ACOUSTIC CONTEXT RECOGNITION USING LOCAL BINARY PATTERN CODEBOOKS

Daniele Battaglino[1,2], *Ludovick Lepauloux*[1], *Laurent Pilati*[1] *and Nicholas Evans*[2]

[1] NXP Software
Valbonne, France

[2] EURECOM
Biot, France

ABSTRACT

Automatic context recognition enables mobile devices to adapt their configuration to different environments and situations. This paper investigates the use of acoustic cues as a means of recognising context. The majority of existing approaches exploit Mel-scaled cepstral coefficients (MFCCs) developed for the analysis of speech signals. The hypothesis in this paper is that new features are needed in order to capture complex acoustic structure. The paper introduces the use of local binary pattern (LBP) analysis which is used to complement MFCCs with acoustic texture information. The second contribution relates to a bag-of-features extension which clusters LBPs into a small number of codewords. Both approaches outperform the current state of the art and the latter is particularly appealing for embedded applications in which computational efficiency is paramount.

Index Terms— acoustic context recognition, spectrogram, local binary pattern, codebook, textural features

1. INTRODUCTION

Context awareness aims to categorize the environment or situation in which a mobile device is used. User demand for customization and personalization is dependent on contextualization which requires new recognition technology in order to understand the context and automatically adapt to it [1]. In this work the context relates to the immediate environment, such as an office, in a bus or street. An example application might be to activate bluetooth functionality in order to connect a device to an audio and infotainment system when the user is in their car.

Context awareness can be achieved by interpreting information from multiple, heterogeneous sensors which provide estimates of motion, position, gravity and acceleration, for example. From this information it may be possible to determine whether a user is moving, and at what speed. This paper concerns acoustic analysis. Acoustic analysis is preferred to alternatives for two principal reasons: (i) almost all modern mobile devices are equipped with at least one microphone; (ii) acoustic analysis can help to distinguish between some contexts which might otherwise be indistinguishable, i.e. bus and car contexts in which other sensors (e.g. motion) might provide identical or similar information.

Almost all existing approaches to acoustic context recognition (ACR) are based on traditional Mel-scaled frequency cepstral coefficients (MFCCs) designed predominantly for speech processing applications such as speech or speaker recognition. Even so, recent work [2] shows that MFCC features may not be sufficiently discriminative for ACR; MFCCs capture only short term variation with minimum dynamic information whereas auto-correlation in the temporal domain can help to discriminate between different contexts. The work in [2] describes the capture of auto-correlation through a similarity matrix which reflects the recurrence between consecutive closely located frame sequences. Features are extracted using recurrence quantification analysis (RQA) of the similarity matrix. Nevertheless, RQA quantifies auto-correlation in the MFCC features, rather than capturing the complex acoustic structure across both time and frequency directly from the spectrogram. Moreover, since it operates on MFCCs, RQA cannot capture non-consecutive structure at the sub-band level; MFCCs reflect the full-band spectral envelope, whereas recurrent acoustic structure is generally observed at the sub-band level.

With the goal of improving ACR performance, this paper reports our recent work to characterise the distribution of acoustic structure through textural features. The proposed method applies an image processing technique to the spectrogram in order to capture 'acoustic patterns' which better reflect complex temporal structure at the sub-band level. To reduce computational and memory requirements, the new features are optionally used to learn a low-footprint codebook of the most significant patterns. The codebook provides a sparse representation of the acoustic structure. The research hypotheses are that: (i) recurrent acoustic patterns can be captured using local binary pattern (LBP) analysis [3] applied to the spectrogram; (ii) the new LBP-based features provide complementary information to traditional MFCC features, and that (iii) LBP can be applied as a 'bag-of-features' approach by creating a codebook of recurrent patterns and by the represention of each sample as combinations of these patterns. The paper validates these hypotheses and reports results which compete with the current state of the art. The remainder of this paper is organised as follows. Section 2 describes prior work with a focus on that relating to the capture of temporal recurrence and acoustic patterns. Section 3 presents the new contribution. Section 4 describes the implementation and assessment framework whereas Section 5 presents experimental results. Conclusions and directions for further work are presented in Section 6.

2. PRIOR WORK

Various approaches to ACR have been reported in the context of the public DCASE challenge literature [4] which illustrates the use of different features and classifiers. Though they are generally fused with different auxiliary features, the use of short-term cepstral coefficients is widespread, e.g. [5, 6, 7, 8]. While cepstral features are popular and even if successful in general, they stem from the analysis of speech signals and may thus be sub-optimal for ACR.

A study in [9] shows that humans utilise a priori knowledge of discrete acoustic events as cues to recognise context (i.e. engine sounds are more likely in car or bus contexts than in an office). Some successful approaches to ACR have accordingly explored the automatic detection of acoustic events, for instance using histograms of event-occurrences [10]. In a similar vein, the work

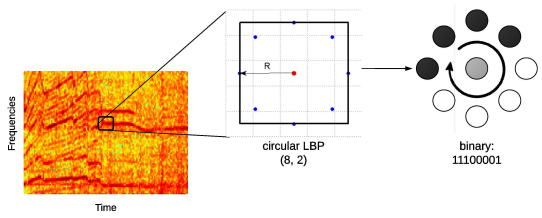


Figure 1: An illustration of LBP extraction using a spectrogram block. The centre the block is used to obtain the other values by interpolation. Starting from the upper-left, the LBP is obtained upon the binary comparison (Eq. 1) of outer values to the centre value. The LBP configuration is *circular*, with 8 neighbours and the radius equal to 2.

in [11] reports the use of a frame-based classifier which combines both short and long term features computed over 45ms and over 1.5s respectively. In another example reported in [12], sound events are learned through an unsupervised algorithm and used to characterize context.

Alternative approaches which capture recurrent acoustic patterns have also proven effective. One example involves audio motif discovery, which uses bio-informatics techniques which find recurrent patterns in genetic sequences. The work in [13] reports an approach which transforms an audio stream into a sequence of discrete states, each of them representing a specific audio pattern. A related approach to music genre classification which uses textural features is reported in [14]. Temporal recurrence, motif discovery and acoustic events share the notion of acoustic pattern analysis and lend support to the benefit of capturing longer-term information than is captured with conventional cepstral features.

3. LOCAL BINARY PATTERN CODEBOOKS

This paper reports the application to ACR of local binary pattern (LBP) analysis, a well known approach to feature extraction for automatic face recognition [15]. LBP is an efficient texture operator which labels the pixels of an image (here an audio spectrogram) by comparing their value to those of neighbouring pixels and by representing the result as a binary number. The general idea is illustrated in Fig. 1. The analysis of acoustic signals using LBP analysis has been reported previously [16] and is applied by treating the spectrogram as a visual representation of the acoustic signal, thereby resulting in [17].

The use of LBP for acoustic analysis and feature extraction is motivated by its suitability to texture and structure representation. LBPs are usually used to create histograms which capture recurrent structure. For ACR they provide more discriminative features which reflect the acoustic texture. The following describes the extraction of raw LBP features, henceforth referred to as LBP-Raw features, and an extension to a *bag-of-features* approach referred to as LBP-Codebook.

3.1. System overview

The new approaches are composed of four stages, as illustrated in Fig. 2:

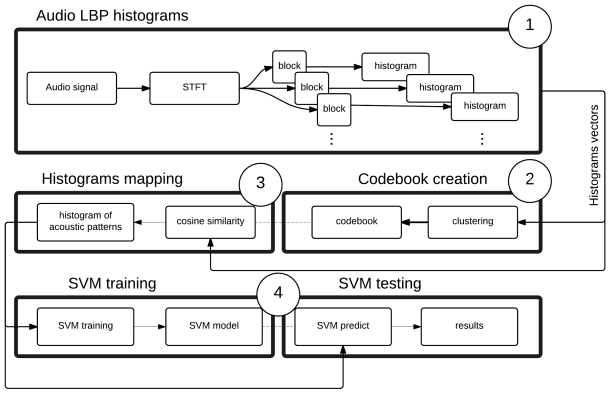


Figure 2: An illustration of the entire system, as explained in Section 3.1: **(1.)** LBP histogram generation for each sub-band; **(2.)** Codebook creation, through clustering; **(3.)** Histograms in (1.) are mapped to the codebook. This is repeated for each histogram extracted from each block; **(4.)** SVM training and testing by using the histogram of acoustic patterns.

1. LBP is applied to the spectrogram representation of the full acoustic signal by comparing the magnitude of each time-frequency 'bin' to those of its immediate neighbours. The set of raw LBPs are used to generate an LBP-Raw histogram which reflects the occurrence of each LBP across the full signal.
2. Histograms are generated for each signal in a large dataset and then clustered to group together the most similar histograms. Resulting clusters are then used to form a codebook.
3. The codebook can be used to map a histogram onto the single, nearest *word* as determined according to a cosine similarity metric. This process results in LBP-Codeword features of reduced dimension (and thus better suited to embedded applications) which are less redundant and less noisy.
4. ACR is performed using a support vector machine (SVM) classifier, applied either to LBP-Raw (1.) or LBP-Codebook (3.) features.

3.2. Local binary patterns

Various modifications to the spectrogram are generally necessary prior to LBP extraction, e.g. spectrum pre-processing techniques reported in [18]. Each bin in the spectrogram reflects the amount of energy present in proximity to specific time and frequency bins. This work shows that analysis of the linear-power spectrogram gives better results than the log-power spectrogram. In particular, bin values are scaled to values in the range 0-255 in order to mimic the application of LBP analysis in image processing:

$$LBP_{P,R} = \sum_{i=0}^{P-1} f(g_i - c)2^P, f(x) = \begin{cases} 1, x \geq c \\ 0, x < c \end{cases} \quad (1)$$

where g_i is the value of the i-th neighbour, c is the centre of the block and where P is the number of values or pixel count.

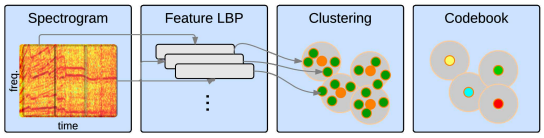


Figure 3: An illustration of codebook generation via k-means clustering.

R is the radius of the neighbourhood: the coordinates of g_i are $Rcos(2\pi i/P)$, $Rsin(2\pi i/P)$. We choose to use $P = 8$ and $R = 2$ in these experiments.

As highlighted in [19], LBP analysis is sensitive to rapid fluctuations, namely transitions in the LBP's code from 1 to 0 and vice-versa, which degrade performance. To remove such noise, while simultaneously reducing the dimension of the histogram, the work in [3] considered only LBPs for which the number of transitions between 0s and 1s is less than or equal to 2. This subset of LBPs represents the set of 58 so-called uniform patterns. The remaining non-uniform patterns are often grouped together and considered as a single, distinct pattern. In our current work the transition along the time (on the horizontal axis) and the frequencies (on the vertical axis) are considered in the same way of image processing. Nevertheless, we agree that transitions across time and frequencies should be treated differently. As future research, the shapes and the space of LBP patterns will be investigated more in details (see Sec. 6).

Uniform patterns typically represent textual elements, such as edges, corners or uniform areas. Noisy, non-uniform patterns are not useful for classification and are simply discarded in the work reported here, resulting in *2-transitions uniform* histograms. Such an approach has been shown previously to perform well also for the design of a voice activity detector [20].

3.3. Codebook creation

In order to reduce computation and memory requirements, codebooks can be used to reduce the dimension of the resulting feature vectors (and also context models). The principal idea is to extract automatically, via unsupervised k-means clustering, the most representative patterns for each context. The cosine distance is well suited as a distance metric for histogram features [21].

This method is based on the well-known *bag-of-features* (BoF) technique popular in image retrieval tasks [22]. The spectrogram of each test sample is represented in terms of the most relevant codebook patterns, as determined according to the same cosine similarity metric.

3.4. SVM classification

ACR is performed using a standard SVM classifier which projects raw data into an higher-dimensional space in which contexts may be linearly separable. This is achieved according to the hyperplane which maximises the margin between classes, thereby minimising classification errors [23].

4. EXPERIMENTS

This section describes the two datasets used for evaluation together with protocols, implementation specifics and metrics.

Context	Total time
Bus	8h56m
Car	3h40m
Office	13h10m
Subway	10h18m
Street	9h45m

Table 1: Duration of recordings for each context in the NXP Softare dataset.

4.1. Datasets

The proposed approach was tested on two different databases, essentially due to the modest size of the first, but standard database which does not afford sufficient statistical significance in exhaustive evaluations.

- **DCASE:** a public, standard dataset used in the past for competitive evaluations and nowadays for the comparison of different methods and algorithms [4]. It is composed of 100 stereo files of 30s duration, each with $f_s = 44.1$kHz. Only the first channel was used for all experiments reported here.
- **NXP Software:** a larger, but non-standard dataset containing 45 hours of recordings (see Table 1) with a sampling frequency of 16kHz, collected with multiple mobile devices. Recordings are manually annotated (context-labelled) before both the audio and the label are stored in a centralized system in the cloud. The dataset is representative of the real problem: it reflects context ambiguity at the user-level and is collected in multiple locations and with different acquisition configurations (i.e. microphones).

4.2. Protocols and metrics

For each dataset, a 5-fold partition was used to separate training and testing data. The codebook is learned on the training set. Except for LBP-Codebook (see later), all features are extracted from 8s of audio. The classifier is trained with the same 8s sub-clip features and then a majority voting is used to produce context decisions every 30s. Partitioning is performed at the file level, not the sub-clip level, to avoid overlap between sub-clips of the same file. The evaluation metric is context recognition accuracy, namely the percentage of trials for which the context is correctly recognised. All results are averaged across the 5-fold partitions.

4.3. Implementation

Baseline features are extracted from 8s audio sub-clips every 10ms using a frame length of 20ms and a bank of 40 Mel-scaled filters up to 900Hz, thereby resulting in 13 MFCCs for each frame. The mean and variance are then determined so that each sub-clip is parametrised with a single feature vector of 26 dimensions. RQA features are extracted according to the method reported in [2]. They capture recurrence in the baseline MFCC features over a period of 400ms but are averaged over the same 8-second sub-clip instead of 30s. Recurrent analysis needs longer time-window, while in real-time scenario the prediction has to be done within smaller sub-clips. This is the reason why the performances on DCASE have a drop from 71% to 62%.

LBP-Raw features are extracted from the acoustic signal after down-sampling to 16kHz. This is done in order to equalise the sampling frequencies of the two datasets. LBPs are extracted from the spectrogram which is first split into 3 sub-bands (900Hz, 2kHz, 8kHz), with the aim of distinguish between similar patterns but coming from different sub-bands of the spectrum.

Histograms are extracted separately for each sub-band and concatenated to form a single feature vector. The resulting histogram is normalized by dividing each bin value by the total block count.

LBP-Codebook features stem from LBP analysis applied to smaller 1-second sub-clips. Clustering is applied to obtain 30 clusters for the DCASE dataset and 100 for the larger NXP Software dataset. Through other experiments, these were found to be optimal given the two, different dataset sizes. LBP-Codebook features extracted from each sub-clip are aggregated over 30s to obtain a single BoF histogram per audio sample.

The SVM classifier is implemented with the well known LibSVM tool-kit, more details of which can be found in [24]. All experiments were performed with a radial basis function kernel and with C and γ parameters optimised through a grid search. A multi-class SVM (for multiple contexts) is learned with all pair-wise combinations. SVM scores are z-score normalised, as derived from the training set and applied to the test set.

5. RESULTS

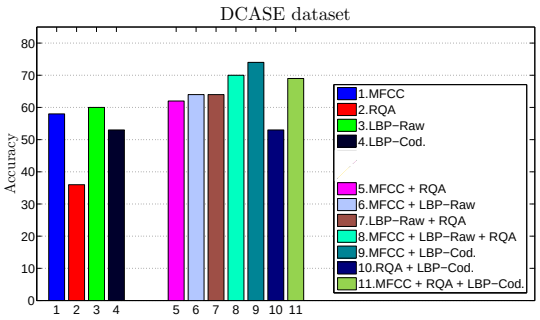
Reported here are experimental results for both databases and multiple feature configurations involving MFCC, RQA, LBP-Raw, LBP-Codebook features and their combinations. MFCC and RQA features form two baselines.

5.1. DCASE dataset

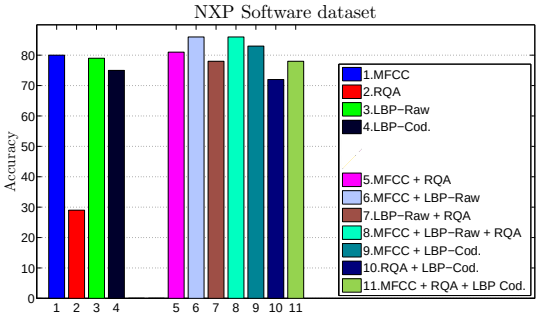
Results for the DCASE dataset are illustrated in Fig. 4(a). With recognition accuracies in the order of 60%, they show that MFCC and LBP-Raw features are the best performing single feature sets. While RQA features on their own perform less well, they are complementary to MFCCs; performance improves with fusion. Better performance is observed when MFCCs are combined with LBP-Raw features. The combination of MFCCs, RQA and LBP-Raw features improves performance further to 70%. While as a single feature set, LBP-Codebook features give worse performance than LBP-Raw, they are the most complementary to MFCCs; when combined, recognition accuracy increases to almost 75%.

5.2. NXP Software dataset

Results for the NXP Software dataset are illustrated in Fig. 4(b). Similar performance trends are observed; MFCC and LBP-Raw features are the best performing single feature sets while RQA and LBP-Codebook features perform less well. RQA, LBP-Raw and LBP-Codebook are still complementary to MFCCs: RQA brings an improvement of 6%, while LBP-Raw and LBP-Codebook deliver improvements in the order of 6% and 3% respectively. Further analysis confirms that these improvements are statistically significant. With a baseline performance of 80%, performance for the NXP Software dataset follows the trend as seen in DCASE: the best feature combinations (bars 6 and 8 in Fig. 4(b)) both involve LBP features. Even if LBP-Codebook features give worse results than LBP-Raw, the former are computationally more efficient.



(a)



(b)

Figure 4: Recognition accuracy averaged over 5-fold partitions for (a) DCASE and (b) NXP Software datasets, obtained with different configurations of MFCC, RQA and LBP features. Two groups of results illustrate performance for single features (left) and fused features (right).

6. CONCLUSIONS

This paper proposes new, promising approaches to feature extraction for acoustic context recognition. Local binary patterns (LBPs) aim to capture the distribution of audio structure and are complementary to conventional Mel-scaled cepstra. Their combination completes that with recurrent quantification analysis and betters the current state of the art, adding further weight to the benefit of capturing textural features and complex acoustic structure. In addition, a bag-of-features approach is shown to reduce feature dimensionality while still improving on baseline performance. With reduced computational complexity, the codebook approach is perhaps better suited to embedded applications. Further work should investigate different LBP shapes, in particular rectangular instead of circular block configurations, in order to optimise the time-frequency resolution. In the second instance, the codebook could be trained using a larger pool of readily available data in order to recognise distinct acoustic events rather than abstract time-frequency patterns. This approach may facilitate the learning of codebooks for distinct events, e.g. car horns, or an engine) which may be beneficial, especially if these events are learned in a discriminative framework tailored to the context recognition task.

7. REFERENCES

[1] A. Zimmermann, A. Lorenz, and M. Specht, "Applications of a context-management system," in *Modeling and Using Context*, ser. Lecture Notes in Computer Science, A. Dey, B. Kokinov, D. Leake, and R. Turner, Eds. Springer Berlin Heidelberg, 2005, vol. 3554, pp. 556–569.

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

[3] T. Ojala, M. Pietikinen, and T. Menp, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transacrtions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[4] D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M. Plumbley, "Detection and classification of acoustic scenes and events," Queen Mary, University of London, Tech. Rep., March 2013.

[5] D. Li, J. Tam, and D. Toub, "Auditory scene classification using machine learning techniques," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.

[6] J. Geiger, B. Schuller, and G. Rigoll, "Recognizing acoustic scenes with large-scale audio feature extraction and svm," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.

[7] W. Nogueira, G. Roma, and P. Herrera, "Identification based on mfcc, binaural features and a support vector machine classifier," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.

[8] B. Elizade, G. Lei, H. Friedland, and N. Peters, "An i-vector based approach for audio scene detection," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.

[9] V. T. K. Peltonen, A. J. Eronen, M. P. Parviainen, and A. P. Klapuri, "Recognition of everyday auditory scenes: Potentials, latencies and cues," in *In Proc. Audio Eng. Soc. Convention*. Hall, 2001.

[10] T. Heittola, A. Mesaros, A. Eronen, and T. Virtanen, "Context-dependent sound event detection," *EURASIP Journal on Audio, Speech, and Music Processing*, no. 1, 2013.

[11] M. Chum, A. Habshush, and C. Rahman, A. Sang, "Scene classification challenge using hidden markov models and frame based classification," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.

[12] H. Lu, W. Pan, N. Lane, T. Choudhury, and A. T. Campbell, "Soundsense: Scalable sound sensing for people-centric application on mobile phones," *MobiSys'09*, 2009.

[13] J. Burred, "Genetic motif discovery applied to audio analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 361–364.

[14] Y. Costa, L. Oliveira, A. Koerich, and F. Gouyon, "Music genre recognition using gabor filters and lpq texture descriptors," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, ser. Lecture Notes in Computer Science, J. Ruiz-Shulcloper and G. Sanniti di Baja, Eds. Springer Berlin Heidelberg, 2013, vol. 8259, pp. 67–74.

[15] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 2518–2525.

[16] N. Chatlani and J. Soraghan, "Local binary patterns for 1-d signal processing," *18th European Signal Processing Conference (EUSIPCO)*, pp. 95–99, 2010.

[17] F. Alegre, A. Amehraye, and N. Evans, "A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns," in *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Sept 2013, pp. 1–8.

[18] J. W. Dennis, "Sound event recognition in unstructured environments using spectrogram image processing," 2014.

[19] T. Kobayashi and J. Ye, "Acoustic feature extraction by statistics based local binary pattern for environmental sound classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 3052–3056.

[20] Q. Zhu and J. Soraghan, "Lbp based recursive averaging for babble noise reduction applied to automatic speech recognition," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*, Sept 2014, pp. 1267–1271.

[21] J. Choi, H. Cho, J. Kwac, and L. S. Davis, "Toward sparse coding on cosine distance," in *International Conference on Pattern Recognition*, 2014.

[22] X. Yuan, J. Yu, Z. Qin, and T. Wan, "A sift-lbp image retrieval model based on bag of features," in *IEEE International Conference on Image Processing*, 2011.

[23] V. N. Vapnik, "The nature of statistical learning theory," 1995.

[24] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.