Bag-of-Features Methods for Acoustic Event Detection and Classification

René Grzeszick, *Student Member, IEEE*, Axel Plinge, *Member, IEEE*, and Gernot A. Fink, *Senior Member, IEEE*

***Abstract*—The detection and classification of acoustic events in various environments is an important task. Its applications range from multimedia analysis to surveillance of humans or even animal life. Several of these tasks require the capability of online processing. Besides many approaches that tackle the task of acoustic event detection, methods that are based on the well known bag-of-features principle also emerged into the field. Acoustic features are calculated for all frames in a given time window. Then, applying the bag-of-features concept, these features are quantized with respect to a learned codebook and a histogram representation is computed. Bag-of-features approaches are particularly interesting for online processing as they have a low computational cost. In this paper, the bag-of-features principle and various extensions are reviewed, including soft quantization, supervised codebook learning, and temporal modeling. Furthermore, Mel and Gammatone frequency cepstral coefficients that originate from psychoacoustic models are used as the underlying feature set for the bag-of-features. The possibility of fusing the results of multiple channels in order to improve the robustness is shown. Two databases are used for the experiments: The DCASE 2013 office live dataset and the ITC-IRST multichannel dataset.**

***Index Terms*—Acoustic event detection, bag-of-features.**

I. Introduction

IN ORDER to allow machines to obtain a better understanding of what is happening in an acoustic scene, sound events have to be identified. The task of acoustic event detection (AED) addresses both the localization in time and the classification of these events. AED is of importance for a large variety of practical applications, spanning outdoors and indoors, offline and online, single and multichannel applications. Offline applications include the analysis of multimedia content. Here, AED is an important building block amongst the recognition of objects or visual actions and movements [1]. The combination of these methods can be used for the understanding of high level semantic events in videos [2]. Outdoor AED is used in urban planning or the analysis of possible noise complaints and human outdoor activities [3]. Further outdoor applications involve mobile robots which are used for security [4] or wildlife investigation [5]. Surveillance in cluttered scenes can be improved by the acoustic detection of unexpected scenarios which are not visually recognizable (e.g. screams or glass breaking) [6], [7]. Indoor online AED is a widely investigated field [8]. It can be used for online analysis and annotation of meetings or lectures [9]. AED can also improve the robustness of other tasks in real world applications, e.g., speaker tracking or controlling a beamformer for speech enhancement [10], [11]. Especially in conference room and domestic settings, multi-channel systems are used. Multichannel fusion has been shown to improve the performance of speech and event recognition [12], [13]. Moreover, the performance of indoor scene analysis can be improved by incorporating spatial information [14].

In order to address the task of AED, a large variety of methods have been proposed. This paper focuses on the well known Bag-of-Features (BoF) principle. The application to the task of indoor online detection and classification of acoustic events is investigated. The work presented in this paper mainly builds on three previous works [15]–[17], but also discusses the development of different BoF approaches in the field (cf. [1], [18]). Furthermore, various extensions to the previous work are also discussed along with a comparison of different design decisions within the BoF framework: Namely, the usage of different feature sets, including MFCCs, biologically inspired features (GFCCs), loudness and the perceptual feature set discussed in [19]. An analysis of different codebook learning techniques such as hard quantization, soft quantization and supervised codebook learning is given. A comparative evaluation is performed on a single channel as well as a multichannel dataset. It will be shown that BoF approaches show state-of-the-art results while having a low computational cost compared to other methods that are capable of online processing.[1]

The remainder of this paper is organized as follows. The next section discusses the related work and various methods approaching the task of AED. The third section discusses the BoF principle in the context of the classification and detection of acoustic events. This includes various extensions to the plain BoF processing pipeline. The fourth section presents experiments for AED on two datasets, the DCASE 2013 office live development set and the ITC-IRST dataset from CLEAR. Evaluation metrics, different design decisions and parameters for the BoF principle will be discussed. Furthermore, a comparison with state-of-the-art approaches for online AED will be given.

Manuscript received June 28, 2016; revised November 10, 2016 and January 18, 2017; accepted January 25, 2017. Date of current version May 23, 2017. This work was supported by the German Research Foundation (DFG) under Projects FI799/5-1 and FI799/9-1. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Tuomas Virtanen. *(Corresponding author: René Grzeszick.)*

The authors are with the Department of Computer Science, TU Dortmund University, Dortmund 44227, Germany (e-mail: rene.grzeszick@tu-dortmund.de; axel.plinge@tu-dortmund.de; gernot.fink@tu-dortmund.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2017.2690574

[1] A video demonstration can be found on https://vimeo.com/95845132

The paper concludes with a short discussion and summary of the influence of BoF methods in the field of AED.

II. Approaches to Acoustic Event Detection

A large variety of approaches have been proposed to address the task of AED. Prominent competitive comparisons include the "Classification of Events, Activities and Relationships" (CLEAR) [8] and the "Detection and Classification of Acoustic Scenes and Events" (DCASE 2013 and 2016) [20], [21].

For offline applications, Hidden Markov models (HMMs) are widely employed. Their main advantage is the inherent modeling of dynamic temporal alignment when using Viterbi decoding [22], which requires the full sequence as input. There are several architectures employed for AED [23], [24]. The general architecture uses a fixed number of states for each event in a sub-model, and then connects them as parallel alternatives. The detected events are then determined by the Viterbi path passing through the states of the corresponding sub-model. In [12], different multichannel fusion strategies are shown to improve the results. The output modelling is commonly achieved with Gaussian mixture models (GMMs) for mel frequency cepstral coefficients (MFCCs). As in speech recognition, the use of deep neural networks (DNNs) based on spectral features for output modeling was also introduced for AED [25]. In [26], promising results are achieved using bi-directional long-term short-term memory networks for the analysis of sequences. Here, a binary output encoding enables the detections of overlapping events.

Methods for online classification and detection of acoustic events are typically applied over short time windows and combined with a sliding window approach in order to allow for an online analysis. Different classifiers have been applied using various feature representations.

The basic method is to use MFCC features and a GMM to model each acoustic event class, similar to the approaches in speaker identification [27]. The mean and variance of the feature vectors are modeled by the Gaussian mixtures. Typically, GMMs are trained separately for each class. For classification the GMMs' estimates are summed over all frames and the class with the highest likelihood is chosen [28], [29]. Since the summation discards any temporal structure, the method is sometimes termed 'Bag-of-Frames' [28]. More sophisticated GMM methods include the use of a background model [30].

Besides GMMs, several approaches incorporate random forests for AED. For example, the classification of statistics over all feature vectors in the time window [3], or frame-by-frame recognition [31]. In [32] a two-stage approach is applied. Each frame is classified with respect to the classes in question. A histogram over a given number of pre-classified frames is computed, which is then referred to as the *superframe*. This representation is combined with a regression forest that is trained for each class in order to determine the temporal extent of the event. Thresholding is used in order to determine the onset and offset based on the forest's output probabilities. This also allows to predict the offset of a class in the future [33]. In [13] it has been shown that multichannel fusion improves the results in conference room settings.

The recently popular DNNs and convolutional neural networks (CNNs) can also be applied to AED based on a sliding window. Often spectra or mel energies are used as features, allowing the network to learn the feature representation. In [34] a DNN is trained based on a binary multilabel encoding for acoustic events, which enables the recognition of multiple overlapping events. The same is achieved in [35] by using a threshold on a sigmoid output layer. In contrast to DNNs, CNNs apply convolutions to the input data and, therefore, make use of weight sharing, reducing the number of parameters in the network. In [36], spectogram image features (SIF) are computed and then the CNN is applied. It could be shown that the features derived from the network are very robust against noise. Although approaches based on DNNs and CNNs were able to improve the results in several tasks, their performance in real life applications is not always superior, especially in cases of limited training material [21].

Besides neural networks, methods that build on the Bag-of-Features (BoF) principle have emerged into the field of AED. The BoF approach originated in text retrieval (cf. [37]) and has successfully been applied to various pattern recognition applications over the years. Acoustic features are extracted for each frame in the training set. These features are clustered in order to build a set of representatives. The occurrences of these representatives within a given window are counted and a histogram is derived from the counts. The histogram is then used for classification. For the task of AED, these representatives are often referred to as an audio or acoustic word [1], [15], [18]. As the temporal information is discarded, feature augmentation approaches have been introduced in order to incorporate coarse temporal information [16]. Similar to [12], [13], the combination of multiple channels is evaluated in order to improve the robustness and ultimately the detection performance. Beyond heuristic combination strategies, a novel method based on classifier stacking was introduced in [17].

In the following the BoF approach will be discussed in detail and its application to the task of online AED will be evaluated. The BoF principle will also be compared with other approaches addressing this task.

III. Bag-of-Features Acoustic Event Detection

The BoF approach for classifying and detecting acoustic events processes short time windows of w seconds of a single microphone signal. For a given time window n, a set of feature vectors $\boldsymbol{Y}_n = (\boldsymbol{y}_1 \dots \boldsymbol{y}_K)$ is derived, where K is the number of frames in the window. Then, according to the BoF principle, the following steps are performed:

1) A set of representatives is computed by clustering the feature vectors of all frames in the training data. Typical methods include k-Means clustering or GMMs.
2) The features within one window are assigned to one cluster (hard assignment) or weights for each cluster are computed (soft assignment). These weights are then added

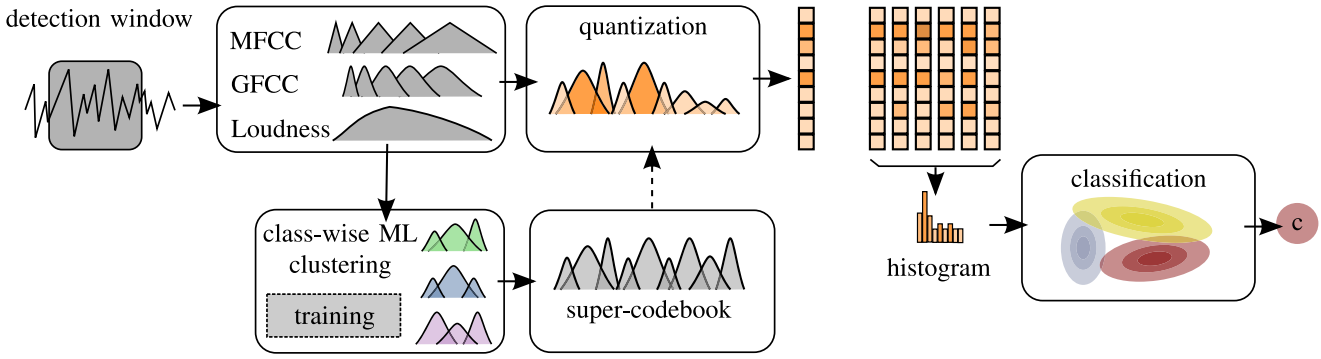

Fig. 1. Overview of the general processing pipeline of BoF methods for acoustic event detection. The input signal is processed with a sliding window approach. On this detection window, features are computed. These are then quantized using a codebook. A histogram over the occurrences of the codebook entries is computed. The histogram is then classified in one of the event classes.

up resulting in a histogram representation, the Bag-of-Features.

3) The histogram representations are then used for classification, e.g., using an SVM, a Random Forest or Maximum Likelihood classification.

An overview of this approach is given in Fig. 1. In combination with a sliding window approach, these steps can be used for detection tasks. In the following section the steps will be reviewed in detail: First, the underlying feature representation will be explained. Then different methods for computing a BoF representation using either hard or soft assignments in either an unsupervised or a supervised manner will be discussed. Furthermore, different approaches for re-introducing temporal information within a BoF representation will be shown. Finally, methods for classification and detection and an extension to multiple channels will be explained.

A. Features

For sound and especially speech processing, a variety of technical features are used. One well known collection of features is the perceptual feature set proposed in [19] which incorporates short time energy, sub-band energies spectral flux, zerocrossing rate and pitch. The most commonly used feature representation are MFCCs [38]. The input signal is split into multiple bands by a mel frequency filter bank and the discrete cosine transform (DCT) is computed on the logarithm of their magnitudes. Here, 13 MFCC coefficients starting at the second one are used.

The long history of psychoacoustic research has been complemented by computational modeling of the human hearing process. In these models, ERB-spaced gammatone filterbanks [39] are used from which the gammatone frequency cepstral coefficients (GFCCs) were derived [40]. The implementation used replaces the mel frequency filterbank of the MFCCs with linear phase gammatone filters. The filters are defined in the spectral domain using a gammatone approximation [41] with

$$G^{(b)}(f) = (1 + j(f - f_b)/w_b)^{-4}, \tag{1}$$

where $G^{(b)}$ is the bth gammatone filter with the center frequency f_b and Glasberg Moore bandwidth w_b. Here, j denotes the imaginary unit. Again 13 coefficients starting at the second one are used.

In addition the feature set incorporates the perceptual loudness which is derived from the A-weighted magnitude spectrum. Hence, the final feature vectors are comprised of regular MFCCs $\boldsymbol{m}$, GFCCs $\boldsymbol{g}$ and a loudness component o so that for the kth frame the feature vector is

$$\boldsymbol{y}_k = (m_1, \ldots, m_{13}, g_1, \ldots, g_{13}, o)^T \ . \tag{2}$$

The mean and standard deviation of the training data are computed and a whitening is performed so that the features heave zero-mean and unit variance.

B. Bag-of-Features

The key idea of the BoF principle is to learn an intermediate representation from the features in an unsupervised manner. This is usually done by deriving a set of representatives, also referred to as the *codebook*. Typically, clustering algorithms that solve the k-Means problem, like Lloyd's clustering (used in [1], [15]) or spherical k-Means (used in [18]), are applied. The clustering then results in a hard assignment of features to the respective centroids. For a soft assignment, GMMs can also be used for computing the codebook.

B. Hard Quantization: In hard quantization, clustering is applied to all $N \cdot K$ feature vectors $\boldsymbol{y}_{1:K \cdot N} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K \cdot N})$ where N denotes the number of windows $\boldsymbol{Y}_n$ in the training and K the number of frames within a given window. From these a sequence of L representative vectors $\boldsymbol{\mu}_{1:L}$ is derived. Here, Lloyd's widely popular k-Means algorithm is used for clustering (cf. [1], [15]). For hard quantization only the cluster centroids are of interest and the assignment is based on the minimal distance:

$$b_l(\boldsymbol{Y}_n, \boldsymbol{\mu}_l) = \frac{1}{K} \sum_k \delta(\boldsymbol{y}_k, \boldsymbol{\mu}_l) \tag{3}$$

with

$$\delta(\boldsymbol{y}_k, \boldsymbol{\mu}_l) = \begin{cases} 1 & \text{if} \quad \underset{i}{\operatorname{argmin}}\, d(\boldsymbol{y}_k, \boldsymbol{\mu}_i) = l \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where d defines a distance function, i.e., the Euclidean distance.

Soft Quantization: For soft quantization, typically GMMs are used. The mixture model is fitted to the data by applying the expectation-maximization (EM) algorithm to all $N \cdot K$ feature

vectors $\boldsymbol{y}_{1:N\cdot K}$. Hence, a codebook $\boldsymbol{V}$ is computed where the lth entry is defined by a tuple of mean and deviation:

$$\boldsymbol{V}_l = (\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l). \tag{5}$$

Based on this codebook, each feature vector $\boldsymbol{y}_k$ is softly quantized by

$$q_{k,l}(\boldsymbol{y}_k, \boldsymbol{V}_l) = \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l) / \sum_{l'} \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{\mu}_{l'}, \boldsymbol{\Sigma}_{l'}). \tag{6}$$

Quantizing all K frames in a short time window, a histogram $\boldsymbol{b}$ is computed that represents this window. The lth histogram entry is then given by

$$b_l(\boldsymbol{Y}_n, \boldsymbol{V}_l) = \frac{1}{K} \sum_k q_{k,l}(\boldsymbol{y}_k, \boldsymbol{V}_l). \tag{7}$$

Supervised Quantization: The introduced hard and soft quantization compute the BoF representation in an unsupervised manner. However, disregarding the class information in the clustering may mitigate significant differences. Supervised clustering provides a remedy for this effect. Codebooks of size I are computed for all C classes Ω_c separately as in the GMM based Bag-of-Frames approach. In order to integrate this in the BoF representation, the codebooks are concatenated into a large super-codebook [15]. The EM algorithm is therefore applied separately to all feature vectors $\boldsymbol{y}_k$ of each class Ω_c and thus I means and deviations $\boldsymbol{\mu}_{1:I,1:C}, \boldsymbol{\Sigma}_{1:I,1:C}$ are estimated for all C classes. The super-codebook $\boldsymbol{V}$ is then the concatenation of these means and deviations so that it consists of $L = I \cdot C$ elements, with the l^{th} element being

$$\boldsymbol{V}_{l=(I\cdot c+i)} = (\boldsymbol{\mu}_{i,c}, \boldsymbol{\Sigma}_{i,c}) \quad . \tag{8}$$

Here, the index l is computed from the class index c and the Gaussian index i as $l = I \cdot c + i$. The quantization can then be computed as for the soft quantization based on the concatenated codebook. In analogy to the super-vector method, which is used in speaker identification (cf. [42]), this method is referred to as *Bag-of-Super-Features*.

C. Temporal Information

Since a BoF is an orderless representation, it does not contain any temporal information. The temporal structure within the frame $\boldsymbol{Y}_n$ is therefore lost. However, the temporal structure may be an important cue of information for distinguishing different acoustic events. There are several methods which address this issue by re-introducing temporal information in the short time windows that are processed by the BoF approach. One approach is subdividing the time windows which is proposed by the pyramid scheme [15], [43]. The alternative is applying feature augmentation which directly encodes the temporal information at feature level [44], [45].

C. Temporal Pyramid: The idea of the pyramid scheme is to subdivide the window after the quantization in a hierarchical manner and to build a BoF histogram for each of the tiles. The histograms of all tiles are then concatenated into a single vector representation before classification. For AED this approach can be directly applied to the detection window by subdividing it in a temporal manner [15]. An illustration is given in Fig. 2. For acoustic events, a shallow two level pyramid has been proposed (cf. [15], [16]). On the bottom level T tiles are computed and only the frames inside each tile are used for computing the histogram. For a feature vector of the nth window the sub-histogram of the jth tile is computed by:

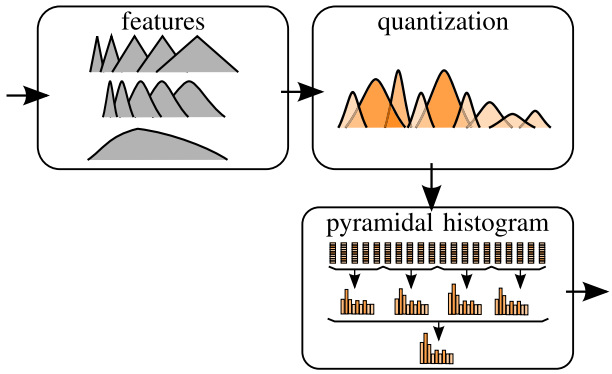

Fig. 2. Overview of the temporal pyramid approach. The window is subdivided after the quantization. A max pooling step is applied for the top level of the pyramid.

$$b_l^{(j)}(\boldsymbol{Y}_n, \boldsymbol{V}_l) = \frac{1}{K/T} \sum_{k=jK/T}^{(j+1)K/T} q_{k,l}(\boldsymbol{y}_k, \boldsymbol{V}_l). \tag{9}$$

While the original pyramid representation [43] sums up the values of all tiles, here, max pooling is applied for computing the top level histogram. This represents the maximum activation within any of the bottom level tiles by computing

$$b_l^{(max)}(\boldsymbol{Y}_n, \boldsymbol{V}_l) = \max\left\{b_l^{(1)}(\boldsymbol{Y}_n, \boldsymbol{V}_l), ..., b_l^{(T)}(\boldsymbol{Y}_n, \boldsymbol{V}_l)\right\}. \tag{10}$$

Finally, all histograms are concatenated into a single feature vector that represents the complete window:

$$\boldsymbol{b}(\boldsymbol{Y}_n, \boldsymbol{V}) = \left(\boldsymbol{b}^{(1)}(\boldsymbol{Y}_n, \boldsymbol{V}), \ldots, \boldsymbol{b}^{(T)}(\boldsymbol{Y}_n, \boldsymbol{V}), \boldsymbol{b}^{(max)}(\boldsymbol{Y}_n, \boldsymbol{V})\right) \tag{11}$$

C. Feature Augmentation: The alternative to the popular pyramid approach is feature augmentation. Here, the temporal information is directly encoded at feature level [16], as shown in Fig. 3. Therefore, each feature vector is extended by additional quantized time coordinates t. Similar to the pyramid, the window is subdivided into T tiles of equal size. Here, the time difference to the beginning of the time window is quantized into a value of $[1, .., T]$. Thus an augmented feature vector consists of MFCCs $\boldsymbol{m}$, GFCCs $\boldsymbol{g}$, the loudness o and the temporal index t so that the kth feature vector is given by:

$$\boldsymbol{y}_k = (m_1, \ldots, m_{13}, g_1, \ldots, g_{13}, o, t)^T \tag{12}$$

When processing the augmented features by any of the quantization approaches described in Section III-B, this generates codebook entries which are specific for each of the temporal tiles. Note that this is a major difference to the pyramid approach in which the same global codebook is applied for each of the tiles. Furthermore, the size of the codebook L is equal to the

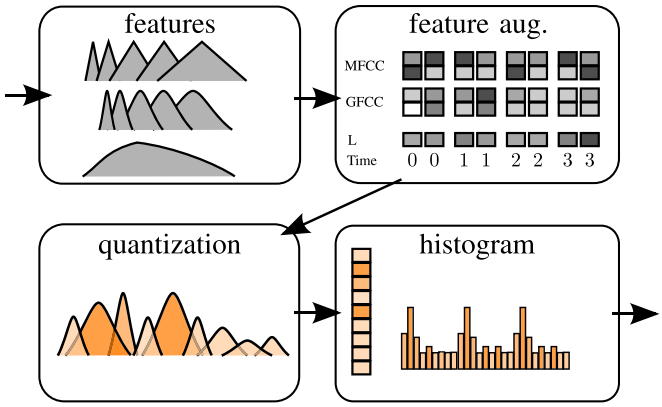

Fig. 3. Overview of the temporal feature augmentation. Temporal information is appended at feature level so that it is included in the quantization process and only one histogram is computed.

size of the overall feature representation. In the pyramid scheme the size of the overall representation grows with the number of tiles (cf. [16], [43]) which may introduce redundancies into the final representation.

D. Classification

The histograms that are computed from a BoF method can be used as a feature representation Y_n of a given window and as an input for a classifier. Typical methods include SVMs, random forests (RF), k-nearest neighbours (KNN) or maximum likelihood (ML) classification [46].

SVMs: A typical classfication approach is using multiclass SVMs with a suitable kernel function, e.g., linear or radial basis functions (RBF). For the task of acoustic event detection the histogram intersection kernel was shown to work very well [1], [43]. Given two histograms $\boldsymbol{a}, \boldsymbol{b}$ it computes the inner product of these as:

$$k_{\mathrm{HI}}(\boldsymbol{a}, \boldsymbol{b}) = \sum_{l=1}^{L} \min\{a_l, b_l\} \,. \tag{13}$$

Random Forests: A random forest consists of a set of decision trees that perform a binary decision based on a single feature at each node. Based on an optimization criterion a variable index and a threshold is chosen for each node. Each decision tree is trained only on a subset of the data. Hence, the forests are known to generalize well. They are successfully applied to AED in [32].

Maximum Likelihood: For the maximum likelihood classification, the probability $P(\boldsymbol{V}_l|\Omega_c)$ of an acoustic word to occur for a given class is evaluated. This probability is estimated based on the training samples $\boldsymbol{Y}_n \in \Omega_c$ for each class c with

$$P(\boldsymbol{V}_l|\Omega_c) = \frac{\alpha + \sum_{\boldsymbol{Y}_n \in \Omega_c} b_l(\boldsymbol{Y}_n, \boldsymbol{V}_l)}{\alpha L + \sum_{u=1}^{L} \sum_{\boldsymbol{Y}_n \in \Omega_c} b_u(\boldsymbol{Y}_n, \boldsymbol{V}_u)} \quad , \tag{14}$$

where α is a weighting factor for smoothing. In practice, Lidstone smoothing with $\alpha = 0.5$ shows good results. The maximum likelihood classification assumes that all classes are equally likely and therefore have the same prior. Based on this assumption, the posterior is estimated by

$$P(\boldsymbol{Y}_n|\Omega_c) = \prod_{\boldsymbol{V}_l \in \boldsymbol{V}} P(\boldsymbol{V}_l|\Omega_c)^{b_l(\boldsymbol{Y}_n, \boldsymbol{V}_l)}, \tag{15}$$

using the weighted product of the relative frequencies of all acoustic words in the codebook.

E. Detection

The simplicity of the BoF approach allows for a rapid computation and allows to easily adapt it to the task of AED. For this task, a sequence of various acoustic events has to be evaluated. The classification window is therefore applied in a sliding window approach. A fine grained analysis is performed by moving the window forward for one frame at a time. The classification result is applied to the frame that is centered in the window. Hence, a short context information is available for the time before and after the respective frame. Considering that the time window has a length of w seconds, there is a processing delay of $w/2$ seconds. The experiments will shown that a delay of only 0.3 seconds is sufficient for practical purposes.

Given that the frame shift may be very small and the results can be varying in some cases, the overall detection result of a sliding window approach may yield noisy labels. However, this can be overcome by post-processing. For example, in [15] the output is smoothed by majority voting over the results from one second. In [47] the segments which are computed from a first pass over the data are post-processed by an additional classification step in order to remove false positives. As the focus of the evaluation is frame-based and concerned with online AED, these post-processing steps will not be considered here.

F. Multi Channel Fusion

Given a sensor network containing R microphones, the presented BoF approaches can be evaluated for each microphone r individually. The detections of all channels are then combined in order to obtain a more robust result. This assumes that the channels are synchronized at least at frame level.

Traditional fusion techniques would combine the classification results from different microphones for each frame based on a heuristic such as majority voting, the maximum rule or the product rule (cf. [12]). Here, a different approach based on classifier stacking is discussed (cf. [17]). Given the posterior probabilities from all microphones in the sensor network, a second classifier is trained that uses these as input features. The final class label is then predicted by the learned classification function $\mathcal{F}$:

$$\hat{c} = \mathcal{F}((P_r(\boldsymbol{Y}_n|\Omega_c))_{(c,r)}) \,. \tag{16}$$

Here, a Random Forest classifier is trained on the posterior probabilities of all single-channel evaluations. Note that either additional training material is required for training the classifier stacking or the existing training material must be split into two disjoint sets: one part that is used for training the single-channel BoF models and classifiers and the other part is used for training the classifier stacking.

TABLE I
OVERVIEW OF THE DCASE 2013 OFFICE LIVE DATASET

	training time [s]		test time [s]	
class	total	event avg.	total	event avg.
alert	40.96	1.37	11.25	1.07
clearthroat	24.95	0.83	10.07	0.84
cough	26.11	0.87	8.44	0.94
doorslam	20.37	0.68	5.36	0.89
drawer	35.41	1.18	8.57	1.14
keyboard	101.44	3.38	15.48	1.41
keys	48.68	1.62	17.22	1.43
knock	23.95	0.80	6.40	0.71
laughter	35.69	1.19	5.69	0.76
mouse	28.48	0.95	7.41	0.93
pageturn	67.89	2.26	15.03	1.37
pendrop	14.06	0.47	4.64	0.52
phone	266.53	8.88	14.26	1.36
printer	332.04	11.07	11.37	2.53
speech	82.04	2.73	12.75	1.21
switch	6.46	0.22	2.84	0.24

Durations are reported as the average over both annotations.

IV. EVALUATION

In the following the BoF principle is evaluated on different AED datasets. A detailed evaluation is given with respect to different parameters and design decisions influencing the BoF. All parameters and design decisions are evaluated on the DCASE 2013 office live development set. Furthermore, a comparison with different methods from the literature is given. In addition, the ITC-IRST dataset is used for evaluation as it provides a more complex setting with 32 microphones and multiple recording sessions and allows for demonstrating the multichannel capabilities of the presented approach.

Metrics: As the proposed method generates a fine grained analysis with one classification result per frame, the evaluation is done in a frame-wise manner. The alternative event-wise measure is biased if a small number of frames is incorrectly classified. In the worst case, a single frame may form an incorrectly recognized event. Based on the relatively low number of events in most test sets (cf. Table I) such cases are penalized too heavily.

Using a sampling rate of 44.1 kHz and 1024 samples per frame, a frame contains 23.2 ms. Each frame has a 50% overlap with its neighboring frame and, therefore, it is used for classifying 11.6 ms. When denoting g, e, and t as the number of ground truth, estimated and correct frames, precision P and recall R can be defined along with the F-score F as in [28]

$$P = \frac{t}{e}, \quad R = \frac{t}{g}, \quad F = \frac{2PR}{P+R}. \tag{17}$$

For the assessment of the event detection performance, the non-event class Ω_0 is excluded from the counts. The acoustic error rate is computed as the number of deletions d, insertions i and substitutions s divided by the total number of non-background event frames:

$$E = \frac{d+s+i}{g}. \tag{18}$$

As this error rate is similar to the Acoustic Event Error Rate (AEER) [8], which is quite frequently computed in an event-wise manner, it is denoted as the Acoustic Frame Error Rate (AFER) in the evaluation.[2]

Significance: A randomization test (with $N = 10^5$) was performed in order to test for significance when comparing different classifiers and feature configurations. This method was chosen since it does not make any assumptions about the distribution of the data.

A. DCASE 2013 office live Dataset

The DCASE 2013 office live dataset is comprised of a variety of indoor sounds. All sounds for this task are typical for office or meeting room scenarios. There are 16 sound classes *alert, clearthroat, cough, doorslam, drawer, keyboard, knock, laughter, mouse, pageturn, pendrop, phone, printer, speech, switch, keys* that have to be detected and a background class which is comprised of *silence*. The dataset provides a training set of segmented sequences for each of the 16 classes. The total length of all segments is 18 minutes and 49 seconds. An overview of the amount of data for each of the classes is given in Table I. There is noticeable variability in the event lengths. However, most of them tend to be shorter than two seconds. Furthermore, there are three scripted, publicly available test sequences. The task is to detect and classify the acoustic events in these sequences correctly. The sequences have a total length of 5 minutes and 21 seconds and for each of these sequences two annotation sets are available. Since the training segments offer no training data for the silence/background class, the silence class has been trained based on the silence portions from the other two scripts when evaluating each script.

All experiments were repeated 10 times for all sequences and annotations. Each time a new codebook has been computed. The results are reported as the average over all three scripts, both annotations and the 10 codebook generations. Note that the F-Score is computed for each script and annotation separately and that the differences in the scripts lead to a larger variance as they differ in their difficulty (on average the scripts deviate around 3% in the F-score).

An exemplary result is shown in Fig. 4. It can be seen that the BoF approach covers most of the ground truth events quite well. However, the predicted labels tend to be slightly noisy at some points in time. Most surprisingly, the onset for a few events appears to happen before the event starts according to the ground truth annotations. This can, for example, be caused by background noises which are not part of the annotation.

Features & Encoding: In the first experiments, different feature types and their encoding in terms of the BoF representation are evaluated. In order to investigate the influence of the MFCC and GFCC features, they were used individually and in combination. These features are also compared with the well known perceptual feature set (cf. [19]). After deriving a codebook based on these feature representations, different encoding types are

[2]The frame-wise acoustic error measure has also been used for both the DCASE 2013 and 2016 challenge.

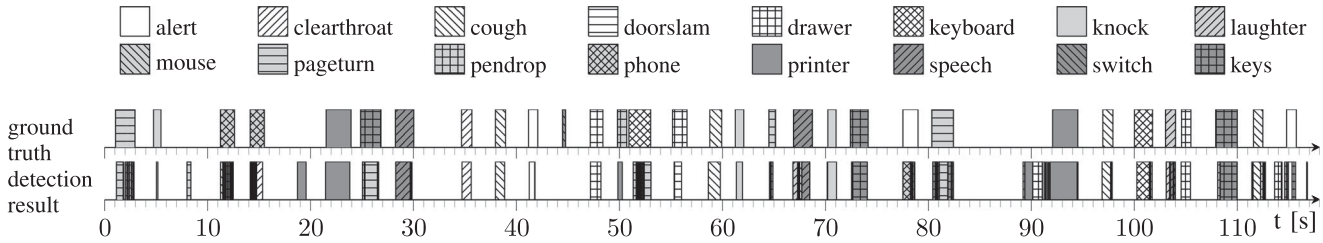

Fig. 4. Example detection results for sequence 01 of the DCASE 2013 office live development set using a BoF approach.

TABLE II
F-SCORES [%] AND STANDARD DEVIATION FOR DIFFERENT FEATURES, ENCODING TYPES AND TEMPORAL INFORMATION ON THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

Codebook	temporal	Quantization		MFCCs, GFCCs, L	MFCCs, GFCCs	MFCCs	GFCCs	Perceptual
unsupervised	no	hard	BoF	48.4 ± 5.0	48.5 ± 4.2	50.0 ± 5.6	38.4 ± 2.5	46.5 ± 2.4
unsupervised	no	soft	BoF	48.1 ± 4.6	45.6 ± 3.4	46.9 ± 3.3	32.9 ± 3.0	47.4 ± 2.4
supervised	no	soft	BoSF	56.1 ± 2.8	55.3 ± 2.9	$\mathbf{54.8 \pm 2.4}$	45.6 ± 2.4	50.3 ± 5.8
supervised	yes	soft	BoSF	$\mathbf{56.3 \pm 3.1}$	$\mathbf{55.4 \pm 2.4}$	51.5 ± 2.8	31.5 ± 4.1	$\mathbf{50.5 \pm 5.1}$
supervised	no	–	GMM	53.6 ± 2.7	52.7 ± 2.9	52.8 ± 2.8	46.0 ± 3.9	46.4 ± 5.8

compared: hard quantization, soft quantization and the soft supervised codebook computation. All combinations of features and encodings have been evaluated using a class specific codebook size of $I = 30$ so that the super-codebook has a size of $L = 30 \cdot 17$. A window size of 0.6 s and a maximum likelihood classifier have been used. The results are shown in Table II.

It can be seen that for both representations, the one with and the one without temporal information, a combination of MFCCs, GFCCs and loudness works best. The combination of loudness, MFCC and GFCC features is significantly better ($p \leq 0.01$,N = 10^5) than the perceptual features or MFCC or GFCC alone for the supervised case with or without temporal augmentation. The results also show the advantage of supervised codebook computation compared to hard or soft quantization. The supervised codebook is outperforming the unsupervised version significantly for all feature sets ($p \leq 0.001$,N = 10^5) .

For further comparison, the GMM approach from [29] has also been evaluated. One mixture model has been fitted to the frames of each class in a supervised manner. For a given window, the class with the highest likelihood over all frames is chosen as a class label. This approach can be seen as a supervised approach that simply omits the quantization step. When comparing the results to the BoF approaches, it can be seen that the additional quantization step allows for a better generalization and improves the performance.

Codebook Size: Different codebook sizes of $I = 20, 30, 40, 50$ were evaluated for a Bag-of-Super-Features (BoSF) approach. Again, a window size of 0.6 s and a maximum likelihood classifier have been used. No temporal information has been incorporated in the BoF representation for the codebook size evaluation. The results are shown in Table III. It can be observed that small codebooks yield good results, e.g., 30 or 40 acoustic words per class. The performance deteriorates when further increasing the size of the codebook, i.e. for $I = 50$. Although the training data can be modeled more precisely with larger codebooks, the model does not generalize well to unknown data anymore.

TABLE III
F-SCORES [%] AND STANDARD DEVIATION FOR DIFFERENT CODEBOOK SIZES ON THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

I	20	30	40	50
BoSF	55.5 ± 3.8	$\mathbf{56.1 \pm 2.8}$	55.8 ± 2.5	54.7 ± 2.6

TABLE IV
F-SCORE AND ERROR RATES [%] WITH STANDARD DEVIATIONS FOR DIFFERENT BOSF CLASSIFIERS ON THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

classifier	F-score	AFER
BoSF ML	$\mathbf{56.1 \pm 2.8}$	$\mathbf{59.8 \pm 3.0}$t2
BoSF SVM HI	48.8 ± 7.1	68.6 ± 6.6
BoSF SVM lin	40.6 ± 8.5	93.5 ± 17.3
BoSF SVM RBF	40.5 ± 7.6	94.8 ± 16.3
BoSF RF	32.8 ± 5.2	85.0 ± 4.9
BoSF KNN	29.2 ± 8.4	84.4 ± 8.0

Classifiers: In the next experiments, different classifiers, as introduced in Section III-D, were compared. All experiments were evaluated using supervised codebook learning and the combined MFCC, GFCC and loudness features without any temporal information. The F-score and the AFER for each classifier are shown in Table IV. The SVM has been evaluated with three different kernels: Histogram Intersection, RBF and a linear kernel. The k-Nearest Neighbor classifier considers 20 neighbors. The Random Forest has been trained using 1000 trees with a maximum depth of 15. Note that the Maximum likelihood classifier outperforms the other classification approaches on this task.

Temporal Processing: For the temporal processing there are two parameters which are of interest. One is the length of the window w in seconds. The other one the number of tiles which are used for the encoding of temporal information within the window. Different numbers of tiles are evaluated for the

TABLE V
F-SCORES [%] AND STANDARD DEVIATION FOR DIFFERENT TEMPORAL STRATEGIES AND DIFFERENT WINDOW LENGTHS AND TILINGS ON THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

	tiles	windowlen			
		0.3	0.6	0.9	1.2
–	–	52.4 ± 2.0	$\mathbf{56.1 \pm 2.8}$	56.0 ± 3.7	52.2 ± 4.2
temporal	2	52.3 ± 2.7	55.3 ± 3.3	53.6 ± 3.9	50.2 ± 3.5
	4	52.7 ± 3.2	55.4 ± 3.3	54.7 ± 3.5	51.7 ± 4.7
	6	52.6 ± 3.1	$\mathbf{56.3 \pm 3.1}$	56.1 ± 2.8	53.5 ± 3.3
	8	52.1 ± 3.3	55.8 ± 3.4	56.0 ± 2.9	54.2 ± 3.1
pyramid	2	52.4 ± 3.1	55.7 ± 3.2	55.9 ± 2.7	52.8 ± 2.6
	4	52.6 ± 3.3	55.8 ± 3.3	56.4 ± 2.9	53.0 ± 3.0
	6	52.3 ± 3.2	56.0 ± 2.8	$\mathbf{57.1 \pm 2.6}$	53.2 ± 3.2
	8	52.1 ± 3.3	56.0 ± 2.5	56.5 ± 3.0	53.1 ± 3.1

TABLE VI
F-SCORE AND ERROR RATES [%] WITH STANDARD DEVIATIONS FOR DIFFERENT CLASSIFIERS ON THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

	classifier	F-score	AFER
proposed	BoSF ML pyramid	$\mathbf{57.1 \pm 3.5}$	60.2 ± 4.1
	BoSF ML temporal	56.3 ± 3.4	60.5 ± 2.7
	BoSF ML	56.1 ± 2.8	$\mathbf{59.8 \pm 3.0}$
re-implemented	BoFr-GMM [29]	49.1 ± 4.0	69.9 ± 4.9
	BAW [1]	37.3 ± 7.2	82.1 ± 7.7
	DNN [35]	29.1 ± 4.6	150.0 ± 2.2
literature	GFB HMM [23] (*)	76.0	69.0
	BG-FG GMM [30]	56.3	88.0
	SVM MFCC [48]	32.5	277.4
	NMF baseline [49]	20.6	162.0

(*) offline approach.

temporal feature augmentation and the temporal pyramid scheme. The F-scores for varying window lengths are shown in Table V. The best overall result of 57.1% is achieved by the pyramid using 6 tiles and a window length of 0.9 s. The best result for the temporal processing is achieved with a smaller context window of 0.6 s and 6 tiles as well. It can be seen that independent of the tiling parameter a window length of 0.6 s shows very stable results for both approaches.

The effect of temporal augmentation is more visible when looking at the class-wise scores in Table VII. For many classes with a temporal dynamic, the recognition improves. When comparing these results with the average event time in Table I, it can be seen that the very short event class *switch* scores lower when introducing temporal information. This is probably due to the fact that these events are shorter than the chosen window length and no good generalization is possible. Similarly the *laughter* and *printer* events seem to be too arbitrary in temporal structure.

Literature Comparison: In order to asses the performance of BoF approaches, the results from the DCASE 2013 office live challenge can be used for comparison. Furthermore, the GMM-based Bag-of-Frames method (denoted as BoFr-GMM) from [29], the Bag-of-Audio words (BAW) method from [1] and the DNN baseline system from [35] were re-implemented and evaluated. All approaches were also applied to time windows of 0.6 s length. Detailed results are shown in Table VI. The Bag-of-Audio words uses MFCC features with first and second derivatives and the mel band energy as features. Hard vector quantization is applied and an SVM with a histogram intersection kernel is used for classification. It achieves an F-score of 47%, which is most likely due to the unsupervised codebook learning. All BoF approaches using supervised codebook learning outperformed the Bag-of-Audio-Words method with F-Scores of 56.1%, 56.3% and 57.1% for the best performing method without temporal information, with temporal features and the temporal pyramid respectively. Similarly, the Bag-of-Frames method that uses MFCC features with first and second derivatives and also applies a supervised learning approach achieves good results but is outperformed by the BoF methods. The difference was proven significant ($p < 0.01, N = 10^5$) by the permutation test.

For the DNN, mel band energies were used as features. The network consists of three fully connected layers with 500 neurons and ReLU activation. A dropout of 10% is applied to each layer. The fully connected layers are followed by a binary class encoding layer with a sigmoid activation function. Thresholding was applied to the networks output scores in order to determine whether a window is considered as silence [35]. It can be seen that in cases of limited training data the DNN does not perform very well.

When comparing these results with the ones published for the DCASE 2013 office live development set, also shown in Table VI, it can be seen that the BoF methods perform in the top range of the other online detection methods. It is however important to note that these results cannot be accurately compared as the evaluation protocol might deviate, for example, with respect to the number of runs or the annotations which have been used in the evaluation. The best performing online method from the literature is the GMM based approach using a separate background model [30]. With an F-score of 56.3% it is well in the range of the BoF approaches. The offline HMM based methods achieve a 20% higher F-score compared to the online methods (cf. [23]).

Discussion: The parameter evaluation revealed a few interesting facts. First of all, the supervised codebook learning plays an important role in the domain of acoustic events. Neither unsupervised hard nor soft quantization came close in performance. In combination with the supervised codebook learning, the maximum likelihood classifier outperforms the other classification approaches. Here, a probability is derived for an acoustic word to occur for a given class which intuitively works well with the supervised codebook learning. As a rule of thumbs a good codebook size can be found somewhere around 30 centroids per class. The size of the classification window is certainly dependent on the length of the events, but if the events are not in the milliseconds area a length of 0.6 s provides a sufficient amount of context.

While MFCCs are the de-facto standard, the combination with GFCCs and loundness improves results. Temporal processing can further improve the performance of BoF approaches.

TABLE VII
CLASSWISE F-SCORE [%] FOR THE DCASE 2013 OFFICE LIVE DEVELOPMENT SET

	alert	clearthroat	cough	doorslam	drawer	keyboard	knock	laughter	mouse	pageturn	pendrop	phone	printer	speech	switch	silence	keys	mean
BoSF ML	**21.3**	25.8	58.7	6.1	7.0	34.9	41.2	**23.2**	2.1	17.6	2.3	57.8	**79.7**	50.2	**23.7**	78.0	42.5	33.7
BoSF ML temporal	13.1	23.0	**51.8**	25.6	**31.7**	**36.1**	57.9	8.5	**6.9**	64.5	4.7	74.4	62.7	**69.3**	2.2	90.9	45.4	39.3
Δ	−8.2	−2.8	−7.0	19.5	24.7	1.2	16.7	−14.7	4.8	47.0	2.3	16.5	−16.9	19.1	−21.4	12.9	3.0	5.7
BoSF ML pyramid	16.6	**26.9**	49.8	**28.3**	26.3	31.5	**64.3**	5.6	3.3	**68.4**	**6.6**	**76.8**	60.4	69.1	2.5	**91.3**	**52.9**	**40.0**
Δ	−4.7	1.1	−9.0	22.2	19.3	−3.4	23.1	−17.6	1.2	50.9	4.3	19.0	−19.2	18.9	−21.2	13.2	10.5	6.4

The influence of incorporating different temporal configurations is shown. The delta is shown with respect to the BoSF ML baseline.

However, the improvement is not as tremendous as in other domains (cf. [43]).

Compared with results from the literature, BoF approaches show state-of-the-art performance for online methods. One limitation of this approach is that it tends to be slightly noisy, as is revealed by the exemplary results (see Fig. 4). This can, for example, be overcome by a post processing step that applies smoothing (cf. [15]).

B. ITC-IRST Dataset

The ITC-IRST dataset is also comprised of 16 different acoustic events, including *door knock, door slam, steps, chair moving, spoon (cup jingle), paper wrapping, key jingle, keyboard typing, phone ring, applause, cough, laugh, door open, phone vibration, mimo pen buzz, falling object* and an additional *unknown/background* class. The recording room for the ITC-IRST dataset was equipped with 32 microphones. Four were table microphones and the remaining 28 were located in seven T-shaped arrays on the walls. The experiments consist of twelve recording sessions which were made on three different days. Thus, four recording sessions were made each day. The first three sessions are considered as training data and the last session of each day is used for testing. It is important to know that the locations of the acoustic events were changed every day. Two thirds of the training set are used for learning the single-channel BoF models and the single-channel classifiers. The remaining third of the data is used for training the classifier stacking. In total the dataset contains 1 hour 42 minutes and 25 seconds of recordings of which 50 minutes and 44 seconds are event data. A detailed overview is given in Table VIII. Note that the dataset is larger than the DCASE 2013 office live set and contains on average longer event classes.

Fusion Experiments: In order to assess the performance of BoF methods in a multichannel setting, an approach without temporal information is evaluated. All 32 channels of the ITC dataset are used for evaluation and the mean result over all channels is reported. The experiments were conducted using all sound classes except *silence* and *unknown* as foreground.

The fusion strategy that is based on classifier stacking is evaluated and compared to heuristic fusion strategies: *max rule*, *product rule* and *majority voting* [12]. The results are shown in Table IX. Note that no standard deviation can be reported for the fusion as there is only one result after fusing the predictions of all 32 channels. It can be seen that a learned fusion outperforms the heuristics approaches. There is a strong improvement in the robustness when fusing the results. In comparison with the mean of the single channels, the F-Score is improved by 6.3%.

TABLE VIII
OVERVIEW OF THE ITC-IRST DATASET

	training time [s]		test time [s]	
class	total	event avg.	total	event avg.
applause	53.45	5.94	18.95	6.32
chair moving	115.60	3.30	38.11	3.18
cough	72.34	2.01	30.44	2.54
door knock	59.84	1.71	19.83	1.65
door open	63.83	1.77	17.16	1.32
door slam	62.63	1.61	20.74	1.73
falling object	51.63	1.43	16.12	1.34
key jingle	244.88	6.80	84.21	7.02
keyboard clicking	216.03	6.17	76.47	6.37
laugh	70.02	1.95	24.59	2.05
paper wrapping	185.17	5.14	71.49	5.96
pen buzz	241.43	6.71	95.27	7.94
phone ring	388.25	5.88	123.17	5.36
phone vibration	50.52	5.05	14.75	4.92
spoon,cup jingle	219.89	6.11	70.65	5.89
steps	178.66	4.70	48.10	4.01

TABLE IX
F-SCORES [%] FOR DIFFERENT FUSION STRATEGIES ON THE ITC-IRST DATASET USING ALL 16 CLASSES AS FOREGROUND

multichannel	fusion	F-score
no	–	77.5 ± 1.0
yes	max rule [12]	77.2
yes	product rule [12]	79.2
yes	voting [12]	79.2
yes	stacking [17]	**83.8**

Literature Comparison: For comparison with the results from the literature, only the first twelve classes are considered as foreground classes (cf. [8], [13]). In [13] a Random Regression Forest has been proposed and has been evaluated using this setup and four channels. The fusion showed of these channels an improvement over a single channel evaluation as well as the baselines reported in [8]. Furthermore, the re-implemented GMM [29] and DNN [35] baseline systems were also evaluated on all channels of the ITC-IRST dataset.

The Results in Table X show that the BoF approach yields a similar performance compared to the Regression Forests and the DNN baseline, while the GMM approach is outperformed. The DNN is able to close the performance gap that was previously

TABLE X
RESULTS ON THE ITC-IRST DATASET USING THE CLEAR EVALUATION PROTOCOL WITH THE FIRST 12 CLASSES AS FOREGROUND IN COMPARISON TO LITERATURE RESULTS

method	multichannel	fusion	F-score	AFER
GMM [29]	no	–	71.1 ± 2.5	48.9 ± 6.3
DNN [35]	no	–	81.7 ± 0.5	34.7 ± 0.2
Regr. Forest [13]	yes	sum (*)	82.8	30.7
BoSF ML	no	–	77.4 ± 1.1	39.0 ± 2.2
BoSF ML	yes	stacking	**84.1**	**26.0**

(*) four of the 32 channels were used for the fusion.

observed when there is enough training data. When fusing the results of all 32 channels using classifier stacking, the BoF approach outperforms the regression forest and also shows better results than the average over all channels using the DNN in both F-score and AFER.

C. Qualitative Remarks

Due to the rapid computation of the BoF principle, the system is capable to run in real time. The current python implementation uses a single core on a standard desktop machine and requires less than 20% of the real time for computation.[3] As a result of the sliding window approach, there is a processing delay of $w/2 = 0.3$ s, which is negligible for most practical applications. Compared to deep neural networks, the BoF approaches require only a fraction of the computational cost.

V. CONCLUSION

This paper reviewed the BoF principle for acoustic event detection and classification. The work presented here builds on the previous work introduced in [15]–[17], but also analyzed further design decisions within the BoF paradigm. A comparative evaluation for indoor meeting room scenarios has been shown. Insights on the parameterizations within the BoF framework and their influence on the acoustic event detection have been provided. Improvements based on psychoacoustic models for features as well as a supervised codebook learning step have been shown. A critical discussion on the influence of temporal information as well as codebook sizes has been provided. Furthermore, an extension to multiple channels has been explained which improves the performance of BoF approaches. Here, it could be shown that classifier stacking outperforms heuristic multichannel fusion strategies.

The evaluation showed that BoF approaches yield state-of-the-art performance in both single and multi channel setups for online AED. The evaluation focused on indoor scenarios with only little noise. A BoF approach for outdoor applications has been discussed in [18]. Scenarios with more background clutter may require additional effort for noise reduction [11]. Furthermore, larger distributed microphone arrays will require additional synchronization effort.

[3]Code available at http://patrec.cs.tu-dortmund.de/resources/

The low computational complexity of BoF approaches is an interesting property as it allows for real time processing. In contrast to the powerful DNN approaches, BoF methods perform well in scenarios where training data is scarce.

ACKNOWLEDGMENTS

The authors would like to thank J. Kürby for providing the implementation and subsequent results of the multichannel fusion. The authors would also like to thank H. Phan for his feedback and implementation details regarding the multichannel regression forest approach.

REFERENCES

[1] S. Pancoast and M. Akbacak, "Bag-of-audio-words approach for multimedia event classification," in *Proc. Interspeech*, Portland, OR, USA, 2012, pp. 2105–2108.
[2] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah, "High-level event recognition in unconstrained videos," *Int. J. Multimedia Inf. Retrieval*, vol. 2, no. 2, pp. 73–101, 2013.
[3] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, USA, Nov. 2014, pp. 1041–1044.
[4] S. H. Young and M. V. Scanlon, "Robotic vehicle uses acoustic array for detection and localization in urban environments," *SPIE Proc. Mobile Robot Perception*, vol. 4364, pp. 264–273, Sep. 2001.
[5] H. Klinck, K. Stelzer, K. Jafarmadar, and D. K. Mellinger, "AAS endurance: An autonomous acoustic sailboat for marine mammal research," in *Proc. Int. Robot. Sailing Conf.*, Jul. 2009, pp. 43–48.
[6] V. Carletti, P. Foggia, G. Percannella, A. Saggese, N. Strisciuglio, and M. Vento, "Audio surveillance using a bag of aural words classifier," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveillance*, Aug. 2013, pp. 81–86.
[7] P. Foggia, A. Saggese, N. Strisciuglio, and M. Vento, "Cascade classifiers trained on gammatonegrams for reliably detecting audio events," in *Proc. 11th IEEE Int. Conf. Adv. Video Signal Based Surveillance*, 2014, pp. 50–55.
[8] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, "CLEAR evaluation of acoustic event detection and classification systems," in *Multimodal Technologies for Perception of Humans* (ser. Lecture Notes in Computer Science), vol. 4122, R. Stiefelhagen and J. Garofolo, Eds. Berlin, Germany: Springer, 2007, pp. 311–322.
[9] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang, "Automatic analysis of multimodal group actions in meetings," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 305–17, Mar. 2005.
[10] A. Plinge and G. A. Fink, "Online multi-speaker tracking using multiple microphone arrays informed by auditory scene analysis," in *Proc. Eur. Signal Process. Conf.*, 2013, pp. 1–5.
[11] A. Plinge and S. Gannot, "Multi-microphone speech enhancement informed by auditory scene analysis," in *Proc. Sensor Array Multichannel Signal Process. Workshop*, Rio de Janeiro, Brazil, 2016, pp. 1–5.
[12] P. Giannoulis, G. Potamianos, A. Katsamanis, and P. Maragos, "Multi-microphone fusion for detection of speech and acoustic events in smart spaces," in *Proc. IEEE Eur. Signal Process. Conf.*, 2014, pp. 2375–2379.
[13] H. Phan, M. Maass, L. Hertel, R. Mazur, and A. Mertins, "A multi-channel fusion framework for audio event detection," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2015, pp. 1–5.
[14] K. Imoto and N. Ono, "Spatial-feature-based acoustic scene analysis using distributed microphone array," in *Proc. IEEE Eur. Signal Process. Conf.*, 2015, pp. 734–738.
[15] A. Plinge, R. Grzeszick, and G. Fink, "A bag-of-features approach to acoustic event detection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2014, pp. 3704–3708.
[16] R. Grzeszick, A. Plinge, and G. A. Fink, "Temporal acoustic words for online acoustic event detection," in *Proc. German Conf. Pattern Recognit.*, 2015, pp. 142–153.
[17] J. Kuerby, R. Grzeszick, A. Plinge, and G. A. Fink, "Bag-of-feature acoustic event detection for sensor networks," in *Proc. Workshop Detect. Classification Acoust. Scenes Events*, Budapest, Hungary, Sep. 2016, pp. 55–59.

[18] J. Salamon and J. P. Bello, "Unsupervised feature learning for urban sound classification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 171–175.
[19] A. Temko and C. Nadeu, "Classification of acoustic events using SVM-based clustering schemes," *Pattern Recognit.*, vol. 39, no. 4, pp. 682–694, Apr. 2006.
[20] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1733–1746, Oct. 2015.
[21] *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016)*. Tampere University of Technology. Department of Signal Processing, Sep. 2016.
[22] G. A. Fink, *Markov Models for Pattern Recognition, From Theory to Applications*, 2nd ed. (ser. Advances in Computer Vision and Pattern Recognition). London, U.K.: Springer, 2014.
[23] J. Schröder *et al.*, "Acoustic event detection using signal enhancement and spectro-temporal feature extraction," *IEEE AASP Challenge: Detect. Classif. Acoustic Scenes and Events,* Tech. Rep., 2013.
[24] A. Diment, T. Heittola, and T. Virtanen, "Sound event detection for office live and office synthetic AASP challenge," IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep., 2013.
[25] J. Schröder, J. Anemüller, and S. Goetze, "Performance comparison of GMM, HMM and DNN based approaches for acoustic event detection within task 3 of the DCASE 2016 challenge," in *Proc. Workshop Detect. Classification Acoust. Scenes Events*, Budapest, Hungary, Sep. 2016, pp. 80–84.
[26] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 6440–6444.
[27] R. Togneri and D. Pullella, "An overview of speaker identification: Accuracy and robustness issues," *IEEE Circuits Syst. Mag.*, vol. 11, no. 2, pp. 23–61, Apr.–Jun. 2011.
[28] 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 *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2013, pp. 1–4.
[29] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Proc. Eur. Signal Process. Conf.*, Sep. 2016, pp. 1128–1132.
[30] L. Vuegen, B. V. D. Broeck, P. Karsmakers, J. F. Gemmeke, B. Vanrumste, and H. V. Hamme, "An MFCC-GMM approach for event detection and classification," *IEEE AASP Challenge: Detect. Classif. Acoust. Scenes Events,* Tech. Rep., 2013.
[31] J. A. Stork, L. Spinello, J. Silva, and K. O. Arras, "Audio-based human activity recognition using non-Markovian ensemble voting," in *Proc. IEEE Int. Symp. Robot Hum. Interact. Commun.*, 2012, pp. 509–514.
[32] H. Phan, M. Maasz, R. Mazur, and A. Mertins, " Random regression forests for acoustic event detection and classification," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 23, no. 1, pp. 20–31, Jan. 2015.
[33] H. Phan, M. Maass, R. Mazur, and A. Mertins, "Early event detection in audio streams," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2015, pp. 1–6.
[34] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2015, pp. 1–7.
[35] Q. Kong, I. Sobieraj, W. Wang, and M. Plumbley, "Deep neural network baseline for DCASE challenge 2016, " in *Proc. Workshop Detection Classif. Acoust. Scenes Events*, Budapest, Hungary, Sep. 2016, pp. 50–54.
[36] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, and W. Xiao, "Robust sound event classification using deep neural networks," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 3, pp. 540–552, Mar. 2015.
[37] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. ACM Press: New York, 1999.
[38] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, no. 4, pp. 195–218, Aug. 1980.
[39] D. Wang and G. J. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Piscataway, NJ, USA: IEEE Press, 2006.
[40] Y. Shao, S. Srinivasan, and D. Wang, "Incorporating auditory feature uncertainties in robust speaker identification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2007, pp. 277–280.
[41] M. Unoki and M. Akagi, "A method of signal extraction from noisy signal based on auditory scene analysis," *Speech Commun.*, vol. 27, no. 3, pp. 261–279, 1999.
[42] H. Tang, S. M. Chu, M. Hasegawa-Johnson, and T. S. Huang, "Partially supervised speaker clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 959–971, May 2012.
[43] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 2, pp. 2169–2178.
[44] R. Grzeszick, L. Rothacker, and G. A. Fink, "Bag-of-features representations using spatial visual vocabularies for object classification," in *Proc. IEEE Int. Conf. Image Process.*, Melbourne, Vic., Australia, 2013, pp. 2867–2871.
[45] J. Sánchez, F. Perronnin, and T. De Campos , "Modeling the spatial layout of images beyond spatial pyramids," *Pattern Recognit. Lett.*, vol. 33, no. 16, pp. 2216–2223, 2012.
[46] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001.
[47] H. Phan, P. Koch, M. Maass, R. Mazur, I. McLoughlin, and A. Mertins, "What make audio event detection harder than classification?", *CoRR*, abs/1612.09089, http://arxiv.org/abs/1612.09089, 2016.
[48] W. Nogueira, G. Roma, and P. Herrera, "Automatic event classification using front end single channel noise reduction, MFCC features and a support vector machine classifier," *IEEE AASP Challenge: Detect. Classif. Acoust. Scenes Events*, Tech. Rep., 2013.
[49] D. Giannoulis, D. Stowell, E. Benetos, M. Rossignol, M. Lagrange, and M. D. Plumbley, "A database and challenge for acoustic scene classification and event detection," in *Proc. Eur. Signal Process. Conf.*, Marrakech, Morocco, 2013, pp. 1–5.

René Grzeszick (S'13) received the Bachelor's and Master's degree in computer science from TU Dortmund University, Dortmund, Germany, in 2010 and 2012, repectively. Afterwards, he joined the Pattern Recognition in Embedded Systems group in the Department of Computer Science, TU Dortmund University, as a Ph.D. student. His research interests include pattern recognition methods with applications in computer vision and acoustics. His research on partially supervised learning approaches for natural scene images is currently funded by the German Research Fundation.

Axel Plinge (M'11) received the Diploma degree with distinction in computer science with a minor in philosophy from TU Dortmund University, Dortmund, Germany, in 2010. From 2000 he worked in different areas of psychophysical research from hearing to color vision and depth perception at the Leibniz Research Centre for Working Environment and Human Factors, Dortmund, Germany. There he participated in EC research projects and is coinventor of novel methods in speech technology for persons with impaired hearing. In 2012, he joined the pattern recognition group of the Department of Computer Science at TU Dortmund University, Dortmund, Germany. There he published multiple papers on acoustic sensor networks geometry calibration, speaker tracking, and sound classification.

Gernot A. Fink (M'94–SM'07) received the Diploma in computer science from the University of Erlangen-Nürnberg, Erlangen, Germany, in 1991. He received the Ph.D. degree (Dr.-Ing.) in computer science and the Venia Legendi (Habilitation) from Bielefeld University, Germany, in 1995 and 2002, respectively. From 1991 to 2005, he was with the Applied Computer Science Group from the Faculty of Technology of Bielefeld University, Germany. Since 2005, he has been a Professor at TU Dortmund University, Dortmund, Germany, where he heads the Pattern Recognition in Embedded Systems group. His research interests include multimodal machine perception statistical pattern recognition, and document analysis. He has published more than 150 papers in these fields and is the author of a book on Markov models for pattern recognition.