A MULTI-CHANNEL FUSION FRAMEWORK FOR AUDIO EVENT DETECTION

Huy Phan[⋆†]*, Marco Maass*[⋆]*, Lars Hertel*[⋆]*, Radoslaw Mazur*[⋆]*,* and *Alfred Mertins*[⋆]

[⋆]Institute for Signal Processing, University of Lübeck, Germany
[†]Graduate School for Computing in Medicine and Life Sciences, University of Lübeck, Germany
Email: {phan, maass, hertel, mazur, mertins}@isip.uni-luebeck.de

ABSTRACT

We propose in this paper a simple, yet efficient multi-channel fusion framework for joint acoustic event detection and classification. The joint problem on individual channels is posed as a regression problem to estimate event onset and offset positions. As an intermediate result, we also obtain the posterior probabilities which measure the confidence that event onsets and offsets are present at a temporal position. It facilitates the fusion problem by accumulating the posterior probabilities of different channels. The detection hypotheses are then determined based on the summed posterior probabilities. While the proposed fusion framework appears to be simple and natural, it significantly outperforms all the single-channel baseline systems on the ITC-Irst database. We also show that adding channels one by one into the fusion system yields performance improvements, and the performance of the fusion system is always better than those of the individual-channel counterparts.

Index Terms— Acoustic event detection, classification, multi-channel fusion, regression forests

1. INTRODUCTION

Acoustic event detection and classification (AED/C) [1] recently draws great attention of audio research community [2]. Its potential applications are diverse, such as surveillance [3], healthcare [4], and meeting room transcription [5], among many others. Compared to AEC [6,7] which performs on isolated AEs, AED on continuous audio recordings [8] is much more difficult due to nonstationary background noise, duration variance, as well as event overlap. In particular, temporal localization, i.e. determination of event boundaries in time, is also important for some applications [5, 9] that require a good temporal resolution of the detected AEs. However, in recent AED evaluation campaigns [10–12], it was found that achieving an accurate AE detection and localization from the continuous audio is very challenging.

So far, the most commonly used method for AED has been based on automatic speech recognition (ASR) framework [13] where AEs are modeled as sequences of frame-level feature vectors using Hidden Markov Models (HMM) [14, 15]. Most of the submissions to the recent AED evaluations, including CLEAR 2006/2007 [10, 11], and AASP DCASE 2013 [12], are of this kind. However, the performance was not satisfactory as expected due to the natural difference between speech and AEs. Specifically, AEs exhibit a wider range of characteristics and non-stationary effects which may not be captured in frame-based features [16]. Furthermore, HMM-based modelling is inefficient to handle high intra-class variation which are usually seen in AEs. Another common trend is detection-by-classification [5, 17, 18] in which classification, e.g. by Support Vector Machines (SVMs), is performed on long sliding windows. Nevertheless, this approach faces difficulty in localization due to large temporal scales of AEs. In general, the ASR framework has shown more advantages than detection-by-classification [10].

In our previous works [8, 19], we proposed a regression approach to deal with the joint AED/C problem using random regression forests [20, 21]. The audio signals are decomposed into *superframes* [8]. Each event superframe maintains its displacements to the corresponding event onset and offset. On testing, inputted with a test superframe, the learned regressors are able to estimate positions of the AE onset and offset positions relative to it, i.e. the boundary of the AE is jointly determined. In this paper, we extend the work in [8] to develop a multi-channel fusion framework for AED/C.

The majority of works in literature have tackled single-channel AED mainly due to its simplicity. Very few attempts have considered to resolve multi-channel fusion. Recording more data with additional microphones offers multiple views of the same problem and one would expect an improvement when multiple audio channels are integrated. Unfortunately, it is mostly not the case for AED. It has been shown that a naive fusion strategy would deteriorate the system instead [11, 22, 23]. While treating the joint AED/C as a regression task is by itself novel, our framework is also very distinguishable from the previous works [22–24]. The experimental results on the ITC-Irst database of non-overlapping events show that our fusion system outperforms not only the common approaches but also its single-channel counterparts. Furthermore, when an additional data source is added, we obtain a better performance.

2. REGRESSION FORESTS FOR AED

In this section, we describe how to learn a regresssor for event onset and offset prediction with the random decision forests framework [8, 20]. The idea is to group the training audio segments, i.e. superframes [8], into hypercubes of the feature space so that those superframes in the same hypercube have similar distances from event onset and offset positions. It turns out that the distances can be modeled with a simple model. The regressor is specific for a target event category.

2.1. Training

The isolated events in training data are divided into interleaved *superframes* [8], which are 100 ms audio segments, to obtain the set of annotated superframes $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{d}_i)\}$. A superframe is

This work was supported by the Graduate School for Computing in Medicine and Life Sciences funded by Germany's Excellence Initiative [DFG GSC 235/1].

represented by $\mathbf{x} \in \mathbb{R}^M$ where M is the feature dimensionality. $\mathbf{d} = (d_s, d_e) \in \mathbb{R}_+^2$ denotes the displacement vector (in superframes) of the superframe to the event onset t_s and offset t_e inclusive. The onset displacement d_s and offset displacement d_e are computed as

$$d_s = t - t_s, \tag{1}$$

$$d_e = t_e - t. \tag{2}$$

In order to construct a tree of the regression forest $\mathcal{F}$, a subset of superframes is randomly sampled from $\mathcal{S}$. Starting from the root, at a split node ℓ a set of binary tests $t_{f,\tau}$ defined in (3) is generated.

$$t_{f,\tau}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x}^f > \tau \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Here, $\mathbf{x}^f$ denotes the value of $\mathbf{x}$ at the randomly selected feature channel $f \in \{1, \ldots, M\}$. The variable τ is a random threshold generated in the range of $\mathbf{x}^f$. An optimal test is then adopted from the test set to split the superframe set $\mathcal{S}_\ell$ at the split node ℓ into two sets $\mathcal{S}_\ell^{\text{right}}$ and $\mathcal{S}_\ell^{\text{left}}$:

$$\mathcal{S}_\ell^{\text{right}} = \{(\mathbf{x}, \mathbf{d}) \in \mathcal{S}_\ell \big| t_{f,\tau}(\mathbf{x}) = 1\}, \tag{4}$$

$$\mathcal{S}_\ell^{\text{left}} = \{(\mathbf{x}, \mathbf{d}) \in \mathcal{S}_\ell \big| t_{f,\tau}(\mathbf{x}) = 0\}. \tag{5}$$

$\mathcal{S}_\ell^{\text{right}}$ and $\mathcal{S}_\ell^{\text{left}}$ are subsequently sent to the right and the left child nodes, respectively. The adoption criteria is to minimize the *displacement uncertainty* U:

$$U = \sum \left\| \mathbf{d}_i^{\text{left}} - \bar{\mathbf{d}}^{\,\text{left}} \right\|_2^2 + \sum \left\| \mathbf{d}_i^{\text{right}} - \bar{\mathbf{d}}^{\,\text{right}} \right\|_2^2. \tag{6}$$

Here, $\bar{\mathbf{d}}$ denotes the mean displacement vector of the corresponding superframe set indicated by the superscript. By this, the superframes are clustered by both their features and their relative positions to event onsets and offsets.

The splitting process is recursively repeated until the maximum depth $D_{\max}$ is reached or a minimum number of superframes $N_{\min}$ is remained. Then a leaf node will be created. The displacement vectors of the remaining superframes at the leaf node are modeled and stored as a two-dimensional Gaussian distribution $\mathcal{N}(\mathbf{d}\big|\bar{\mathbf{d}}, \mathbf{\Gamma})$:

$$\mathcal{N}(\mathbf{d}\big|\bar{\mathbf{d}}, \mathbf{\Gamma}) = \frac{1}{2\pi\sqrt{\det(\mathbf{\Gamma})}} \exp\big(-\frac{1}{2}(\mathbf{d}-\bar{\mathbf{d}})^T \mathbf{\Gamma}^{-1}(\mathbf{d}-\bar{\mathbf{d}})\big), \tag{7}$$

where $\bar{\mathbf{d}} = (\bar{d}_s, \bar{d}_e)$ and $\mathbf{\Gamma} = \begin{pmatrix} \Gamma_s & 0 \\ 0 & \Gamma_e \end{pmatrix}$ are, respectively, the mean and the covariance matrix of the displacement vectors. However, for simplicity we do not consider covariance between onset and offset displacements. That is, $\mathcal{N}(\mathbf{d}\big|\bar{\mathbf{d}}, \mathbf{\Gamma})$ is equivalent to two univariate Gaussian distributions $\mathcal{N}_s(d\big|\bar{d}_s, \Gamma_s)$ and $\mathcal{N}_e(d\big|\bar{d}_e, \Gamma_e)$:

$$\mathcal{N}_s(d\big|\bar{d}_s, \Gamma_s) = \frac{1}{\sqrt{2\pi\Gamma_s}} \exp\big(-\frac{(d-\bar{d}_s)^2}{2\Gamma_s}\big), \tag{8}$$

$$\mathcal{N}_e(d\big|\bar{d}_e, \Gamma_e) = \frac{1}{\sqrt{2\pi\Gamma_e}} \exp\big(-\frac{(d-\bar{d}_e)^2}{2\Gamma_e}\big). \tag{9}$$

The above algorithm is repeated to grow all the trees in the forest $\mathcal{F}$.

2.2. Testing

Given a test superframe $\mathbf{x}$, we aim at estimating its displacements from the onset and offset of a target event using the learned regression forest $\mathcal{F}$. We input $\mathbf{x}$ into a tree $\mathcal{T}_i$ of $\mathcal{F}$. At each split node, the stored binary test is evaluated on $\mathbf{x}$, directing it either to the right or left child until ending up at a leaf node ℓ_i. From (8) and (9), estimates of the onset and offset displacements are obtained in terms of the Gaussian distributions stored at ℓ_i:

$$p_{d_s}(d|\ell_i, \mathbf{x}) = \mathcal{N}_s(d\big|\bar{d}_s^{\ell_i}, \Gamma_s^{\ell_i}), \tag{10}$$

$$p_{d_e}(d|\ell_i, \mathbf{x}) = \mathcal{N}_e(d\big|\bar{d}_e^{\ell_i}, \Gamma_e^{\ell_i}). \tag{11}$$

The posterior probabilites are finally computed by summing up $p_{d_s}(d|\ell_i, \mathbf{x})$ and $p_{d_e}(d|\ell_i, \mathbf{x})$ over all trees of the forest $\mathcal{F}$:

$$p_{d_s}(d|\mathbf{x}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_s(d\big|\bar{d}_s^{\ell_i}, \Gamma_s^{\ell_i}), \tag{12}$$

$$p_{d_e}(d|\mathbf{x}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_e(d\big|\bar{d}_e^{\ell_i}, \Gamma_e^{\ell_i}). \tag{13}$$

Here $\big|\mathcal{F}\big|$ denotes the number of trees of the forest $\mathcal{F}$. The expectations of $p_{d_s}(d|\mathbf{x})$ and $p_{d_e}(d|\mathbf{x})$, respectively, indicate the onset and offset displacements estimated by the superframe $\mathbf{x}$.

3. MULTI-CHANNEL FUSION FRAMEWORK

3.1. The proposed fusion framework

We want to estimate where in time an event starts and ends in a continuous audio signal. Let t and t' both denote the time index. From (12) and (13), an event superframe $\mathbf{x}_{t'}$ at the time t' gives estimates of the onset and offset displacements as

$$p_{d_s}(d|\mathbf{x}_{t'}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_s(d\big|\bar{d}_s^{\ell_i}, \Gamma_s^{\ell_i}), \tag{14}$$

$$p_{d_e}(d|\mathbf{x}_{t'}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_e(d\big|\bar{d}_e^{\ell_i}, \Gamma_e^{\ell_i}). \tag{15}$$

From (1) and (2), estimates for the onset and offset positions are then obtained by placing $\mathcal{N}_{d_s}(d\big|\bar{d}_s^{\ell_i}, \Gamma_s^{\ell_i})$ in (14) at $\bar{d}_s^{\ell_i}$ backward from t' and $\mathcal{N}_{d_e}(d\big|\bar{d}_e^{\ell_i}, \Gamma_e^{\ell_i})$ in (15) at $\bar{d}_e^{\ell_i}$ forward from t':

$$p_{t_s}(t|\mathbf{x}_{t'}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_s(t\big|t' - \bar{d}_s^{\ell_i}, \Gamma_s^{\ell_i}), \tag{16}$$

$$p_{t_e}(t|\mathbf{x}_{t'}) = \frac{1}{\big|\mathcal{F}\big|} \sum_i \mathcal{N}_e(t\big|t' + \bar{d}_e^{\ell_i}, \Gamma_e^{\ell_i}). \tag{17}$$

The estimates by all superframes are accumulated to yield the confidence scores that the onset and offset positions of the target event coincide at a time t:

$$f_s(t) = \sum_{t'} p_{t_s}(t|\mathbf{x}_{t'}), \tag{18}$$

$$f_e(t) = \sum_{t'} p_{t_e}(t|\mathbf{x}_{t'}). \tag{19}$$

When multiple sources are available, data fusion can be done very naturally by accumulating the confidence scores:

$$f_s(t) = \sum_c \sum_{t'} p_{t_s}^c(t|\mathbf{x}_{t'}^c), \tag{20}$$

$$f_e(t) = \sum_c \sum_{t'} p_{t_e}^c(t|\mathbf{x}_{t'}^c). \tag{21}$$

Here, c indicates the channel index. The regressors are learned separately for different channels. Ideally, if there exists only one event instance in the signal, its onset and offset positions can be determined as:

$$\hat{t}_s = \arg\max_t f_s(t), \tag{22}$$

$$\hat{t}_e = \arg\max_t f_e(t). \tag{23}$$

However, an audio stream typically contains multiple event occurrences, one after another, which must be detected sequentially. To accomplish this, we determine a threshold for the confidence scores. For simplicity, we employ a common threshold β for both onset and offset confidence scores f_s and f_e. As soon as both accumulating scores reach the threshold, the event is considered detected.

3.2. Handling multi-class event categories

Our regression forests are specific for a target event category. In general, it is common that multiple event types are targeted. Out of $\mathcal{Y}$ event categories of interest and $\mathcal{C}$ available channels, we trained a regression forest $\mathcal{F}^{y,c}$ for a category $y \in \{1,\ldots,\mathcal{Y}\}$ on a channel $c \in \{1,\ldots,\mathcal{C}\}$. Due to the fact that the regression forests were trained with class-specific data, it is necessary to provide them with class-specific data to make proper estimates. We perform a superframe-wise classification step before regression. The superframes are firstly passed to the binary classifier M^c_{bg} which filters out the background and only allows event superframes passing through. Subsequently, these event superframes are classified as one of the event categories of interest by the multi-class classifier M^c_{ev}. Finally, a superframe recognized as class y is inputted to the regression forest $\mathcal{F}^{y,c}$ for estimation. We trained the classifiers M^c_{bg} and M^c_{ev} specifically for each channel c with random forest classification [25]. Moreover, we take advantage of the probability outputs M^c_{ev} to weight the contribution of a superframe to the final confidence scores. As a result, (20) and (21) are re-written as:

$$f_s(t) = \sum_c \sum_{t'} \delta_{\hat{y}^c_{t'},y} \cdot p(\hat{y}^c_{t'} = y) \cdot p^c_{t_s}(t|\mathbf{x}^c_{t'}), \tag{24}$$

$$f_e(t) = \sum_c \sum_{t'} \delta_{\hat{y}^c_{t'},y} \cdot p(\hat{y}^c_{t'} = y) \cdot p^c_{t_e}(t|\mathbf{x}^c_{t'}). \tag{25}$$

Here, $\hat{y}^c_{t'}$ denotes the predicted label of $\mathbf{x}^c_{t'}$ and δ is Kronecker delta function. $p(\hat{y}^c_{t'} = y)$ is the probability that the predicted class label $\hat{y}^c_{t'}$ equals y. By weighting, a superframe recognized with higher confidence will reasonably contribute more into the estimation.

4. EXPERIMENT

4.1. Experiment setup

Database: we conducted experiments on the ITC-Irst database [26] which does not contain event overlap. It was recorded with multiple microphone arrays and consists of twelve recording sessions. There are totally 16 semantic event categories including door knock, door slam, steps, chair moving, spoon cup jingle, paper wrapping, key jingle, keyboard typing, phone ring, applause, cough, laugh, mimo pen buzz, falling object, phone vibration, and unknown. To agree with CLEAR 2006 challenge [10] and our previous work [8], we evaluate the first twelve classes while the rest is considered as background. Nine recording sessions were employed as training files

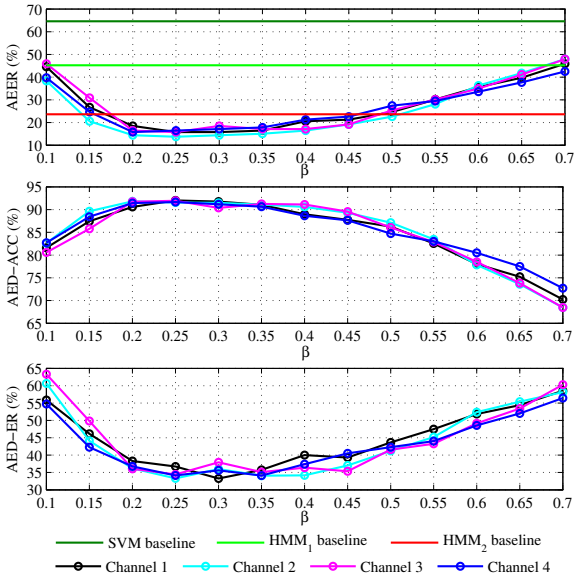

Figure 1: Single-channel detection performance over the parameter β for ITC-Irst database.

and three remaining sessions were employed as test files. Four microphones, including: *T0_1*, *T0_2*, *T4_1*, and *T5_1*, with respect to four side-walls of the room [26], were selected for use in the experiments. A similar setting was attempted by Temko *et al.* [22], but their experiments showed a deterioration on the performance.

Features and Parameters: the audio signals were decomposed into interleaved superframes, which are 100 ms long, with an overlap of 90%. The dense overlap is to ensure a high level of data correlation. To represent a superframe, we divide it into small 30 ms frames with Hamming window and 20 ms overlap. We utilize the set of 60 acoustic features suggested in [8, 10] to represent a small frame. They consist of: (1) 16 log-frequency filter bank parameters, along with the first and second time derivatives, and (2) the following set of features: zero-crossing rate, short time energy, four sub-band energies, spectral flux calculated for each sub-band, spectral centroid, and spectral bandwidth. In turn, a superframe consisting of multiple small frames is represented by the empirical mean and the standard deviation of the frame feature vectors.

The classifiers M^c_{bg} and M^c_{ev} were trained using random forest classification, and we set the number of trees to 300. The regressors were trained with ten random trees each. For a category y, a randomly sampled subset containing 50% superframes of the training set was used to train each random tree. During training, 20,000 binary tests were generated for a split node. In addition, we set the maximum depth to $D_{\max} = 12$ and the minimum number of superframes at leaf nodes to $N_{\min} = 10$. The threshold β can be selected by cross validation and it should be adapted for different categories since their posterior probabilities show different characteristics [8]. However, for simplicity, we utilized a common threshold β for all event categories and we set $\beta = 0.25$ in the performance comparison in Section 4.2.

Evaluation metrics: we used three metrics for evaluation: *Acoustic Event Error Rate (AEER)* [10], *AED accuracy (AED-ACC)*, and *AED error rate (AED-ER)* [11]. The AEER and AED-ACC metrics focus on the detection of AE instances. On the other hand, AED-ER focuses more on AE temporal localization. These metrics were used in CLEAR 2006/2007 challenges [10, 11] and

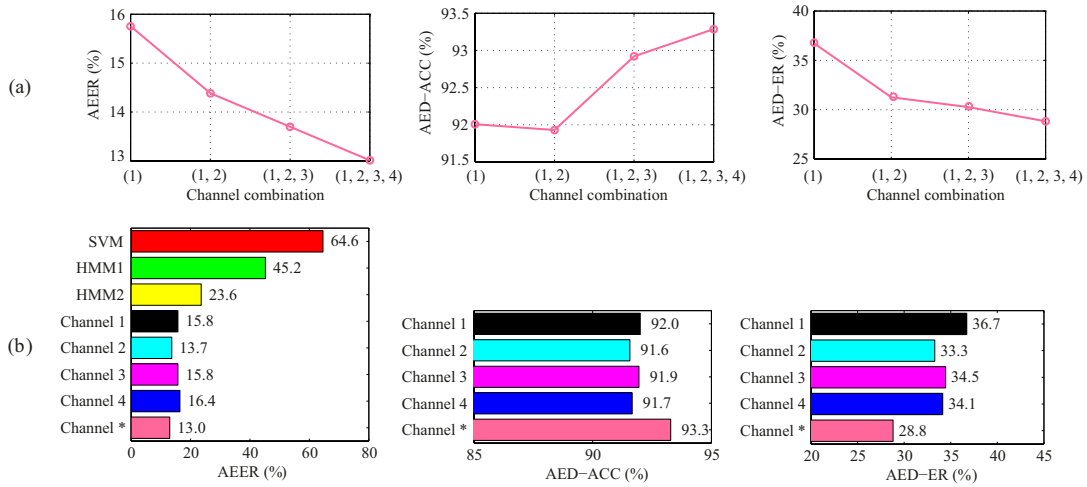

Figure 2: Multi-channel fusion results: (a) improvements on the evaluation metrics when adding channels one-by-one in order of Channel 1,2,3, and 4, (b) performance comparison of the all-channel fusion system (Channel *) with CLEAR 2006 systems and its single-channel counterparts.

Table 1: Superframe classification accuracies on the selected channels of ITC-Irst database.

	Channel 1	Channel 2	Channel 3	Channel 4
M_{bg}	84.1 %	85.1 %	86.4 %	86.6 %
M_{ev}	70.4 %	70.7 %	71.2 %	71.7 %

their details can be found in [10, 23].

Baseline systems: we compare the system's performance with three systems submitted to CLEAR 2006 [10]: *SVM*, HMM_1, and HMM_2. They correspond to UPC-D, CMU-D, and ITC-D submissions in the challenge. *SVM* and HMM_1 pursued the detection-by-classification approach for the event/background segmentation and event classification. However, while *SVM* employed SVM classifiers with RBF kernel, HMM_1 used HMM for classification. HMM_2 merged segmentation and classification in one step as performed in common ASR frameworks.

4.2. Experiment results

The superframe classification accuracies on four selected channels are listed in Table 1. In Fig. 1, the event detection results on AEER, AED-ACC, AED-ER for each individual channel are shown with different values of cutoff threshold β from 0.1 to 0.7 with a step size of 0.05. We can see that all four channels illustrate a comparable performance. On comparison with three systems *SVM*, HMM_1, and HMM_2 on AEER, it is clear that using any one of the selected channels, our detection system always beats *SVM* and HMM_1 with a large margin over all β, and it also outperforms HMM_2 with a wide range of β from 0.2 to 0.45. These results are consistent with the results in our previous work [8]. The threshold value corresponding to the best results is approximately 0.25.

The results of multi-channel fusion are illustrated in Fig. 2 with the fixed $\beta = 0.25$. As we can see in Fig. 2a, when we add an additional channel in order of Channel 1, 2, 3, and 4, we obtain better results. Specifically, AEER and AED-ER are decreasing whereas AED-ACC is increasing when the channels are added one-by-one. Fig. 2b shows the performance comparison. The multi-channel fusion system not only maintains a large margin with all the CLEAR 2006 systems but also enjoys significant improvement over the single-channel counterparts. Compared to HMM_2, the best system of CLEAR 2006, the improvement is of 10.6% on AEER. Moreover, compared the best single-channel counterpart on the Channel 2, multi-channel fusion leads to 0.7%, 1.7%, and 4.5% improvements on AEER, AED-ACC, and AED-ER, respectively.

4.3. Discussion

It is reasonable that an event can take place at any location in the room and the power of the recorded signals would be inversely proportional to the distances to the microphones. The closer microphones will most likely produce more powerful signals, allowing event recognition with higher confidence. Multi-channel fusion acts as a sum over the microphones to take advantage of the high-SNR signals and compensate the low-SNR ones. From the experiment results, one may argue that using only Channel 2 with a little bit lower performance can avoid computational overhead caused by multi-channel fusion. However, in practice we are not able to determine in advance which single channel is the best, and the events can happen at any location within the room, not favoring a specific placement of a microphone.

5. CONCLUSIONS

We proposed a multi-channel fusion framework for joint AED/C. On each individual channel, the joint problem is treated as a regression problem and subsequently addressed by class-specific random regression forests to estimate onset and offset positions of AEs in time which are measured by posterior probability output. The data fusion can be done very naturally by accumulating the posterior probabilities over all channels. The experimental results on the ITC-Irst database show that the fusion system outperforms not only the single-channel systems using common approaches in CLEAR 2006 challenge but also its single-channel counterparts.

6. REFERENCES

[1] J. Portêlo, M. Bugalho, I. Trancoso, J. Neto, A. Abad, and A. Serralheiro, "Non-speech audio event detection," in *Proc. ICASSP*, 2009, pp. 1973–1976.

[2] R. F. Lyon, "Machine hearing: An emerging field," *Signal Processing Magazine*, vol. 27, no. 5, pp. 131–139, 2010.

[3] P. K. Atrey, N. C. Maddage, and M. S. Kankanhalli, "Audio based event detection for multimedia surveillance," in *Proc. ICASSP*, 2006.

[4] J. Schröder, S. Wabnik, P. W. J. van Hengel, and S. Götze, *Ambient Assisted Living*. Springer, 2011, ch. Detection and Classification of Acoustic Events for In-Home Care, pp. 181–195.

[5] A. Temko and C. Nadeu, "Acoustic event detection in meeting-room environments," *Pattern Recognition Letters*, vol. 30, pp. 1281–1288, 2009.

[6] J. Dennis, H. D. Tran, and E. S. Chng, "Image feature representation of the subband power distribution for robust sound event classification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, no. 2, pp. 367–377, 2013.

[7] H. Phan and A. Mertins, "Exploring superframe cooccurrence for acoustic event recognition," in *Proc. EUSIPCO*, 2014.

[8] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Random regression forests for acoustic event detection and classification," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 20–31, 2015.

[9] G. Ye, I.-H. Jhuo, D. Liu, Y.-G. Jiang, D. T. Lee, and S.-F. Chang, "Joint audio-visual bi-modal codewords for video event detection," in *Proc. 2nd ACM International Conference on Multimedia Retrieval*, 2012.

[10] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, "CLEAR evaluation of acoustic event detection and classification systems," *Lecture Notes in Computer Science*, vol. 4122, pp. 311–322, 2007.

[11] R. Stiefelhagen, K. Bernardin, R. Bowers, R. T. Rose, M. Michel, and J. Garofolo, "The CLEAR 2007 evaluation," in *Multimodal Technologies for Perception of Humans*, 2009, pp. 3–34.

[12] 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. 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

[13] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[14] X. Zhou, X. Zhuang, M. Liu, H. Tang, M. Hasegawa-Johnson, and T. Huang, "HMM-based acoustic event detection with adaboost feature selection," *Lecture Notes in Computer Science*, vol. 4625, pp. 345–353, 2008.

[15] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Proc. EUSIPCO*, 2010, pp. 1267–1271.

[16] B. Ghoraani and S. Krishnan, "Time-frequency matrix feature extraction and classification of environmental audio signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2197–2209, 2011.

[17] A. Plinge, R. Grzeszick, and G. Fink, "A bag-of-features approach to acoustic event detection," in *Proc. ICASSP*, 2014, pp. 3704–3708.

[18] H. Phan and A. Mertins, "A voting-based technique for acoustic event-specific detection," in *Proc. 40th Annual German Congress on Acoustics (DAGA)*, 2014.

[19] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Acoustic event detection and localization with regression forests," in *Proc. Interspeech*, Singapore, September 2014.

[20] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu, "Regression forests for efficient anatomy detection and localization in computed tomography scans," *Medical Image Analysis*, vol. 17, no. 8, pp. 1293–1303, 2013.

[21] J. Gall, A. Yao, N. Razavi, L. van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2188–2202, 2011.

[22] A. Temko, C. Nadeu, and J.-I. Biel, "Acoustic event detection: SVM-based system and evaluation setup in CLEAR'07," *Lecture Notes in Computer Science*, vol. 4625, pp. 354–363, 2008.

[23] A. Temko, C. Nadeu, D. Macho, R. Malkin, C. Zieger, and M. Omologo, *Computers in the Human Interaction Loop*. Springer London, 2009, ch. Acoustic Event Detection and Classification, pp. 61–73.

[24] P. Giannoulis, G. Potamianos, A. Katsamanis, and P. Maragos, "Multi-microphone fusion for detection of speech and acoustic events in smart spaces," in *Proc. EUSIPCO*, 2014.

[25] L. Breiman, "Random forest," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[26] C. Zieger and M. Omologo, "Acoustic event detection - ITC-irst AED database," Internal ITC report, Tech. Rep., 2005.