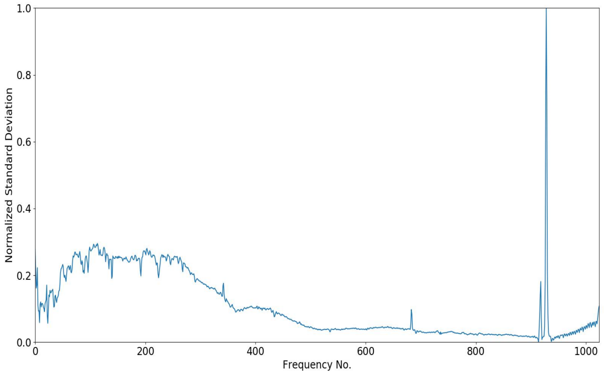

Figure 6: Inter-scene standard deviation in the training set

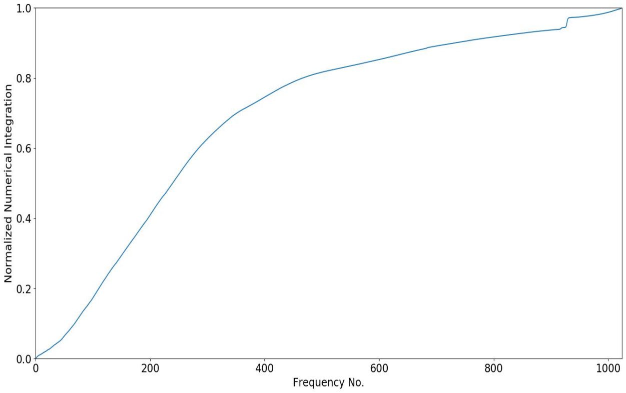

Figure 7: Inter-scene normalized numerical integration of standard deviation

The polynomial fitting method is used to fit the above integral curve as follows:

$$p(x) = p_1 x^n + p_2 x^{n-1} + ... + p_n x + p_{n+1} \tag{5}$$

$$x_i = p^{-1}\left(\frac{i}{D}\right), i = 0,1,2,...,D \tag{6}$$

Then the value of the polynomial is taken as an integer between [0, D] and the equation is solved in reverse, so that the corresponding frequency $[x_i, x_{i+1}]$ can be determined. The D-group boundary $[x_i, x_{i+1}]$ can form a set of ASFSTD filters, which can be used to replace the Mel filter bank to get the ASFSTD-SIF feature. Figure 4 shows the ASFSTD-SIF feature extraction process. The ASFSTD-SIF features are also used as inputs to above mentioned CNN model.

2.3. Late Fusion

In order to fully utilize the aforementioned MFS and ASFSTD-SIF (Fig. 8), we fuse them in frequency domain. Specifically, every audio is represented by five feature maps of 100*100 in terms of MFS or ASFSTD-SIF, and we remain MFS feature map fixed and turn ASFSTD-SIF spectrum upside down. Finally, we merged two feature map into a single fused feature map as shown in Fig. 9.

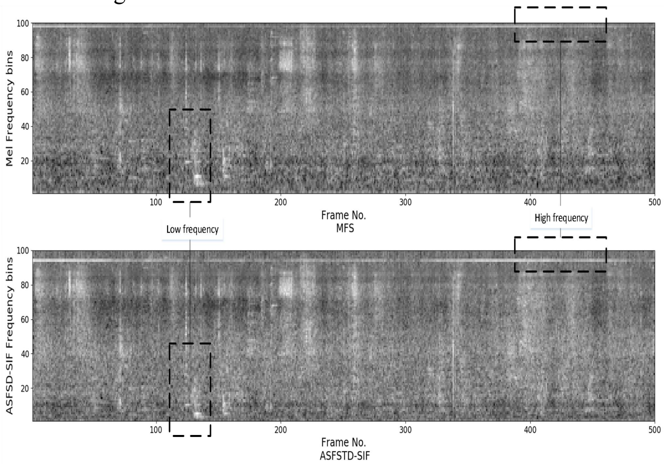

Figure 8: MFS and ASFSTD-SIF spectrogram

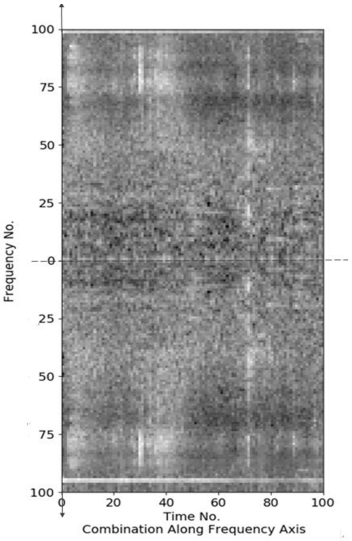

Figure 9: Fused feature map of MFS and ASFSTD-SIF

2.4. Pre-training based on semantic stratification

The classification of sound scenes has a distinct hierarchical relationship[5], we can use two semantic levels of classification labels. The high-level is a rough classification, mainly to distinguish between the environment, such as indoor, outdoor and vehicle, and the low-level is 15 scene labels given by dataset.

Based on two level labels, a supervised convolution neural network pre-training is proposed. the whole training process is divided into two steps: First, CNN1 is trained to predict these three kinds of high-level semantic labels. Second, we transfer CNN1's weight to CNN2 as weight initialization. CNN1 and CNN2 have the same network structure except for the final softmax layer, as illustrated in Fig. 10.

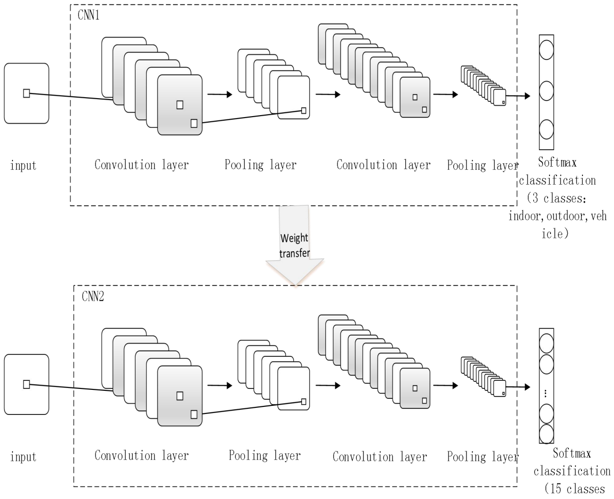

Figure 10: CNN pre-training based on two semantic levels

3. RESULTS

3.1. Submissions

We provided 4 different submissions based on the methods described in the previous sections for the DCASE-2017 challenge,as illustrated in Tab. 1.
Our submissions are:
1. MFS+CNN: Convolutional Neural Network (in Section 2.1)
2. ASFSTD-SIF + CNN (in Section 2.2)
3.Feature fusion + CNN (in Section 2.3)
4.Finetune pre-trained model (in Section 2.4)

Table 1:The results of four methods

Method	4-Fold cross validation accuracy(%)				Average Accuracy (%)
	1	2	3	4	
MFS+CNN	82.3	80.4	79.3	83.1	81.3
ASFSTD-SIF+CNN	82.7	78.6	76.7	83	80.3
Feature fusion+CNN	82.5	81.0	80.4	83.9	82
Transfer learning	82.9	81.9	79.3	85.2	82.3

4. CONCLUSIONS

We found that devised simple ConvNet is effective for ASC task, and the proposed ConvNet architecture outperformed given baseline. We also found that ASFSTD-SIF alone didn't outperform the MFS approach, but feature fusion of two features can improve the accuracy further. Finally, we found fine-tune pre-training can improved the overall and class-wise performance of ASC.

5. REFERENCES

[1] Mesaros, Annamaria, Heittola, Toni, & Virtanen, Tuomas. (2017). TUT Acoustic scenes 2017, Development dataset [DB/OL]. Zenodo. http://doi.org/10.5281/zenodo.400515

[2] Eghbal-Zadeh H, Lehner B, Dorfer M, et al. CP-JKU submissions for DCASE-2016: a hybrid approach using binaural ivectors and deep convolutional neural networks[R/OL].[2016-09-03]. http://www.cs.tut.fi/sgn/arg/dcase2016/documents/challenge_technical_reports/Task1/Eghbal-Zadeh_2016_task1.pdf.

[3] Han Y, Lee K. Acoustic scene classification using convolutional neural network and multiple-width frequency-delta data augmentation[J]. CoRR, 2016, abs/1607.02383.

[4] Valenti M, Diment A, Parascandolo G, et al. DCASE 2016 acoustic scene classification using convolutional neural networks[R/OL]. [2016-09-03]. http://www.cs.tut.fi/sgn/arg/dcase2016/documents/challenge_technical_reports/Task1/Valenti_2016_task1.pdf.

[5] Salamon J, Jacoby C, Bello J P. A Dataset and Taxonomy for Urban Sound Research[C]// ACM International Conference on Multimedia. ACM, 2014:1041-1044.