DCASE 2018 TASK 1A: ACOUSTIC SCENE CLASSIFICATION BY BI-LSTM-CNN-NET MULTICHANNEL FUSION

Wenjie Hao *Lasheng Zhao*[†] *Qaing Zhang* *Hanyu Zhao*[*] *Jiahua Wang*[*]

Dalian University.
Key Laboratory of Advanced Design and Intelligent Computing Ministry of Education.
116622,Jinzhou District, Dalian, Liaoning. 116000, China.

xiaoniuchushi@163.com goodzls@126.com zhangq@dlut.edu.cn

ABSTRACT

In this study, we provide a solution for acoustic scene classification task in the DCASE 2018 challenge. A system consisting of bidirectional long-term memory and convolutional neural networks(BI-LSTM-CNN) is proposed. And, improved logarithmic scaled mel spectra as input to our system. Besides，we have adopted a new model fusion mechanism. Finally, to validate the performance of the model and compare it to the baseline system, we used the TUT Acoustic Scene 2018 dataset for training and cross-validation, resulting in an 13.93% improvement over the baseline system.

Index Terms— Acoustic scene classification, log-scaled mel-spectrograms, convolutional neural networks, bidirectional Long Short Term Memory

1. INTRODUCTION

Acoustic Scene Classification(ASC) refers to inferring the corresponding label from the given audio data [1]. This audio information includes all the information of this scene, such as ambient noise, vocals, and so on. We need to determine from the order of the information, which scene it belongs to. Sound field classification has a large application market. For example, activate a wearable device in a specific scenario [2]. In other specific scenarios, the device is turned off. This operation saves power on the wearable device and extends the life of the wearable device.

For extracting audio features, We can divide into two major categories where (1) Using traditional signal analysis and processing methods [3-5]. (2) Using machine learning methods for research and processing [6-7]. For the first method, before SoundNet [8] proposed, it required researchers to have deep domain knowledge in the field of sound. However, compared to the first method, the second method has many applications. Especially after the Detection and Classification of Acoustic Scenes and Events(DCASE) challenge was launched, a large number of public data sets were being studied by more and more people. On the DCASE 2013 challenge, the competitors use the Support Vector Machine(SVM) classifier to classify features such as Mel-Frequency Cepstral Coefficients(MFCC), and finally get a good result [9]. At DCASE 2016 challenge, entrants Eghbal-Zadeh [10] used Deep Convolutional Neural Network (DCNN) architecture trained on spectrograms of audio excerpts in end-to-end fashion. Among them, their model system input data is characteristic of MFCC. Participant Sangwook Park [11] proposed a system based on Gaussian mixture model and neural network fusion. In the DCASE 2017 challenge, the contestant Yoonchang Han [12] proposed a variety of preprocessing methods that emphasise different acoustic characteristics. Participants Seongkyu Mun [13] proposes to use Generative Adversarial Networks(GAN) based method for generating additional training database(DB). As can be seen from the above paper, the researchers just started to pay attention to the features, then began to pay attention to the model structure, and finally began to pay attention to the features in the case of insufficient data sets.

The structure of this article is as follows. Section 2 describes the network architecture for feature preprocessing and feature extraction classification. Section 3 describes the source of experimental data, the advance processing of experimental data and the experimental process.

2. SYSTEM DESCRIPTION

2.1. Feature Extraction

After looking at the training set, we found that the audio frequency range is distributed from 0 to 22K (as shown in Figure 1). In the picture, the horizontal axis represents the frequency and the vertical axis represents the decibel value. The blue line represents the power spectrum for each band. In the drawing of this figure, the frame of 2048 is used, and 50% of the frame is used as the noverlap. From the figure, we can see that there is obviously energy in the frequency of 20K.

In order to ensure that more feature information is analyzed, we used the 216 bands log-scaled mel-spectrogram as the input representation to the BI-LSTM-CNN. We used short-time Fourier transform (STFT) is applied using hamming windows of 2048 frame size with 50% overlap. And, the STFT spectrogram is then used to calculate the 216 bands between energetics with an 22 kHz high frequency bound. Finally, the Mel energies are

[†] Corresponding author.

[*] Thanks for the extra support in mathematics.

converted to the log scale by using the formula *log(100000 • Mel + 1)*. Through the above operation, the one-dimensional audio signal is converted into a two-dimensional(2D) image signal, as shown in Figure 2.

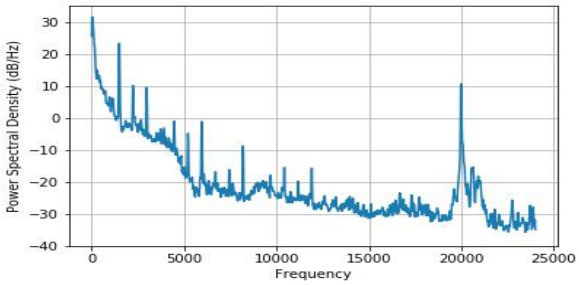

Figure 1: Spectrum in development dataset for DCASE 2018 challenge; in library a part of " metro-helsinki-45-1363-a.wav"

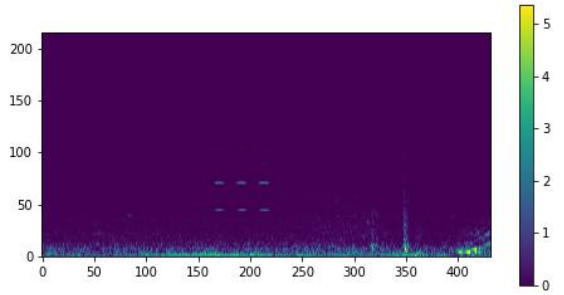

Figure 2: Original mel-spectrograms of " metro-helsinki-45-1363-a.wav"

In Figure 2 and Figure 3, the abscissa represents time and the ordinate represents frequency band. The color on the right indicates the correspondence between color and number. We can see that the high frequency signal did not appear in the image. As shown in Figure 2, a significant energy band appears at the normal 20K position, but now it is completely invisible. By analyzing the matrix data, we know that the energy value of the high frequency band is relatively small compared to the energy value of the low frequency band, so there is no discrimination.

In order to solve this problem, we have made improvements on the basis of the original image, and the specific improvements are as follows:

1) We need to normalize the 2D image features to make their values range from 0 to 1 (Equation 1, 2, 3).

$$X_2 = \max(abs(X_1)) \tag{1}$$

$$X_3 = \min(X_1) \tag{2}$$

$$X_4 = (X_1 - X_3 + \xi)/X_2 \tag{3}$$

In Equation 1, 2 and 3, *abs* represents the absolute value of each element in the matrix, and *max* and *min* represent the maximum and minimum values in the matrix, respectively. X_1 is a matrix representing the original 2D feature data. And, ξ represents a small positive number, this is to prevent the data from being equal to 0, causing the error to be set.

2) Map the features and increase the distance between the feature data.

$$X_5 = \log(X_4) \tag{4}$$

3) A unified and simple treatment of new features prevents some features from appearing different from similar features(Equation 4).

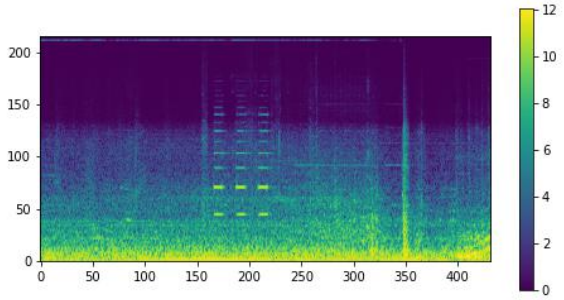

Figure 3: Processed mel-spectrograms of "metro-helsinki-45-1363-a.wav"

Figure 2 shows the characteristics through the above transformation. The results are shown in Figure 3. We can see that the information in Figure 3 is significantly more abundant than in Figure 2. Especially in the high frequency area, the picture information is more refined.

2.2. Network Architecture

CNNs appear to be a reasonable choice for this task for various reasons. However, our current image feature information is different from ordinary image information, and our feature information has a strict context. Therefore, we added a bidirectional long-term memory(Bi-LSTM) on the first level of input. The complete network structure diagram is shown in Table 1. The overall network is implemented with *Keras library* [14].

Table 1: BI-LSTM-CNN architecture; BN: Batch Normali-zation; ELU: Exponential Linear Unit.

5×5 Bidirectional(ConvLSTM2D) -64-BN-ELU
3×3 Conv2D-64-BN-ELU
3×2 Max-pooling2D
3×3 Conv2D-128-BN-ELU
3×3 Conv2D-128-BN-ELU
2×2 Max-pooling2D
3×3 Conv2D-512-BN-ELU
3×3 Conv2D-512-BN-ELU
3×3 Conv2D-128-BN-ELU
3×3 Conv2D-10-BN-ELU
global_average_pooling2d_1
Dense-10-BN-softmax

In Table 1, all convolutional layers are initialized with He uniform. And all the cores of the convolutional layer are regularized by *l1* and *l2*, where the parameter used by *l1* is *3e-4*, and the parameter used by *l2* is *1e-4*. At the output of the network,

we use the loss function as cross entropy and optimize the network with the optimizer *ADAM*. In addition, we dynamically adjust the learning rate with a monitorable learning rate.

2.3. Late Fusion

Although in this study, we only use one network structure, we will get different training samples, and the training samples are very different. At this time, there will be a model parameter, which is designed to be model fusion. The traditional approach is to use a voting mechanism: which tag has more tags and which one to choose. Or to use arithmetic mean as the late fusion method, which combines prediction probabilities from model1 and model2 systems by taking the arithmetic mean of the probabilities for each recording [15] as Equation 5.

$$pred^{i} = \arg\max\left(\frac{proba_1^{i} + proba_2^{i}}{2}\right) \quad (5)$$

where $proba_1^{i}$ and $proba_2^{i}$ are respectively the prediction probabilities for the recording i from model1 and model2; $pred^{i}$ is the predicted label. This strategy led to better results than geometric mean.

However, all of these methods have some drawbacks, which in some cases do not improve the final result. To this end, we propose a new method, which itself is also a scoring mechanism. The difference is that we have added weights for scoring on the basis of scoring. As we all know, when evaluating multi-classification systems, precision, recall, and F1-Measure(FM) often perform the performance evaluation of the model. Therefore, when the model is merged, we also introduce the performance indicators of the model itself as a reference indicator of its importance. For the sake of convenience, we used the FM most weighted reference value in this paper. For example, there are now three models. The prediction result of the first model is *A*, the value of FM in the model is *AF1*, and the result of the second model is *B*, which does not correspond to the value of *FM* in the model for *FM1*; the third model with this result is *FM2*, which does not correspond to the value of FM in the model is *FM3*. First, for models with the same prediction results, their FM correspondences are added. In this example, FM_2 and FM_3 need to be added to get *FM23*. Then compare FM_{23} with FM_1 and choose a larger *FM* (as in Equation 6), then this *FM*.

$$FM^{i} = \arg\max(FM_1^{i}, FM_{23}^{i}) \quad (6)$$

In Equation 6, the *i* in the upper right corner is the prediction of the i-th data.

3. EXPERIMENTS

3.1. Dataset

The dataset for this task is the TUT Urban Acoustic Scenes 2018 dataset, consisting of recordings from various acoustic scenes. The dataset was recorded in six large european cities, in different locations for each scene class. For each recording location there are 5-6 minutes of audio. The original recordings were split into segments with a length of 10 seconds that are provided in individual files. Available information about the recordings include the following: acoustic scene class, city, and recording location. There are 10 acoustic scenes, which are Airport, Indoor shopping mall, Metro station, Pedestrian street, Public square, Street with medium level of traffic, Travelling by a tram, Travelling by a bus, Travelling by an underground metro, Urban park.

3.2. Multichannel Audio Preprocessing

Since the audio sample data of the game is relatively small, in order to extract more common features from the sample, we extract five monoaural versions from each audio file as Table 2.

Table 2: Five Monoaural Versions

Monaural versions	Method
First version	Left channel
Second version	Right channel
Third version	(Left channel + Right channel)/2
Fourth version	(Left channel - Right channel)/2
Fifth version	(Right channel + Left channel)/2

3.3. Evaluation Setup

First, we processed all development datasets and test datasets into 2D mel spectra, all processed by the new method proposed in this paper. The development dataset is then divided using the officially recommended classification. Then, under the train dataset, 10% is randomly assigned as the verification set. Make sure that all files having same location id are placed on the same side of the evaluation. Next, the audio information is extracted from the perspective of the channel, and the final classification is divided into five monoaural versions. Finally, use the network structure given in Section 2.2 to train the model separately.

Through the above operations, five models are finally obtained. We need to weight the model predictions based on the final precision of each model.

4. RESULTS AND DISCUSSION

In order to compare with the baseline system, we used the same validation set as the baseline system. After several experiments and other operations, we selected the best one from each of the five types of data to compare with each other, such as Table 3. From the table we can see that although the five versions of the model are roughly the same, there are still many gaps in each category. Therefore, it is necessary to merge.

In order to compare with the baseline system, we used the same validation set as the baseline system. After several experiments and other operations, we selected the best one from each of the five types of data to compare with each other, such as Table 3. From the table we can see that although the five versions of the model are roughly the same, there are still many gaps in each category. Therefore, it is necessary to merge.

Then we use the model fusion mechanism proposed in this paper to fuse the best five models, and draw the final

experimental results, and then compare with the baseline system, as shown in Table 4.

Table 3: Classification accuracy of five models for each category

Classes	First	Second	Third	Fourth	Fifth
Airport	**0.58**	0.45	0.56	0.58	0.65
Bus	**0.76**	0.61	0.69	0.73	0.74
Metro	0.55	**0.71**	0.67	0.66	0.62
Metro Station	**0.75**	0.73	0.71	0.71	0.56
Part	0.84	**0.91**	0.87	0.85	0.84
Public Square	0.62	0.41	0.44	0.36	**0.66**
Shopping Mall	0.85	**0.91**	0.89	0.86	0.77
Street Pedestrian	0.52	0.49	**0.59**	0.55	0.39
Street Traffic	**0.90**	0.90	0.86	0.87	0.86
Tram	0.70	0.63	**0.77**	0.68	0.65
Average	0.71	0.68	0.71	0.69	0.67

Tabel 4: The final model of this paper is compared with the limit system(%)

Classes	Baseline system	Bi-LSTM-CNN
Airport	72.9	55
Bus	62.9	75
Metro	51.2	72
Metro Station	55.4	76
Part	79.1	90
Public Square	40.4	49
Shopping Mall	49.6	92
Street Pedestrian	50.1	58
Street Traffic	80.5	89
Tram	55.1	76
Average	59.7(±0.7)	**73.63(±0.9)**

5. CONCLUSION

This paper provides a new algorithm for sound field recognition. This algorithm proposes some new ideas in terms of feature extraction, system architecture, and model fusion. And the task 1A of DCASE 2018 has achieved very good results. In the public kaggel rankings [16], 71% of the results were achieved.

6. REFERENCES

[1] D. Barchiesi, D. Giannoulis, S. Dan, M.D. Plumbley, “Acoustic Scene Classification: Classifying environments from the sounds they produce,” *IEEE Signal Processing Magazine*. 32(3): pp. 16-34, 2015.

[2] Y. Xu, W.J. Li, Lee, “Intelligent Wearable Interfaces,” 2007.

[3] R. Chassaing, D. Reay, "Finite Impulse Response Filters." Wiley-IEEE Press, 2001:146-209.

[4] D.J. Hedley, J. Richards, “Infinite impulse response filters,” US. pp. 210 - 254, 1989.

[5] P. Fauchais, A. Vardelle, “Wiley Encyclopedia of Electrical and Electronics Engineering,” *Library Journal*. 3(15): pp. 71, 1999.

[6] Williams, W. J., & Neill, J. C. "Decomposition of time-frequency distributions using scaled-window spectrograms," *Advanced Signal Processing Algorithms*,vol. 2563, pp. 44-59, June. 1995.

[7] S.B. Davis, P. Mermelstein, “Evaluation of acoustic parameters for monosyllabic word identification,” *Journal of the Acoustical Society of America*, 64(S1): pp. S180-S181, 1978.

[8] Y. Aytar, C. Vondrick, A. Torralba, “SoundNet: Learning Sound Representations from Unlabeled Video,”2016.

[9] G. Roma, W. Nogueira, and P. Herrera, “Recurrence quantification analysis features for auditory scene classification,”*IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, 2013.

[10] Y. Han, J. Park, K. Lee, “CP-JKU Submissions for DCASE-2016: a Hybrid Approach Using Binaural I-Vectors and Deep Convolutional Neural Networks,”2016.

[11] S. Park, S. Mun, Y. Lee, and H. Ko, “Score fusion of classification systems for acoustic scene classification,” *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2016.

[12] Y. Han, J. Park, K. Lee, “Convolutional Neural Networks with Binaural Represntations and Background Subtraction for Acoustic Scene Classification,” in *Detection and Classification of Acoustic Scenes and Events*. 2017.

[13] S. Zhai, Y. Cheng, R. Feris, et al., “Generative Adversarial Networks as Variational Training of Energy Based Models,” 2016.

[14] https://github.com/fchollet/keras.

[15] E. Fonseca, R. Gong, D.Bogdanov, O. Slizovskaia, E. Gomez, and X. Serra, “Acoustic scene classification by ensembling gradient boosting machine and convolutional neural networks,” Workshop on Detection and Classification of Acoustic Scenes and Events, 2017, pp. 37-41.

[16] https://www.kaggle.com/c/dcase2018-task1a-leaderboard