ACOUSTIC SCENE CLASSIFICATION: AN EVALUATION OF AN EXTREMELY COMPACT FEATURE REPRESENTATION

*Gustavo Sena Mafra**

Universidade Federal de Santa Catarina
Florianópolis, Santa Catarina, Brazil
gsenamafra@gmail.com

Ngoc Q. K. Duong†*, Alexey Ozerov, Patrick Pérez*

Technicolor
975 Avenue des Champs Blancs CS 17616
35576 Cesson-Sévigné, France
firstname.lastname@technicolor.com

ABSTRACT

This paper investigates several approaches to address the acoustic scene classification (ASC) task. We start from low-level feature representation for segmented audio frames and investigate different time granularity for feature aggregation. We study the use of support vector machine (SVM), as a well-known classifier, together with two popular neural network (NN) architectures, namely multilayer perceptron (MLP) and convolutional neural network (CNN), for higher level feature learning and classification. We evaluate the performance of these approaches on benchmark datasets provided from the 2013 and 2016 Detection and Classification of Acoustic Scenes and Events (DCASE) challenges. We observe that a simple approach exploiting averaged Mel-log-spectrogram, as an extremely compact feature, and SVM can obtain even better result than NN-based approaches and comparable performance with the best systems in the DCASE 2013 challenge.

***Index Terms*—** Acoustic scene classification, Audio features, Multilayer Perceptron, Convolutional Neural Network, Support Vector Machine.

1. INTRODUCTION

Acoustic scene classification (ASC), a particular form of audio classification, consists in using acoustic information (audio signals) to imply about the context of the recorded environment [1]. Examples of such environments are bus, office, street, etc... It offers a wide range of applications in connected home, *e.g.* expensive video cameras can be replaced by cheap microphones for monitoring daily activity, and for smartphones, *e.g.* they could automatically switch to silence mode during a meeting or automatically increase the sound volume in a noisy environment. However, real-life ASC is not a trivial task as recognising a greater variety of sounds in both indoor and outdoor environments would require a new set of strategies and adjustments of existing machine learning techniques to make the most out of the available data.

While speaker identification [2], speech recognition [3], and some audio classification tasks in music information retrieval such as music genre recognition [4, 5] or music instrument recognition [5] have existed for long time, the real-life ASC task has become active quite recently in the research community. This can be seen by the new initiative of DCASE challenges in 2013 and 2016 which aims to provide a benchmark for the task. Various techniques have been proposed to tackle the problem with the use of different acoustic features (*e.g.* cochleogram representation, wavelets, auditory-motivated representation, features learned by neural networks) and different classifiers (*e.g.* Support Vector Machine (SVM), Gaussian Mixture Model (GMM), Hidden Markov Model (HMM)) [6]. One of the most popular approaches, known as bag-of-frames (BOF) approach [7, 8] is used as a baseline in the DCASE challenge, and exploits the long-term statistical distribution (by GMM) of the short-term MFCCs. Besides the DCASE challenge, nonnegative matrix factorization (NMF) was recently exploited for sound event detection in real life recordings [9]; recurrent neural networks (RNN) were investigated for polyphonic sound event detection in real life recordings [10]; and deep neural networks (DNN) have been developed for sensing acoustic environment [11]. It would be interesting to note that while DNNs [12, 13] were recently applied with great success to many different audio, visual and multimedia tasks, it was less investigated within the DCASE 2013 challenge and one of the reasons would be the lack of a substantial amount of labeled data for training.

This paper aims to study the use of well-established low-level acoustic feature representations and different machine learning techniques, including DNN-based methods and SVM, for the ASC task. While most existing approaches extract an acoustic feature vector for each short-term audio frame, then perform a frame-based classification based either on BOF over GMMs [7, 8] or simple majority voting [14, 15, 6], we investigate the use of an another feature representation, *i.e.* a single vector for a whole audio scene, aiming an extremely compact representation that greatly reduces the computation cost for the whole ASC system. We evaluate the use of this compact feature with SVM and MLP on both DCASE 2013 and DCASE 2016 datasets and interestingly the performance are more or less equivalent to a frame-based approach with majority voting strategy. Furthermore, it results in classification accuracy comparable to the best systems participating in the DCASE 2013 challenge.

The rest of the paper is organized as follows. In Section 2 we present the general framework which involves different approaches for feature extraction and classification. Experiment results on DCASE dataset obtained by our approaches and some state-of-the-art methods are discussed in Section 3. We finally conclude in Section 4.

2. ACOUSTIC SCENE CLASSIFICATION FRAMEWORK

The general workflow of an ASC system is usually divided into two major steps as shown in Fig. 1. In the first, the feature extrac-

*Part of this work has been done while the first author was with Technicolor.

†Email:quang-khanh-ngoc.duong@technicolor.com

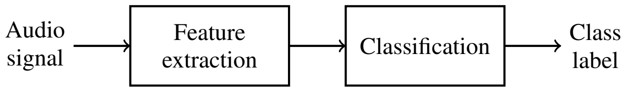

Figure 1: General workflow of the acoustic scene classification framework.

tion step, various types of hand-crafted representations have been considered in the literature such as chroma, pitch, spectrograms, zero-crossing rate, and linear predictive coding coefficients, etc., [1]. Among them, features based on Mel-frequency Cepstrum Coefficients (MFCCs) computed for each short-time frame are arguably the most common one. More recent DNN-based approaches usually learn higher level features from these low-level signal representations [11, 16]. In the classification step, the most popular classifiers would be SVM or GMM [6]. In the following, we will first describe the standard Mel-log-spectrogram, as the low-level feature used in this work, and the proposed compact representation from it in Section 2.1. We then briefly present some exploited classification approaches in Section 2.2. The choice of hyperparameters for both feature extraction and classifiers is discussed in Section 2.3.

2.1. Feature extraction

The time domain audio signal $x(n)$ is first transformed into the frequency domain by means of the short-term Fourier transform (STFT) as

$$\mathrm{STFT}\{x\}(m,\omega) = \sum_{n=-\infty}^{+\infty} x(n)w(n-mL)e^{-j\omega n} \tag{1}$$

where $w(n)$ is a window function (which is Hanning window in our implementation), m denotes frame index and L the frame shift. The spectrogram is then defined as

$$\mathbf{S}(m,\omega) = |\,\mathrm{STFT}\{x\}(m,\omega)|^2 \tag{2}$$

In our DNN-based system, we use spectrogram with a logarithmic amplitude scale (named log-spectrogram) as the frame input feature which is computed as

$$\mathbf{F}_{\text{Log-spec}}(m,\omega) = \log(\mathbf{S}(m,\omega)). \tag{3}$$

In our other systems, we first map the spectrogram $\mathbf{S}(m,\omega)$ into the auditory-motivated Mel frequency scale - denoted by $\mathbf{MS}(m,\omega)$, then transform it into logarithmic scale as

$$\mathbf{F}_{\text{Mel-log-spec}}(m,\omega) = \log(\mathbf{MS}(m,\omega)). \tag{4}$$

Note that with the CNN-based system, we use the raw log-spectrogram as the input feature in order to give flexibility for the CNN to learn a higher level feature representation optimized for the ASC task. For SVM-based systems we have tested four different features: spectrogram, log-spectrogram, Mel-log-spectrogram, and MFCC, and found that the two last ones result in a very similar ASC performance that is better than the two first ones. As the Mel-log-spectrogram is simpler to compute than MFCC, we focus more on it in this paper. Finally, we propose to average the feature vectors for all frames so as to present a whole audio example by an extremely compact feature vector whose entries are computed as

$$\mathbf{f}_{Avg-mel-log-spec}(\omega) = \frac{1}{M}\sum_{m=1}^{M} \mathbf{F}_{\text{Mel-log-spec}}(m,\omega). \tag{5}$$

2.2. Classification approaches

2.2.1. *Support vector machine*

SVM has been known as one of the most popular classifiers for many different tasks. It was also widely used in the DCASE 2013 challenge [6]. In our work, we used SVM as a benchmark classifier to evaluate the effectiveness of different features, as mentioned in Section 2.1, as well as to obtain the optimal choice of hyperparameters (*e.g.* the window size and the number of Mel-frequency coefficients) for the considered task.

In our implementation, we train SVMs using a coordinate descent algorithm and following a one-vs-the-rest scheme to perform classification of multiple classes [17]. We have tested SVM with linear kernel and Gaussian radial basis function (RBF) kernel and found that the linear kernel works slightly better than RBF kernel for the DCASE 2013 dataset.

2.2.2. *Multilayer Perceptron*

Multilayer Perceptron (MLP) is a fully connected feedforward artificial neural network architecture that maps sets of input data onto a set of appropriate outputs. It can be seen as a logistic regression classifier where the input is first transformed using a non-linear transformation [18, 19]. A typical set of equations for an MLP is the following. Layer k computes an output vector $\mathbf{h}^k$ using the output $\mathbf{h}^{k-1}$ of the previous layer, starting with the input $\mathbf{x} = \mathbf{h}^0$,

$$\mathbf{h}^k = f(\mathbf{W}^k\mathbf{h}^{k-1} + \mathbf{b}^k) \tag{6}$$

where $\mathbf{b}^k$ denotes a vector of offsets (or biases) and $\mathbf{W}^k$ a matrix of weights. The function f is called the activation function and it is applied element-wise. Common options for it are sigmoid function, hyperbolic tangent, and rectified linear unit (ReLU). The latter, *i.e.* $f(x) = \max(0,x)$, was used to obtain the results reported in this document.

The top layer output is used for making a prediction and is combined with the groundtruth label into a loss function. We use softmax as the classification layer and the loss function is regularized with ℓ_1 and ℓ_2 penalties. This cost function is then optimized using mini-batch stochastic gradient descent (SGD) with an adaptive learning rate [20] and dropout is performed between the hidden layers [21].

2.2.3. *Convolutional Neural Network*

Convolutional Neural Network (CNN) is a type of neural network designed to exploit the redundancy and correlation between neighbour units. It has gained great success in different fields such as image and video recognition, natural language processing, speech recognition, etc., [13]. This motivates us to investigate the use of CNN for the ASC task in this work.

We trained CNNs over the log-spectrogram of the signals with a structure of vertical filters, *i.e.* the frequency bins can also be interpreted as a CNN channel (as in RGB channels for images) instead of a dimension and the convolution is ran over the time axis. This type of structure was proposed for music recommendation in Spotify[1] and is justifiable by the fact that an audio "pattern" detected in a high-frequency region is usually different from that same pattern in a low-frequency region. Thus it is desirable to model the vertical

[1] http://benanne.github.io/2014/08/05/spotify-cnns.html

Method	Acoustic feature	Classifier	Accuracy
Baseline	MFCC	"bag-of-frames" GMM	55%
Chum *et al.* [22]	Various features at 2 frame sizes	HMM	64%
Geiger *et al.* [14]	Diverse features	SVM + majority voting	69%
Nam *et al.* [23]	Learned by sparse RBM, event detection and max-pooling	SVM	60%
Nogueira *et al.* [24]	MFCCs + temporal modulations, event density estimation, binaural modelling features, feature selection	SVM	60%
Rakotomamonjy and Gasso [25]	Computer vision based features applied to constant-Q spectrogram	SVM	69%
Li *et al.* [15]	Wavelets, MFCC and others	Treebagger + majority voting	72%
Roma *et al.* [26]	MFCC with Recurrence Quantification Analysis	SVM	76%
Proposed SVM-A	**Averaged Mel-log-spectrogram**	**Linear SVM**	**75%**
Proposed SVM-V	**Frame Mel-log-spectrogram**	**Linear SVM + majority voting**	**78%**
Proposed MLP	Averaged Mel-log-spectrogram	MLP with softmax as classification layer	72%
Proposed CNN	Log-spectrogram	CNN with softmax as classification layer	62%

Table 1: Acoustic scene classification results with DCASE 2013 test dataset (for state-of-the-art approaches) and development dataset (for our proposed approaches and Li *et al.*). Note that other submitting systems resulting in less classification accuracy are not mentioned in the table.

Method	Acoustic feature	Classifier	Accuracy
Baseline	MFCC	"bag-of-frames" GMM	75%
Proposed SVM-A	**Averaged Mel-log-spectrogram**	**Linear SVM**	**80%**
Proposed SVM-V	**Frame Mel-log-spectrogram**	**Linear SVM + majority voting**	**78%**
Proposed MLP	Averaged Mel-log-spectrogram	MLP with softmax as classification layer	75%
Proposed CNN	Log-spectrogram	CNN with softmax as classification layer	59%

Table 2: Acoustic scene classification results with DCASE 2016 development dataset.

filters to extract more meaningful information from the spectral representation. More details about the implemented CNN architecture can be found in Section 3.1.

2.3. Hyperparameter optimization

It is known that the choice of hyperparameters in each step of the ASC system or in any machine learning task can significantly affect the final classification result. Such hyperparameters would be *e.g.* the window length and hop length in the STFT computation for feature extraction, the regularization parameter for SVM, the number of hidden units in an MLP, and the step size for the SGD algorithm in DNN based methods, etc. The conventional strategy of tuning these parameters manually would not be feasible as it requires a great number of trials so that all parameters can be optimized together. Thus, in this work we incorporate the Bayesian optimization [27] method to find these parameters altogether. This algorithm models the generalization accuracy of a classifier as a function of the corresponding parameters, and finds the optimal parameters that maximize the expected accuracy given the observed dataset. Typically, this kind of algorithm can be considered as a class of sequential model-based optimization (SMBO) [28].

In our implementation, we use Hyperopt [29], a Python library for optimizing hyperparameters in machine learning algorithms, with the Tree of Parzen Estimators (TPE) algorithm [30], that performs cross-validation with the development datasets of DCASE 2013 and DCASE 2016 and finds an optimal set of hyperparameter values. It is interesting to note that the optimal window size for STFT computation found by the TPE algorithm is quite long, *i.e.* about half of a second. This can be explained by the fact that the acoustic events are more spread in time compared to *e.g.* speech which is very localized so as the window length used for STFT is usually much smaller.

3. EXPERIMENTS

We evaluate the ASC performance of our four implementing systems with the benchmark DCASE 2013 dataset, which allows to compare with the state-of-the-art approaches participating in the challenge, in Section 3.1. We then present the result with DCASE 2016 dataset in Section 3.2. Our first system (named *Proposed SVM-A*) uses an extremely compact feature as the Mel-log-spectrogram coefficients averaged for all frames, and SVM with a linear kernel as classifier. The second system (named *Proposed SVM-V*) performs frame classification by SVM with a linear kernel, then majority voting in the end. The third system (named *Proposed MLP*) takes the compact averaged Mel-log-spectrogram as input, learns a higher feature representation by MLP, then classifies by softmax as the last layer of the MLP. The fourth system (named *Proposed CNN*) takes log-spectrogram as low-level input feature, learns higher feature representations by CNN layers, then classifies by softmax.

3.1. Results with the DCASE 2013 dataset

The DCASE 2013 dataset consists of 30-second audio segments belonging to 10 classes, namely: *bus, busy street, office, open air market, park, quiet street, restaurant, supermarket, tube, tube station*. Each class has 10 segments in the development set and 10 other examples in the test set [31].

The ASC performance was evaluated in terms of the classification accuracy, averaged over all classes, and shown in Table 2.

	Beach	Bus	Cafe/restaurant	Car	City center	Forest path	Grocery store	Home	Library	Metro station	Office	Park	Residential area	Train	Tram
Beach	47	0	3	4	5	4	1	0	1	0	3	4	2	0	2
Bus	2	61	0	4	1	0	0	0	0	2	0	2	2	3	1
Cafe/restaurant	4	0	46	0	0	2	19	4	0	0	0	2	0	0	1
Car	0	2	0	71	0	0	0	0	0	0	0	0	1	0	4
City center	0	0	0	0	75	0	0	0	0	2	0	0	1	0	0
Forest path	1	0	0	0	0	75	0	0	0	0	0	0	2	0	0
Grocery store	0	0	0	0	2	0	76	0	0	0	0	0	0	0	0
Home	6	0	1	0	0	13	1	30	7	0	3	6	9	1	1
Library	0	1	0	0	0	0	0	0	72	1	0	0	0	4	0
Metro station	0	0	0	0	0	0	0	0	0	76	0	2	0	0	0
Office	0	0	0	0	0	0	0	8	1	0	69	0	0	0	0
Park	9	0	5	0	2	1	0	1	0	0	0	49	11	0	0
Residential area	5	0	0	0	9	7	0	0	2	0	0	19	35	0	1
Train	17	3	0	0	2	0	0	0	1	0	0	1	0	49	5
Tram	2	0	0	0	0	0	0	0	0	2	0	0	0	0	74

Figure 2: Confusion matrix after a 4-fold cross-validation over 78 samples of each class.

Note that as we do not have access to the groundtruth of the test set, we evaluated our systems averaging with the standard 5-fold cross-validation on the development set only, while results for most other approaches in the table are obtained with the test set [6]. Some hyperparameters for each systems were found by the Bayesian optimization method presented in Section 2.3. More detailed settings for each system are as follows. The window length for the STFT was set by 0.57 seconds and 0.41 seconds for the SVM-A and SVM-V system, respectively, the number of Mel-frequency coefficients is about 1900, the regularization parameter C in SVM for SVM-A and SVM-V were 0.98 and 0.62, respectively. MLP had one hidden layer with 677 units, dropout rate and learning rate for parameter training was set by 0.08 and 0.011, respectively. CNN had 3 convolutional layers, the number of filters for each layer are 50, 29, and 19, respectively, and the max-pooling ratios between layers are 3, 4, and 3.

As it can be seen, better results are usually obtained by systems using SVM as classifier. This can be explained by the fact that the dataset may be not large enough for training DNNs directly. Three of our proposed systems (SVM-A, SVM-V, and MLP) achieve comparable performance with some of the best approaches in the challenge - as we suppose that there is not much difference between development set and the test set. Moreover, we achieve higher accuracy than Li *et al.* [15] in the same development set. Finally it is very interesting to note that the proposed feature, which is extremely compact so as to represent a whole 30-second audio segment by just a single vector, can be sufficient for the classification as the SVM-A and MLP obtained 75% and 72% accuracy, respectively.

3.2. Results with the DCASE 2016 dataset

The DCASE 2016 dataset is structured in a similar way as the DCASE 2013 dataset. However the number of acoustic classes is extended to 15 (*bus, cafe/restaurant, car, city center, forest path, grocery store, home, lakeside beach, library, metro station, office, residential area, train, tram and urban park*), and the number of examples for each class is significantly enlarged to 78 for the development set and 78 for the test set.

The results for development set obtained by our four systems are shown in Table 2.2.3, where the best performance of 80% is achieved by the SVM-A system with a window length of 0.42 seconds and a hop size of 0.14 seconds for the STFT computation. This result confirms again the benefit of using the proposed compact feature representation and the use of a long window for the spectral transformation. The MLP, which obtains similar performance as the baseline, had two hidden layers with 66 and 199 units, respectively, SGD was used for parameter training with learning rate of 0.003 and batch size of 100, weights for ℓ_1 and ℓ_2 penalties were 10^{-5} and 10^{-4}, respectively. The CNN, with the same configuration used for the DCASE 2013 dataset, still resulted in the lowest performance. These four systems will also be tested with the test set for participating in the DCASE 2016 challenge.

The confusion matrix for SVM-A is shown in Fig. 2, where rows are groundtruth, columns are the inferred class label, and values are number of the classified acoustic scene. As it can be seen, some environments containing a specific type of noise (such as car, metro station, forest path) are quite easy to recognize, while some others (such as home, residential area, park) are quite confusing.

4. CONCLUSION

In this article we present several approaches for the ASC task, targeting on fast systems working with very compact feature representations so that ASC can be implemented *e.g.* in smartphones. We investigate the use of Bayesian optimization for hyperparameter optimization and find its benefit in *e.g.* choosing the optimal window length for STFT or setting DNN parameters. By evaluating on benchmark DCASE datasets, we find that (1) a long window size for spectral transformation is more relevant for the environmental acoustic scenes, and (2) a very compact feature representation by long-term temporal averaging of Mel-log-spectrogam coefficients would be sufficient for the task compared to more complicated approaches. Finally, it is worth noting that DNN approaches have not reached the same performance of the more classical SVM based systems so far. Thus future work would be devoted to investigate transfer learning strategies for DNN based systems where part of the DNN can be initially learned by a large amount of external audio data.

5. REFERENCES

[1] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, “Acoustic scene classification: Classifying environments from the sounds they produce,” *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[2] D. Reynolds, R. C. Rose, *et al.*, “Robust text-independent speaker identification using gaussian mixture speaker models,” *Speech and Audio Processing, IEEE Transactions on*, vol. 3, no. 1, pp. 72–83, 1995.

[3] L. Rabiner and B.-H. Juang, “Fundamentals of speech recognition,” 1993.

[4] T. Li, M. Ogihara, and Q. Li, “A comparative study on content-based music genre classification,” in *Proceedings of the 26th*

annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003, pp. 282–289.

[5] B. Kostek, A. Kupryjanow, P. Zwan, W. Jiang, Z. W. Raś, M. Wojnarski, and J. Swietlicka, "Report of the ISMIR 2011 contest: music information retrieval," in *Foundations of Intelligent Systems*. Springer Berlin Heidelberg, 2011, pp. 715–724.

[6] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *Multimedia, IEEE Transactions on*, vol. 17, no. 10, pp. 1733–1746, 2015.

[7] J.-J. Aucouturier, B. Defreville, and F. Pachet, "The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music," *The Journal of the Acoustical Society of America*, vol. 122, no. 2, pp. 881–891, 2007.

[8] M. Lagrange, G. Lafay, B. Defreville, and J.-J. Aucouturier, "The bag-of-frames approach: a not so sufficient model for urban soundscapes," *The Journal of the Acoustical Society of America*, vol. 138, no. 5, pp. EL487–EL492, 2015.

[9] A. Mesaros, T. Heittola, O. Dikmen, and T. Virtanen, "Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 151–155.

[10] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6440–6444.

[11] N. D. Lane, P. Georgiev, and L. Qendro, "Deepear: Robust smartphone audio sensing in unconstrained acoustic environments using deep learning," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '15. ACM, 2015, pp. 283–294.

[12] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[14] J. T. Geiger, B. Schuller, and G. Rigoll, "Recognising acoustic scenes with large-scale audio feature extraction and SVM," Tech. Rep., 2013.

[15] D. Li, J. Tam, and D. Toub, "Auditory scene classification using machine learning techniques," Tech. Rep., 2013.

[16] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1798–1828, 2013.

[17] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[18] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

[19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, p. 3, 1988.

[20] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *The Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[21] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[22] M. Chum, A. Habshush, A. Rahman, and C. Sang, "IEEE AASP scene classification challenge using hidden Markov models and frame based classification," Tech. Rep., 2013.

[23] J. Nam, Z. Hyung, and K. Lee, "Acoustic scene classification using sparse feature learning and selective max-pooling by event detection," Tech. Rep., 2013.

[24] W. Nogueira, G. Roma, and P. Herrera, "sound scene identification based on MFCC, binaural features and a support vector machine classifier," Tech. Rep., 2013.

[25] A. Rakotomamonjy and G. Gasso, "Histogram of gradients of time-frequency representations for audio scene classification," Tech. Rep., 2013.

[26] G. Roma, W. Nogueira, P. Herrera, and R. de Boronat, "Recurrence quantification analysis features for auditory scene classification," *IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep*, 2013.

[27] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, 2012, pp. 2951–2959.

[28] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *International Conference on Learning and Intelligent Optimization*. Springer, 2011, pp. 507–523.

[29] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms," in *Proceedings of the 12th Python in Science Conference*. Citeseer, 2013, pp. 13–20.

[30] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Advances in Neural Information Processing Systems*, 2011, pp. 2546–2554.

[31] D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: An IEEE AASP challenge," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.