DEVELOPMENT OF THE INRS-EMT SCENE CLASSIFICATION SYSTEMS FOR THE 2020 EDITION OF THE DCASE CHALLENGE (TASKS 1A AND 1B)

Technical Report

Amr Gaballah [1*], *Anderson Avila* [1*], *Joao Monteiro* [1*], *Parth Tiwari* [1,2*], *Shruti Kshirsagar* [1*], *Tiago H. Falk* [1]

[1]Institut National de la Recherche Scientifique - Centre EMT, Montreal - Canada
[2] Dept. of Industrial and Systems Engineering, IIT Kharagpur, India

ABSTRACT

In this report, we provide a brief overview of a set of submissions for the scene classification sub-tasks of the 2020 edition of the DCASE challenge. Our submissions comprise efforts at the feature representation level, where we explored the use of modulation spectra and log-mel filter banks, as well as modeling strategies, where recent convolutional deep neural network models were used. Results on the Challenge validation set show several of the submitted methods outperforming the baseline model.

***Index Terms*—** Scene classification, i-vectors, Modulation spectra, Convolutional models

1. SUMMARY OF CONTRIBUTIONS

1.1. Task 1A

We submit systems consisting of convolutional models trained on top of spectral representations of audio, namely:

1. System S1: The first system builds on a standard ResNet-18 [1] and removes parts of its layers, which we empirically found to improve performance on the validation partition. We refer to this model as ResNet-12. Kaldi style log-mel filter banks are then used in the inputs and treated as single channel images, i.e. spatial-temporal convolutions are employed. Pre-processing steps besides feature extraction consist of data augmentation, which are performed in two steps: 1) prior to feature computation using sox distortions in gain and tempo; 2) directly on the spectra by randomly dropping out continuous chunks along both the time and frequency dimension, as well as addition of Gaussian noise. All augmentations are performed in an online fashion, and every time a given recording is sampled, we randomly decide whether it will be distorted or not such that half of the examples are presented to the model after some sort of augmentation was performed on average.
2. System S2: Our second submission makes use of time delay neural networks (TDNN) [2]. Such models are often used within the context of speech recognition for computation of frame-level representations. Utterance-level variations of TDNNs were shown in recent literature to be effective in computing speaker- or language-dependent representations if some sort of temporal pooling is further used. We thus leverage that architecture for the task considered herein and train an x-vector TDNN [3] with statistical temporal pooling on top of the same representations discussed for system 1, employing exactly the same augmentation strategy described above. The TDNN we employed is made up of 5 temporal dilated convolutional layers followed by temporal pooling and 2 dense layers. We further remark that, in the case of both system 1 and 2, we initialize models from pre-trained versions on the data released for task 1B, which we observed improved validation performance in some classes.
3. System S3: Our third submission is once more based on a ResNet architecture. In this case, we employed a ResNet-18 as is, but on top of modulation spectra computed from the log-mel filter banks described before. The modulation spectra are obtained by computing the STFT over each frequency bin of the mel-spectra, computed in advance. We average the results across time and end up with a representation with two dimensions: acoustic vs. modulation frequency. The same types of augmentations were used in this case as well. No pre-training step was performed in this case and the ResNet-18 was trained from scratch.
4. System S4: Our fourth submission corresponds to a score-level fusion of five systems. We thus considered the three systems discussed above, and added a simple 2-layered convolutional model and further included a ResNet-12 trained from scratch, and in both cases the log-Mel spectra were used as inputs to the models. Fusion is performed in a simple averaging scheme: given a test example, we project it in the probability simplex by forwarding it into each of the five considered models, and average the final results. Our final prediction is thus given by the most likely class according to the combined set of scores.

1.2. Task 1B

For this task, we employ a small ReLU activated 2-layered convolutional model, trained on top of log-mel filter banks extracted in Kaldi-style. Each convolutional layer is followed by batch normalization. Features are computed such that 40 log-mel filter banks are extracted using the Kaldi compliant API of torchaudio[1]. Data augmentations are performed in order to increase the diversity of train data, which we do by randomly deciding when to augment, and further randomly deciding which kinds of distortions will be

*Equal contribution. Authors listed in alphabetical order.

[1]https://pytorch.org/audio/compliance.kaldi.html

Table 1: Results in terms of accuracy for each class for task 1A.

Class	System			
	S1	S2	S3	Baseline
Airport	**60.6%**	44.1%	49.2%	45.0%
Public square	39.7%	**48.8%**	42.8%	44.9%
Bus	**84.5%**	71.0%	69.7%	62.9%
Metro	**59.3%**	58.2%	49.2%	53.5%
Metro station	**63.6%**	48.5%	55.2%	53.0%
Park	**78.5%**	70.4%	69.7%	71.3%
Shopping mall	**62.3%**	50.8%	53.9%	48.3%
Street pedestrian	**43.4%**	37.0%	28.3%	29.8%
Street traffic	**85.5%**	**85.5%**	80.8%	79.9%
Tram	**68.7%**	62.3%	61.6%	52.2%
Average	**64.6%**	57.7%	56.0%	54.1%

Table 2: Results in terms of accuracy for each class for task 1B.

Class	Proposed System	Baseline
Indoor	**85.7%**	82.0%
Outdoor	83.2%	**88.5%**
Transportation	**93.8%**	91.5%
Average	**87.6%**	87.3%

performed. In summary, for each recording at training time, we first decide whether it will be augmented with equal chances of doing so or not, and in case we decide to augment it, we flip fair coins to decide when to apply each of the following set of distortions: Sox distortion on tempo, Sox distortion on gain, SpecAug on frequency, SpecAug on time, and Gaussian noise addition.

2. VALIDATION SET RESULTS

In this Section, we present results of the submitted systems for both tasks 1A and 1B.

2.1. Task 1A

Results are presented in Table 1.

2.2. Task 1B

Results are presented in Table 2.

3. FEATURES USED

3.1. Modulation spectrum representation

The signal processing steps involved in the computation of the modulation-spectral representation are depicted in Fig 1. For Task 1B, the modulation toolbox from the University of Washington was used [4]. First, the audio signal $x(n)$ (here, sampled at 16 Hz) is segmented into consecutive overlapping frames using a 512-point hamming window with 75 % overlap, which are then transformed to the frequency domain using a 512-point fast Fourier transform (FFT), thus resulting into a conventional spectrogram. Spectral magnitude components $X(f, m)$ are then segmented over the time axis into consecutive overlapping frames using 128-point "modulation" windows with 75% overlap, which are further processed by a 512-point FFT into the final frequency–frequency representation

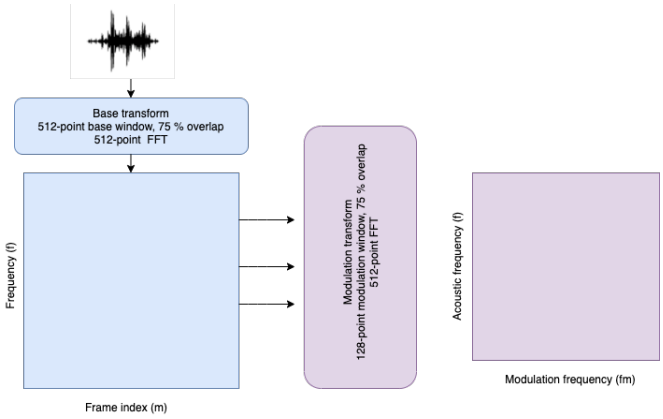

Figure 1: Modulation spectrogram representation

$X(f, fm)$. The representation $X(f, fm)$ is called the modulation spectrogram, where f corresponds to acoustic frequency and fm to modulation frequency [5].

3.2. Mel-band spectra

These correspond to Kaldi-style mel-band spectra with similar parameters as those described in the Challenge website.

4. REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[2] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE transactions on acoustics, speech, and signal processing*, vol. 37, no. 3, pp. 328–339, 1989.

[3] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[4] L. Atlas, P. Clark, and S. Schimmel, "Modulation toolbox version 2.1 for matlab," *University of Washington*, 2010.

[5] T. H. Falk, M. Maier, *et al.*, "Ms-qi: A modulation spectrum-based ecg quality index for telehealth applications," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 8, pp. 1613–1622, 2014.