EXPLORING DEEP VISION MODELS FOR ACOUSTIC SCENE CLASSIFICATION

Technical Report

Octave Mariotti, Matthieu Cord, Olivier Schwander

Sorbonne Université, CNRS
Laboratoire d'Informatique de Paris 6, LIP6
F-75005 Paris, France
name.surname@lip6.fr

ABSTRACT

This report evaluates the application of deep vision models, namely VGG and Resnet, to general audio recognition. In the context of the IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events 2018, we trained several of these architecture on the task 1 dataset to perform acoustic scene classification. Then, in order to produce more robust predictions, we explored two ensemble methods to aggregate the different model outputs. Our results show a final accuracy of 79% on the development dataset, outperforming the baseline by almost 20%.

Index Terms— Acoustic scene classification, DCASE 2018, Vision, VGG, Residual networks, Ensemble

1. INTRODUCTION

Despite remarkable improvement over the last decade in computer vision, its neighbor field audio signal processing seems to be lagging behind, a delay often attributed to the lack of available labeled data to train deep networks. Nonetheless, recent results show that if sufficient data is provided, vision models seem to perform quite well in audio classification tasks[1], implying that a transition towards deeper models will soon occur.

Current state-of-the-art algorithm for acoustic scene classification rely on preprocessing audio signal to obtain a spectral representation of the signal, and then training a convolutional neural network to classify the data. There does not seem to be a general consensus on which features are best, although most commonly found are mel-spectrograms[2] and Mel Frequency Cepstral Coefficients (MFCC)[3]. Other popular features include standard or Constant-Q Transforms spectrograms[4] or more complex constructions such as i-vectors [5]. It should be noted that apart from regular and CQT spectrograms, all other preprocessing are specifically designed for music or speech processing, thus may introduce bias in a general audio dataset.

Classification of these features is almost always performed using CNNs whose architecture are loosely based on VGG networks [2][4]. Variation exists, often including recurrent layer to capture the temporal structure of audio signal[6], and Support Vector Machines are sometimes used to better classify high-level extracted features[4]. Models rarely exceed ten layers.

In comparison, current vision models are often deeper because image datasets contain much more examples. Nonetheless, some network architectures, although deep, are remarkably strong at avoiding overfitting, using regularization layers such as dropout or batch normalization. We intend to explore how these network behave on tasks they were not directly designed for.

Ensemble methods are often used to obtain better predictions either via direct methods like voting[6] or indirect, such as fitting an auxiliary classifier[4].

Our contribution relies on testing models of different depths and different structures in order to determine up to how many layers can a model grow before overfitting, and ensembling several of these models to benefit from different levels of abstraction.

2. PREPROCESSING

The preprocessing of audio sample is based on the COCAI DCASE 2017 submission[2], using several kind of log-mel spectrograms. The mel-spectrograms were computed from audio file converted from 24 to 16-bit encoding, with original 48kHz sample rate. The window was 2,048 samples long with a hop length of 1,024, for 128 mel bins. After converting the spectrograms to logarithmic scale, a final step of normalization was performed, consisting in subtracting μ and dividing each spectrogram by σ, where μ and σ are the average mean and standard deviation of spectrograms obtained from the development dataset.

Four types of spectrograms were created and used to train networks: Mono, Stereo, Mid / Side and Harmonic / Percussive. The mono preprocessing produces one mel-

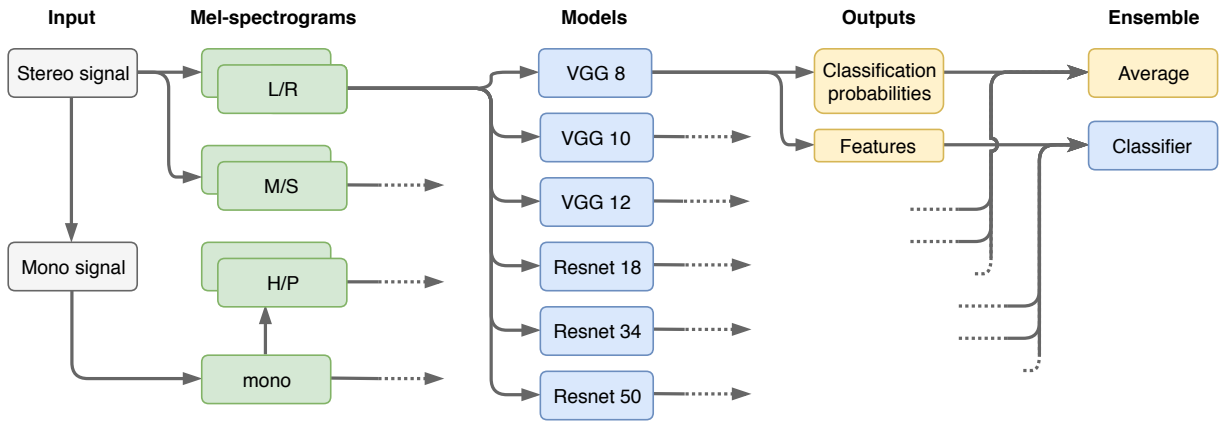

Figure 1: Architecture of the whole system

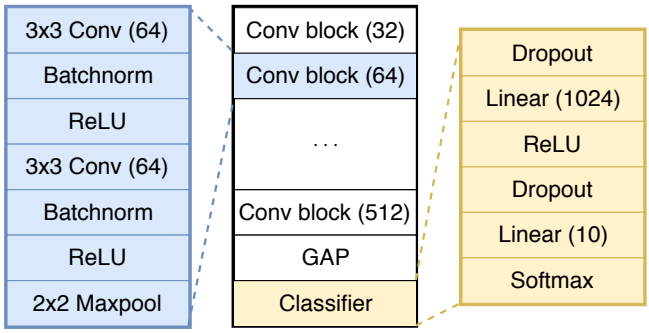

Figure 2: Architecture of VGG-inspired network

spectrogram per audio file, while the remaining three types produce a pair.

A stereo pair consists in a spectrogram extracted from the left (L) and right (R) channel respectively, and a mid/side pair from $L + R$ and $L - R$ channels.

Harmonic / Percussive preprocessing also produces two spectrograms, one containing harmonic features (or stationary frequencies) the other containing percussive features (or transient frequencies). Harmonic Percussive Source Separation (HPSS) first originated in music processing to isolate other instruments from the drum sounds that often made signals harder to process[7]. The algorithm used to perform HPSS was the median-based algorithm[8] from librosa audio processing library with default parameters.

3. NETWORK ARCHITECTURES

We explored two kinds of network architectures inspired by well-tested vision models : VGG and residual networks.

3.1. VGG

VGG networks[9] have a straightforward architecture, consisting in a sequence of blocks containing two or three convolution layers followed by max-pooling. Because of the pooling layers, the spatial extent of the signal is reduced after each block. Conversely, the number of channels increases, allowing for more abstraction. We chose to limit the maximum number of channels to 512. After several of these blocks, two fully-connected layers are used to classify features.

Our architecture uses the same principle with two distinct features : regularization layer and global average pooling (figure 2).

regularization layers Between each convolution layer and its activation function is inserted a batch normalization layer[10]. This introduces very few parameters and greatly improves generalization, yielding an improvement of 10% in accuracy for the VGG8 model. Additionally, the classifier also incorporated dropout layer before fully-connected filter[11].

Global average pooling Instead of flattening the features maps obtained at the end of the convolution sequence and plugging the MLP on the resulting features, we chose to use global average pooling, averaging each feature map both on the time and frequency domain. This reduces the number of parameters with no apparent loss on accuracy. Moreover, GAP layers have been extensively used in object localization tasks[12], making this architecture interesting for audio segmentation.

We used three models totalizing 8, 10 and 12 layers respectively, including the fully-connected layers. Larger sizes were shown to overfit. These models are much shallower than actual VGG networks used in computer vision (at least

16 layers), but the drastically reduced number of parameters (5.3 versus 138 millions) allows them to cope with the small amount of examples in the dataset.

3.2. Residual networks

Residual networks[13] have introduced the concept of skip-connection, that have been reused in many architectures, including audio processing tasks[14]. It consists in adding the input to the output of a convolution block, with the initial motivation of not harming performances when adding layers. These skip connections coupled with batch normalization make them remarkably robust to overfitting.

We have tested several standard sizes of residual networks in order to assess their behavior on a relatively small dataset, selecting three in the end: 18, 34 and 50. These are canonical architectures described in the original paper, the only addition we made was a dropout layer between the GAP layer and the fully-connected classifier. Note that the 50-layered version uses bottlenecks blocks while the other two use standard ones.

3.3. Training

Our networks were trained using the data segmentation provided with the data, on each of the four dataset of mel-spectrograms. When processing a pair of spectrograms (L/R, M/S and H/P), we set the input channels of our networks to two. Training was performed using gradient-descent with momentum, with a batch-size of 64, an initial learning rate of 10^{-2} divided by 2 if no improvement was observed on validation loss for 10 epochs. To prevent overfitting, we stopped the training if the average loss did not improve over a sliding window spanning the last 30 epochs.

After training, each network produced a set of predictions and a set of features for each dataset. The features are defined as the output of the GAP layer for both architectures. These outputs were stored in order to perform ensemble methods (figure 1).

4. ENSEMBLING

Ensemble methods are a common technique used to improve generalization, involving making a classification choice based on the output of several models. Common ensemble methods include voting, taking the maximum average classification probability over models or using special algorithms designed to optimize the weight of each model.

We explored two methods : mean of probabilities and training a multi-layer perceptron on extracted features.

Mean probabilities Averaging prediction probabilities is a straightforward method that provides results a bit more robust compared to voting, as it is more stable on difficult examples because outputs are not binary.

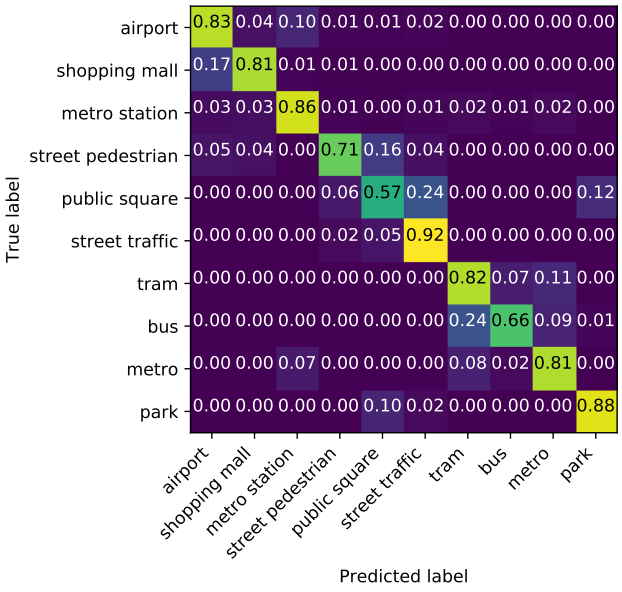

Figure 3: Confusion matrix of the NN no50 system

MLP We hoped that a multi-layer perceptron trained on the extracted features can detect complex patterns across models and perform better than simple ensembling of outputs. However, it is also very prone to overfitting, as extracted feature are already very expressive. Consequently, we tried to train it with different subsets of extracted features.

In the end, each set of spectrograms was used to train 6 models, so the ensemble was performed on up to 24 models in total.

Another possible system could be trained end-to-end with each network outputting features an a global classifier trained on the aggregation of the networks outputs, allowing each to specialize on different features. However, this requires holding many parameters in memory thus severely limits the size of networks used.

5. EXPERIMENTAL RESULTS

The experimental results in table 1 tend to show that VGG-like models perform better, most likely because residual networks tend to overfit the data. Nonetheless, they exhibit interesting performances, with the 50-layer version showing little accuracy drop (2-3%) compared to the 18-layer one despite having more than twice as many layers.

Mid / Side preprocessing consistently outperforms its

Model	Mono	L/R	M/S	H/P
VGG8	67.9	68.4	73.8	68.5
VGG10	72.8	70.9	77.7	72.8
VGG12	74.6	72.5	77.1	76.2
Resnet18	71.0	70.8	73.7	72.2
Resnet34	69.1	65.6	72.3	68.1
Resnet50	69.2	68.9	71.0	69.4

Table 1: Accuracy of each network

Ensemble	Decision method	Resnet50	Accuracy
MP all	Mean probabilities	included	78.4
MP no50	Mean probabilities	excluded	79.1
NN all	Neural network	included	76.4
NN no50	Neural network	excluded	79.3

Table 2: Accuracy of ensembles

peers, most likely because the side channel isolates interesting spatial information of the signal.

Both ensemble techniques were tried on two subset of models : the whole set, or all VGG models and Resnets 18 and 34. This is because Resnet 50 results show severe signs of overfitting, thus might hinder predictions.

The mean probabilities ensemble allowed to reach an accuracy of 79.1, a small improvement the best model (VGG10 M/S). Although the gain was only marginal, we expect that generalization was improved.

The trained MLP achieved a final accuracy of 79.3 when trained on all features but those obtained from Resnet50. We tried isolating some training example to use them only for the training of the aggregating classifier, but this yielded poor results as either extracted features were not good enough, or the classifier overfitted the data. Although VGG8 display similar accuracies, it is most likely caused by underfit, thus less problematic when ensembling.

The confusion matrix (figure 3) shows problematic confusion between several classes: airport an shopping mall, tram and bus, and most surprisingly, public square and street traffic.

5.1. Other experiments

Other architectures were also considered but dropped because they performed poorly:

Attention Both VGG-like and Resnet architectures were tested with the addition of gated activation function, often referred to as attention mechanism. Such designs were first introduced by vision models[15] and popularized on audio processing models by Wavenet[14]. However, experiments did not show a statistically significant improvement from this design compared to simply fitting a ReLU activation function.

Raw audio Models were also adapted to work on raw audio, effectively by swapping all 2D convolution filters with 1D equivalents. First results were promising, but these models were difficult train, because of the time and memory needed to run them, and hyperparameter optimization was fairly complex. We have also tested two existing architectures working with raw audio - Envnet[16] and Wavenet[14] - but their results were only slightly above the baseline. Moreover, because these models work with lower-level data, they need more layers compared to spectral-based models, resulting in higher overfit probability.

Cepstral features Models were also trained using cepstral features, namely MFCC obtained from the mel spectrograms, but this yielded a significant drop in accuracy, so they were not considered for the ensemble methods.

6. CONCLUSION

Our system using an ensemble of classical vision models to classify acoustic scenes yielded an improvement of nearly 20 points in accuracy compared to the baseline of the challenge. Although not really original, our approach allowed to highlight the importance of regularization layers, in particular batch normalization, and validated the use of global average pooling layers to reduce number of parameters.

Despite skip connections not being able to completely negate overfitting, residual networks proved to perform well with respect to their depth. As such we believe they could become the next state-of-the-art model in audio recognition once larger datasets are available.

7. REFERENCES

[1] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, "Cnn architectures for large-scale audio classification," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 131–135.

[2] Y. Han and J. Park, "Convolutional neural networks with binaural representations and background subtraction for acoustic scene classification," DCASE2017 Challenge, Tech. Rep., September 2017.

[3] R. Hyder, S. Ghaffarzadegan, Z. Feng, and T. Hasan, "BUET bosch consortium (B2C) acoustic scene classi-

fication systems for DCASE 2017," DCASE2017 Challenge, Tech. Rep., September 2017.

[4] Z. Weiping, Y. Jiantao, X. Xiaotao, L. Xiangtao, and P. Shaohu, "Acoustic scene classification using deep convolutional neural network and multiple spectrograms fusion," DCASE2017 Challenge, Tech. Rep., September 2017.

[5] B. Lehner, H. Eghbal-Zadeh, M. Dorfer, F. Korzeniowski, K. Koutini, and G. Widmer, "Classifying short acoustic scenes with I-vectors and CNNs: Challenges and optimisations for the 2017 DCASE ASC task," DCASE2017 Challenge, Tech. Rep., September 2017.

[6] S. Mun, S. Park, D. Han, and H. Ko, "Generative adversarial network based acoustic scene training set augmentation and selection using SVM hyper-plane," DCASE2017 Challenge, Tech. Rep., September 2017.

[7] N. Ono, K. Miyamoto, J. Le Roux, H. Kameoka, and S. Sagayama, "Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram," in *Signal Processing Conference, 2008 16th European*. IEEE, 2008, pp. 1–4.

[8] D. Fitzgerald, "Harmonic/percussive separation using median filtering," in *13th International Conference on Digital Audio Effects (DAFx-10)*, 2010.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[12] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2921–2929.

[13] 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.

[14] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio." in *SSW*, 2016, p. 125.

[15] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, *et al.*, "Conditional image generation with pixelcnn decoders," in *Advances in Neural Information Processing Systems*, 2016, pp. 4790–4798.

[16] Y. Tokozume and T. Harada, "Learning environmental sounds with end-to-end convolutional neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2721–2725.