i-ETC: ISEL Academic Journal of Electronics,
Telecommunications and Computers
CETC2016 Issue, Vol. 3, n. 1 (2017) ID-2

Automatic Acoustic Scene Classification

Gonçalo Marques[a], Thibault Langlois[b]
[a]ISEL, Electronic Telecommunications and Computer Department, Lisbon, Portugal
[b]FCUL, Informatics Department, Lisbon, Portugal
gmarques@deetc.isel.pt tl@di.fc.ul.pt

***Abstract*— This paper presents a baseline system for automatic acoustic scene classification based on the audio signals alone. The proposed method is derived from classic, content-based, music classification approaches, and consists in a feature extraction phase followed by two dimensionality reduction steps (principal component analysis and linear discriminant analysis) and a classification phase done using a k nearest-neighbors algorithm. This paper also reports on how our system performed in the context of the DCASE 2016 challenge, for the acoustic scene classification task. Our method was ranked fifteenth amongst forty nine contest entries, and although it is below the top performing algorithms, in our perspective it is still interesting to see a low-complexity system such as ours obtain fairly good performances.**

Keywords: Machine Learning, Signal Processing, Music Information Retrieval, Bag of Frames.

I. Introduction

Automatic identification of sound sources in an urban environment has a huge potential in several applications related to the current panorama of intelligent cities. These applications include monitoring systems able to recognize activities, sound environments, and create city sound maps to provide to the general public information about environmental noise, or other acoustic factors. However, a lot of research is still needed to reliably detect and recognize sound events and scenes in realistic environments where multiple sources, often distorted, are simultaneously present. This work focuses on one particular aspect of urban sound analysis: acoustic scene classification.

The system we propose is a classical classification system in the sense that it uses typical machine-learning data transformation and classification algorithms in the decision making process. First, each audio excerpt is converted into a single feature vector which is the representation of choice for standard machine-learning methods. Then, the whole dataset is transformed via principal component analysis (PCA) [10], an unsupervised dimensionality reduction technique, followed by a linear discriminant analysis (LDA) projection [9]. LDA is a supervised process, and the projection tries to maximize the ratio between intra and inter class scatter, but it is not a classification method since no decision is involved. For classification, a k-nearest neighbors (k-NN) algorithm was used [7]. The experimental configuration used in our tests is common in many audio classification works (or at least parts of it – see for instance [6], [11], [14], [16], [18]) and therefore it does not bring any original contribution in terms of the algorithmic setup. In fact, our system falls under the standard "bag of frames" classifiers commonly used in music information retrieval applications (see [3], [4], [12], [17] and references therein). Our main objective was not to bring forth a new audio classification or feature extraction method, but rather see how a simple, non parametric algorithm performed in the acoustic scene classification challenge. We used the same data partitioning and cross-validation setup provided with the database and our results are a bit better than the ones reported in [13]. The method in [13] was the baseline system provided with the challenge and ranked twenty eighth while our system ranked fifteenth (amongst a total of forty nine entries). However, the experiments we conducted also revealed some unexpected variations in terms of accuracy when the whole dataset or just part of it was used to estimate the PCA and LDA projections. This is an indication that there may be differences between feature class-dependent distributions among folds. The structure of the remainder of this paper is as follows: Section II describes the data and the feature extraction process used in our experiments, Section III describes our approach to acoustic scene classification, followed by Section IV where we present our results. In this section, we also report on the challenge results and how we faired against other contestants. Section V concludes this paper.

II. Data and Feature Extraction

The dataset used in this work was created in the context of the DCASE2016 challenge [1] for the acoustic scene classification task. The dataset contains 1170, 30-seconds audio excerpts from the following acoustic scenes: Beach, Bus, Café/Restaurant, Car, City Center, Forest Path, Grocery Store, Home, Library, Metro Station, Office, Urban Park, Residential Area, Train, and Tram. The dataset is divided into four folds for cross-validation testing. We used the same data partition in our experiments and our results are averaged over the four test folds.

The features used are the all-purpose Mel frequency cepstral coefficients (MFCCs), a very popular representation in speech recognition (see for instance [15]), and also widely used in content-based music information applications. The audio was decomposed into 23 ms segments (1024 samples at 44.1 kHz) with 50% overlap, and we used 100 Mel bands to extract 23 MFCCs plus the zero order MFCC and the frame's log-energy, plus the delta and acceleration coefficients. This means that the audio is converted into a sequence of $25 \times 3 = 75$ dimensional vectors. We applied the VoiceBox software [2] to extract the features. In order to convert each audio excerpt into a single feature vector, the sequence of MFCC features is summarized using the median and logarithmic standard deviation. The median was used instead of the mean since this statistic is more robust to outliers. The log-standard deviation is given by $20\log_{10}(\sigma_i)$ where σ_i is the standard deviation of feature i (with $i = 1,\ldots,75$). The reason to use the log-standard deviation instead of plain standard deviation was to convert these feature values to an order of magnitude comparable to the median feature values - otherwise the standard deviation values would be a few orders of magnitude lower, and during the PCA pre-processing step, these dimensions would be discarded as noise since they would not contribute in any significant way to the overall data variance. The statistics return two 75-dimensional vectors which are concatenated, so each audio excerpt is represented by a 150-dimensional feature vector.

III. Method

The proposed classification approach is divided into three main blocks: feature pre-processing via principal component analysis, feature transformation by linear discriminant analysis and finally a classification step performed by a k-nearest neighbor classifier.

Principal Component Analysis: PCA is a standard dimensionality reduction technique, where the data is decorrelated by projecting it into orthogonal directions of maximum variance. These directions, the principal components, are obtained using a eigen-decomposition of the data covariance matrix, and in our experiments we kept enough components to explain 99.9% of the total data variance. The PCA-transformed data was also whitened - each data dimension was scaled in order to have unit variance.

Linear Discriminant Analysis: LDA is commonly used as a pre-processing step for pattern classification. It is also a dimensionality reduction technique since the data is projected into $c-1$ dimensional space where c is the total number of classes ($c = 15$ for this challenge).

LDA is a supervised learning method, and therefore the projection should be calculated with the training set only, otherwise we are indirectly including information about the class labels in the test set. Estimating the LDA projection with the whole dataset can result in overly optimistic performance values, specially when there is a relatively large number of classes and a relatively low number of examples, as in the case of this challenge dataset. We tested the performance of our system using the whole dataset to estimate the LDA projection in order to assess the increase in performance compared to the “correct” evaluation procedure. The results showed a significant increase in performance, which in our perspective, is somewhat surprising. These are reported in Section IV-B, along with a discussion on possible causes of such a performance discrepancy.

k-Nearest Neighbors: k-NN is an instance-based learning, where class membership of a pattern is assigned based on a majority vote of its neighbors. k-NN is possibly one of the simplest classification methods, and therefore it is well suited for a baseline system. We tested two distance metrics with the k-NN algorithm, the cosine and the Euclidean distance, and opted for the Euclidean distance because it yielded slightly better accuracy results. We also ran the algorithm with different number of neighbors (from 5 to 31 - using an increment of two) and chose empirically $k = 9$. The results reported in Section IV are obtained using the Euclidean distance metric, and $k = 9$.

IV. Experimental Results

This section is divided into three subsections. In the first, we present the results obtained with our method. The experimental setup is described, the system performance is measured in terms of accuracy, either with mean or class specific values. In the second, we present the performances obtained when the whole dataset is used to estimate the LDA projection. This is not the correct procedure to estimate our system performance. The intent is to have an idea of by how much the performance values are inflated. In the third part, we give a brief overview of the results on the DCASE-2016 acoustic scene classification challenge.

A. System Performance

The results presented in this section were obtained using the following experimental setup. The PCA and the LDA projections were estimated using only the training set. In our tests, we used 4-fold cross validation and the same data partitioning provided with the dataset. This means that a total of four LDA projections where estimated with three training folds. The presented result pertain to the tests folds only. The obtained average accuracy was 77.6%. In Table I reports the (average) accuracies per class. These context-wise performances vary from 52.6% (Train class) to 93.6% (City Center and Metro Station classes). Table II shows the accuracies per fold.

TABLE I

Accuracy per class. Accuracy values obtained with the mean of all four test folds.

1. Beach	76.9%
2. Bus	66.7%
3. Café/Restaurant	79.5%
4. Car	84.6%
5. City Center	93.6%
6. Forest Path	87.2%
7. Grocery Store	82.1%
8. Home	64.1%
9. Library	87.2%
10. Metro Station	93.6%
11. Office	92.3%
12. Urban Park	60.3%
13. Residential Area	65.4%
14. Train	52.6%
15. Tram	87.2%

Figure 1 shows the confusion matrix (obtained summing the four confusion matrices in each test fold). Each line refers to the examples of a single class; the class order is the same as the one in Table I. The columns refer to the classification results.

Fig. 1. Confusion matrix - the rows represent the true classes, the columns represent the classification results. The class order is the same as the one given in Table I: in the first row are the samples from the class Beach, in the second from class Bus, and so on. This matrix was obtained by the sum of the four confusion matrices - one per test fold. A perfect classification would result in this matrix having the value 78 on the main diagonal (number of examples per class) and zeros for the rest of the entries.

60	0	0	0	**1**	**2**	0	0	**1**	0	**1**	**9**	**3**	0	**1**
0	**52**	**6**	**2**	0	0	**1**	0	**5**	0	0	0	0	**11**	**1**
0	0	**62**	0	0	**2**	**8**	0	**2**	**4**	0	0	0	0	0
0	**3**	0	**66**	0	0	0	0	**1**	0	0	0	0	**7**	**1**
1	0	0	0	**73**	0	0	0	**1**	**1**	0	0	**2**	0	0
1	0	0	0	0	**68**	0	0	0	0	**2**	**7**	0	0	0
0	0	**4**	0	0	0	**64**	0	**1**	**9**	0	0	0	0	0
2	**2**	**7**	0	0	**2**	0	**50**	**8**	0	**6**	0	0	0	**1**
0	0	**9**	0	0	0	**1**	0	**68**	0	0	0	0	0	0
0	0	0	0	0	0	**2**	0	**3**	**73**	0	0	0	0	0
0	0	0	0	0	**1**	0	**5**	0	0	**72**	0	0	0	0
2	**1**	0	0	**1**	**13**	**3**	0	**6**	0	0	**47**	**5**	0	0
2	0	**2**	0	**1**	**3**	0	0	0	0	**1**	**17**	**51**	0	**1**
0	**11**	**8**	0	0	**2**	**4**	0	0	**5**	**1**	0	0	**41**	**6**
0	**1**	0	0	0	0	**7**	0	0	0	**1**	0	0	**1**	**68**

For example, for the class Beach, 60 audio excerpts were correctly classified, 9 were classified as the class Urban Park, and nine others were also misclassified. The results also reveal some particular error correlations among certain classes. For instance many Residential Area samples were misclassified as Urban Park (a total of 17 errors). Five Urban Park pieces were attributed to Residential Area, however the excerpts of this class have a higher tendency to get confused with another class: Forest Path (a total of 13 errors). These mislabeling seems understandable since these acoustic scenes share some resemblances. Another example of classes that have similar acoustic characteristics and a high number of errors between them are Bus and Train. Other relations that seem to make some sense could be found such as the case of Beach and Urban Park, or Home and Library, but further tests would be needed to determine if a real correlation exists.

TABLE II

Accuracy per fold. The mean accuracy is 77.4%. These results were obtained using only the training set to estimate the PCA projection.

	Fold 1	Fold 2	Fold 3	Fold 4
Accuracy	79.3%	71.7%	82.6%	76.0%

B. LDA estimation with the whole dataset

In this section, we discuss the results obtained when we used the whole dataset to estimate LDA projection. This is not the correct testing methodology, since when doing this, we are implicitly including test label information in the model training process. The intent here is just to determine by how much the performances are over evaluated when using this incorrect experimental procedure.

We performed some tests in order to have an idea of how the classification performance is affected by using just part or the whole dataset. The results (see Table III) show that there is no significant decrease in accuracy (less than 1%) when the PCA projection is estimated with only the training set. Nevertheless, when we applied the same methodology to estimate the LDA projection, the results were significantly affected. Table IV shows the accuracies per test fold. The mean accuracy is 90.8% which is 12.6% points higher than our baseline system. Since LDA is a supervised technique, an increase in performance is expected when the entire dataset is used, but in our perspective such a high bias was unforeseen. This is also an indication that there is some variability of class-dependent feature distributions among folds. The partition process used for this dataset may be the cause, since it was based on recording location [13]. This division was done in order to avoid overestimating systems performances, since in this way, segments from a single recording are assigned to only one fold. We believe that the variability between folds is also due to the relatively low number of examples per class, and increasing the number of examples in the database will reduce this variation.

Figure 2 shows the confusion matrix. There is some similarities to the error patterns found in Section IV-A. For instance, the Residential Area samples are still misclassified as Urban Park, Urban Park as Forest Path, and Home as Library. Other errors though, like the confusion between Bus and Train classes, have almost vanished.

TABLE III

Accuracy per fold. The mean accuracy is 78.2%. These results were obtained using the whole dataset to estimate the PCA projection. The whole dataset was used to measure the increase in performance compared to using only the training set for the PCA estimations (results in Table II).

	Fold 1	Fold 2	Fold 3	Fold 4
Accuracy	79.0%	72.1%	82.9%	78.8%

TABLE IV

Accuracy per fold. The mean accuracy is 90.8%. These results were obtained using (inappropriately) the whole dataset to estimate the LDA linear projection.

	Fold 1	Fold 2	Fold 3	Fold 4
Accuracy	95.9%	85.9%	92.3%	89.0%

C. DCASE 2016 Challenge Results

The DCASE 2016 challenge, provided two datasets for acoustic scene classification. The first one, described in Section II, was created for developing purposes and the results reported in the previous sections were based on this dataset. A second dataset consisting of 390 audio excerpts belonging to one of 15 classes was also provided without the ground truth

Fig. 2. Confusion matrix obtained by the sum of the four confusion matrices - one per test fold. The results were obtained using (inappropriately) the whole dataset to estimate the LDA linear projection.

67	0	0	0	**1**	**2**	**1**	0	**1**	0	0	**2**	**3**	0	**1**
0	**72**	**3**	**1**	0	0	0	0	**1**	0	0	0	0	**1**	0
0	0	**77**	0	0	0	**1**	0	0	0	0	0	0	0	0
0	**1**	0	**77**	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	**76**	0	0	0	0	0	0	0	**1**	0	0
0	0	0	0	0	**75**	0	0	0	0	0	**3**	0	0	0
0	0	**1**	0	0	0	**73**	0	0	**4**	0	0	0	0	0
1	**1**	**2**	0	0	0	**1**	**61**	**8**	0	**3**	0	0	0	**1**
0	0	**2**	0	0	0	0	0	**76**	0	0	0	0	0	0
0	0	0	0	0	0	0	0	**1**	**77**	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	**78**	0	0	0	0
2	0	0	0	0	**7**	**2**	0	**4**	0	0	**62**	**1**	0	0
1	0	0	0	0	**2**	0	0	**2**	0	0	**14**	**58**	0	**1**
0	**1**	**3**	0	0	0	**3**	0	**1**	**2**	**1**	0	0	**60**	**7**
0	0	0	0	0	0	**3**	0	0	0	**1**	0	0	**1**	**73**

for evaluation purposes. Our submission obtained an accuracy of 83.1%, and ranked 15thplace among 49 submissions (see Table V for the top 20 results). The first place submission [8] achieved a 89.7% accuracy, with a fairly complex approach. Their system was based on a combination of different scores obtained with distinct features such as spectrograms or I-vectors [5] (using the audio stereo information), and also distinct classifiers such as Deep Convolutional Neural Networks or creating an Universal Background Model for the I-vectors using a mixture of Gaussians. In the end, each audio piece was represented by 16 different scores, which were fused together in order to provide the class estimate. The majority of other submissions also used some sort of neural network (NN) for classification (recurrent-NN, convolutional-NN, deep-NN), and various sets of features. The second method of choice was support vector machines (SVM), while the third was a combination of various classification algorithms through fusion or ensemble methods. In terms of algorithmic complexity, our method is by far the simplest, and in that sense, we believe that is interesting that such low-complexity system can obtain fairly reasonable performances.

V. Conclusion

In this work, we presented a baseline for acoustic scene classification system composed of two dimensionality reduction transformation (PCA and LDA) followed by a k-NN classification algorithm.We trained and tested our method on the DCASE 2016 acoustic scene classification dataset, and submitted it to the challenge provided by the organization. Our approach was not the top ranked one: 6.6% points below the 1st place. Nevertheless, the performance obtained (83.1% accuracy) is still a relatively high value, specially taking into consideration the simplicity of the method. These results also highlight the benefits of pre-processing the data with dimensionality reduction techniques.

TABLE V

Top 20 ranking positions in terms of accuracy scores for the acoustic scene classification task of the DCASE 2016 challenge (for details see their web page: `http://www.cs.tut.fi/sgn/arg/dcase2016/task-acoustic-scene-classification`.)

Rank	Acc.	Author	Classifier
1	89.7%	E-Zadeh *et al.*	fusion
2	88.7%	E-Zadeh *et al.*	I-vector
3	87.7%	Bisot *et al.*	NMF
4	87.2%	Park *et al.*	fusion
5	86.4%	E-Zadeh *et al.*	I-vector
5	86.4%	Marchi *et al.*	fusion
6	86.2%	Valenti *et al.*	CNN
7	85.9%	Elizalde *et al.*	SVM
8	85.6%	Takahashi *et al.*	DNN-GMM
9	85.4%	Kim & Lee	CNN-ensemble
10	84.6%	Han & Lee	CNN
11	84.1%	Bae *et al.*	CNN-RNN
11	84.1%	Wei *et al.*	ensemble
12	83.8%	Liu *et al.*	fusion
13	83.6%	Liu *et al.*	fusion
14	83.3%	E-Zadeh *et al.*	CNN
14	83.3%	Pham *et al.*	CNN
14	83.3%	Lidy & Schindler	CNN
15	83.1%	Bao *et al.*	fusion
15	**83.1%**	**Marques & Langlois**	**k-NN**
16	82.3%	Mun *et al.*	DNN
16	82.3%	Wei *et al.*	ensemble
17	82.1%	Yun *et al.*	GMM
17	82.1%	Rakotomamonjy	SVM
18	81.8%	Lidy & Schindler	CNN
19	81.3%	Ghodasara *et al.*	SVM
20	81.0%	Kong *et al.*	DNN
20	81.0%	Nogueira	SVM

In our tests, we also found a large variation in accuracies when the LDA transformation was estimated using the whole dataset versus using only the training set. This is an indication that there is some variability of feature distribution among folds in this particular dataset.

Acknowledgement

Gonçalo Marques thanks FI-Sonic[1] for supporting his research.

References

[1] DCASE 2016 - Detection and Classification of Acoustic Scenes and Events - `http://www.cs.tut.fi/sgn/arg/dcase2016/`.

[2] VOICEBOX: Speech Processing Toolbox for MATLAB (2005) by Mike Brookes - `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`.

[3] J.-J. Aucouturier and F. Pachet. Improving timbre similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

[4] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, 96(4):668–696, April 2008.

[5] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-end factor analysis for speaker verification. *Trans. Audio, Speech and Lang. Proc.*, 19(4):788–798, May 2011.

[1] `http://www.fi-sonic.com/`

[6] S. Dieleman and B. Schrauwen. Multiscale approaches to music audio feature learning. In *14th International Society for Music Information Retrieval Conference (ISMIR)*, pages 116–121, 2013.

[7] R. Duda, P. Hart, and D. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[8] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer. A hybrid approach using binaural i-vectors and deep convolutional neural networks. Technical report, DCASE2016 Challenge, September 2016.

[9] R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[10] H. Hotelling. Analysis of a complex of statistical variables with principal components. *Journal of Educational Psychology*, 24:417–441, 1933.

[11] C.-H. Lee, J.-L. Shih, K.-M Yu, and H.-S. Lin. Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features. *IEEE Transactions on Multimedia*, 11(4):670–682, 2009.

[12] G. Marques, T. Langlois, F. Gouyon, M. Lopes, and M. Sordo. Short-term feature space and music genre classification. *Journal of New Music Research*, 40(2):127–137, 2011.

[13] A. Mesaros, T. Heittola, and T. Virtanen. TUT database for acoustic scene classification and sound event detection. In *24th European Signal Processing*, Budapest, Hungary, 2016.

[14] S. R. Ness, A. Theocharis, G. Tzanetakis, and L. G. Martins. Improving automatic music tag annotation using stacked generalization of probabilistic SVM outputs. In *Proc. of the 17th ACM Int. Conf. on Multimedia (ACM-MM'9*, New York, U.S.A., 2009. ACM.

[15] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1993.

[16] J. Salamon and J. P. Bello. Unsupervised feature learning for urban sound classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 171–175. IEEE, 2015.

[17] B. Sturm. A survey of evaluation in music genre recognition. *Proc. Adaptive Multimedia Retrieval, Denmark*, 2012.

[18] G. Wu, J. Zhu, and H. Xu. A hybrid visual feature extraction method for audio-visual speech recognition. In *16th IEEE International Conference on Image Processing (ICIP)*, pages 1829–1832, 2009.