Rapid Bird Activity Detection Using Probabilistic Sequence Kernels

Anshul Thakur, R. Jyothi, Padmanabhan Rajan, A.D. Dileep
School of Computing and Electrical Engineering, Indian Institute of Technology, Mandi
E-mail: anshul_thakur@students.iitmandi.ac.in, jain.jyothi.91@gmail.com, {padman,addileep}@iitmandi.ac.in

***Abstract*—Bird activity detection is the task of determining if a bird sound is present in a given audio recording. This paper describes a bird activity detector which utilises a support vector machine (SVM) with a dynamic kernel. Dynamic kernels are used to process sets of feature vectors having different cardinalities. Probabilistic sequence kernel (PSK) is one such dynamic kernel. The PSK converts a set of feature vectors from a recording into a fixed-length vector. We propose to use a variant of PSK in this work. Before computing the fixed-length vector, cepstral mean and variance normalisation and short-time Gaussianization is performed on the feature vectors. This reduces environment mismatch between different recordings. Additionally, we also demonstrate a simple procedure to speed up the proposed method by reducing the size of fixed-length vector. A speedup of almost 70% is observed, with a very small drop in accuracy. The proposed method is also compared with a random forest classifier and is shown to outperform it.**

I. INTRODUCTION

Automated acoustic monitoring of habitats is an important and useful tool for biodiversity analysis [1]. Several studies [2], [3] have shown the effectiveness of this method, when compared to traditional field studies, which are human and cost intensive. Because many birds vocalize, acoustic monitoring is particularly suited to study avian diversity in a given region. Given that it is relatively easy to collect audio recordings from the field, one must first determine which of these recordings contain a bird sound. This was the task addressed in the recently concluded bird activity detection (BAD) challenge [4], [5]. The challenge provided two datasets with audio recordings labeled as either bird (having a bird sound) and non-bird (having no bird sound.) This paper describes an efficient bird activity detector, which uses support vector machines (SVMs) using dynamic kernels.

Extracting conventional acoustic features like Mel frequency cepstral coefficients (MFCCs) from a given audio recording results in a set of feature vectors. For a given sampling rate and frame rate, the cardinality of the set depends on the duration of the audio recording. To measure the similarity between two sets of feature vectors having different cardinalities, SVMs make use of dynamic kernels [6]. In this work, we propose a variant of the probabilistic sequence kernel (PSK) [7] for bird activity detection.

Short time features like MFCCs are prone to channel and environment variations, and this can result in degradation of classifier performance. In the context of an archive of bird audio recordings, the recordings could be made using various recording devices (including automatic bioacoustic recorders, hand-held microphones, even smartphones.) The acoustic environment where these recordings are made could also be significantly different, with background sounds like humans talking, passing vehicles, wind, rain, other animals etc. To overcome some of these variations, our BAD framework utilises techniques which have been used in automatic speaker recognition. These include cepstral mean and variance normalisation, and short-time Gaussianization.

We also demonstrate a simple procedure to speed up the proposed bird activity detector. The BAD algorithm must be able to process large collections of audio recordings in a reasonable amount of time. The proposed method achieves a speedup of almost 70% with a very small drop in accuracy.

II. FEATURE EXTRACTION

In our proposed bird activity detector, MFCCs along with delta and delta-delta coefficients are used as the feature representation. Since acoustic characteristics can vary significantly in an archive of bioacoustic recordings, the difference between training and testing conditions has to be compensated. We use post-processing in the form of cepstral mean and variance normalization (CMVN) and short-time Gaussianization to mitigate the affects of mismatched conditions to some extent. Both these techniques are briefly discussed in this section.

A. Cepstral mean and variance normalization (CMVN)

The presence of channel effects due to different recording devices/conditions and convolutive noise lead to changes in the mean and variance of feature representations. These feature representations can be made robust to changes in training and testing conditions by making them zero-mean and unit-variance. The convolutive channel effects become additive in the cepstral domain. Assuming that the channel effects are stationary for a recording, the effects of the channel can be mitigated by subtracting the mean and dividing by the standard deviation [8], [9]. Here, the mean and variance are determined individually for each recording, and each feature dimension is considered independently.

A classical utterance based cepstral mean and variance normalization [9] is utilised. Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ be the set of feature vectors from an audio recording having N frames and each $\mathbf{x_n}$ is a 39-dimensional MFCC vector. $\mathbf{x_n}$(i) represents the i-th dimension for the n-th feature vector. To apply CMVN, first the mean (μ) and the variance (σ^2) are

ISBN 978-0-9928626-7-1 © EURASIP 2017

calculated across i-th dimension for all N frames, then this is processed using equation 1

$$\hat{x}_n(i) = \frac{x_n(i) - \mu(i)}{\sigma(i)}. \quad (1)$$

Here, $\hat{x}_n$(i) is the normalized value of the i-th dimension of the MFCC vector of the n-th frame. This is applied on all dimensions of the feature vectors in $\mathcal{X}$. Figure 1(b) shows the histogram of the first MFCC coefficient after applying CMVN.

B. Short-time Gaussianization

The distribution of feature vectors is also changed by the presence of channel effects and noise. Mapping this feature to an ideal distribution, like the standard normal distribution, also can provide robustness against channel effects and additive noise [10]. In short-time Gaussianization (STG), each feature dimension is treated independently and is warped so that its cumulative distribution function (CDF) matches the standard normal distribution $N(0,1)$ [10]. Let $\mathcal{X}$ be a set of features to be warped. Then STG is applied on $\mathcal{X}$ as

$$\hat{\mathcal{X}} = T(\mathcal{X}). \quad (2)$$

Here T represents a non-linear transform implementing short-time Gaussianization. A moving window of size N is used and CDF matching is applied on the central frame. The values in the moving window are sorted in descending order, and if r is the rank of the central frame, its CDF value can be approximated as [10]

$$\phi = \frac{(r - 1/2)}{N}. \quad (3)$$

The warped value, $\hat{x}$ of any feature, x should satisfy the equation

$$\phi = \int_{-\infty}^{\hat{x}} f(z)dz, \quad (4)$$

where $f(z)$ is the PDF of the standard normal distribution. Figure 1(c) shows the histogram of the first MFCC coefficient after applying short-term Gaussianization.

III. Probabilistic sequence kernel for BAD

Support vector machines using dynamic kernels deal with different cardinalities of feature sets by either matching local feature vectors in the set or by mapping a feature set on to a fixed-length representation [6]. One such dynamic kernel is the probabilistic sequence kernel (PSK) and has been utilised for speaker verification [7]. PSK was also recently utilised in bird species identification [11].

In the context of speaker verification, PSK utilizes the universal background model (UBM)-Gaussian mixture model (GMM) framework. For the task of BAD, we use a variant of PSK which utilises a single GMM instead of a UBM-GMM. A GMM is built using the examples of bird class only. Suppose $\mathcal{X} = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}\}$ is a set of feature vectors. Then, the probabilistic alignment vector, $\boldsymbol{\Psi(x_i)}$, for feature vector $\mathbf{x_i}$ is given as $\boldsymbol{\Psi}(\mathbf{x_i}) = [\gamma_1(\mathbf{x_i}), \gamma_2(\mathbf{x_i}), ...\gamma_Q(\mathbf{x_i})]^T$. Here, Q is

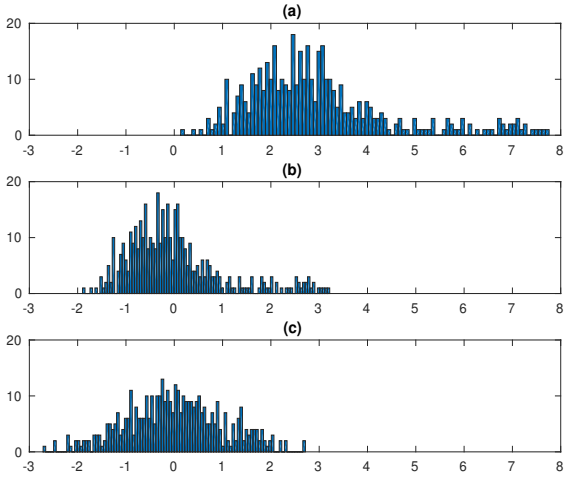

Fig. 1. Histogram of first MFCC coefficient extracted from a song recording of Cassin's Vireo (a) before pre-processing (b) after applying CMVN (c) after applying CMVN and short-time Gaussianization.

the number of components in the GMM and $\gamma_q(\mathbf{x_i})$ represents the probabilistic alignment of $\boldsymbol{x_i}$ with the q-th component, and is calculated as

$$\gamma_q(\mathbf{x_i}) = \frac{w_q \mathcal{N}(\mathbf{x_i}|\mu_q, \mathbf{\Sigma}_q)}{\sum_{j=1}^{Q} w_j \mathcal{N}(\mathbf{x_i}|\mu_q, \mathbf{\Sigma}_q)}. \quad (5)$$

Here w_q, μ_q and $\mathbf{\Sigma}_q$ represent weight, mean and covariance of q-th component of the GMM.

The set, $\mathcal{X}$, of feature vectors (and hence the audio recording) is represented as a fixed-length vector $\mathbf{\Phi}_{\text{PSK}}(\mathcal{X})$, defined as

$$\mathbf{\Phi}_{\text{PSK}}(\mathcal{X}) = \frac{1}{N} \sum_{n=1}^{N} \mathbf{\Psi}(\mathbf{x_n}). \quad (6)$$

The length of $\mathbf{\Phi}_{\text{PSK}}(\mathcal{X})$ is Q. The probabilistic sequence kernel between two feature sets i.e $\mathcal{X}_a$ and $\mathcal{X}_b$ is defined using equation 7

$$K_{\text{PSK}}(\mathcal{X}_a, \mathcal{X}_b) = \mathbf{\Phi}_{\text{PSK}}(\mathcal{X}_a)^T \mathbf{S}^{-1} \mathbf{\Phi}_{\text{PSK}}(\mathcal{X}_b). \quad (7)$$

Here S is a correlation matrix defined as

$$\mathbf{S} = \frac{1}{Z} \mathbf{R}^T \mathbf{R}. \quad (8)$$

$\mathbf{R}$ is a $Z \times Q$ matrix having rows which are the probabilistic alignment vectors from the feature vectors of the training set having Z training examples. Using $\mathbf{\Phi}_{\text{PSK}}$ vectors of bird and non-bird recordings, an SVM learns support vectors to discriminate between the two classes.

Since the GMM is built using only bird class, the responsibility terms for some of the components are significantly different for bird and non-bird recordings, providing distinction between $\mathbf{\Phi}_{\text{PSK}}$ representations of both the classes. Figure 2 shows the framework based on PSK for bird activity detection.

ISBN 978-0-9928626-7-1 © EURASIP 2017

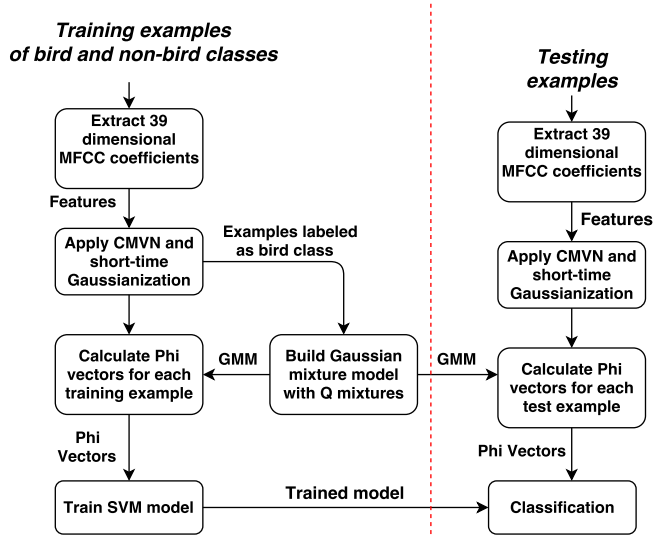

Fig. 2. Proposed PSK based framework for BAD

Algorithm 1: Proposed procedure for choosing relevant P components for calculating Φ_{PSK} vectors

- Randomly choose K audio recordings which are labeled as bird activity from the training dataset (K is much smaller than the number of training examples).
- Segment bird calls from each recording using weighted inverse spectral flatness (ISF) and thresholding as described in [12].
- Calculate probabilistic alignment vectors for each segment.
- Choose highest 30 responsibility terms along with their index from each vector.
- Calculate frequency of each component index pooled together in the previous step.
- Choose P component indexes having maximum frequencies to compute $\mathbf{\Phi}_{\text{PSK}}$.

IV. Improving Computational Efficiency

In the proposed framework, as discussed in the previous section, a GMM built using bird class examples is used for calculating probabilistic alignment vectors. Audio recordings labeled as bird may also contain other background sounds, including silence regions. Hence, every component of the GMM need not correspond to bird sounds. This observation can be exploited to bring down the size of the probabilistic alignment vectors and hence the size of $\mathbf{\Phi}_{\text{PSK}}$ vectors. Instead of using all Q components for calculating probabilistic alignment vectors, only P components can be used, such that $P < Q$.

The computational complexity of the proposed framework is directly dependent on mapping a recording to a $\mathbf{\Phi}_{\text{PSK}}$ vector. Hence, this complexity is also dependent on calculating responsibility terms for each component of the GMM. By using only P relevant components, the computational complexity required to calculate the $\mathbf{\Phi}_{\text{PSK}}$ vector for any feature set is $O(N \times P)$ instead of $O(N \times Q)$. Here N is the number of feature vectors in any feature set and $P < Q$.

The classification accuracy can still be maintained if these P components correspond to the bird-calls and not to the background. The procedure to choose these P components is described in algorithm 1. For a given GMM, this is a one-time process. Since responsibility terms are calculated only for segmented bird sounds not the background, it is most likely that the top components chosen using algorithm 1 will correspond to bird sounds.

In this work, we have considered $K = 15$ randomly chosen recordings to choose P components. One can use the entire training set to choose P components. However, our experimentation showed that the same results are obtained even for a small number of recordings from the training set.

The bar plots of $\mathbf{\Phi}_{\text{PSK}}$ representations for a bird and a non-bird recording calculated using $P = 8$ and $P = 16$ GMM components (estimated using algorithm 1) instead of $Q = 128$ components are depicted in Figure 3 and Figure 4. By analyzing these figures, it is clear that the magnitude of responsibility terms of some of the components for bird and non-bird recordings are different. This difference in responsibility terms leads to the distinction between two classes.

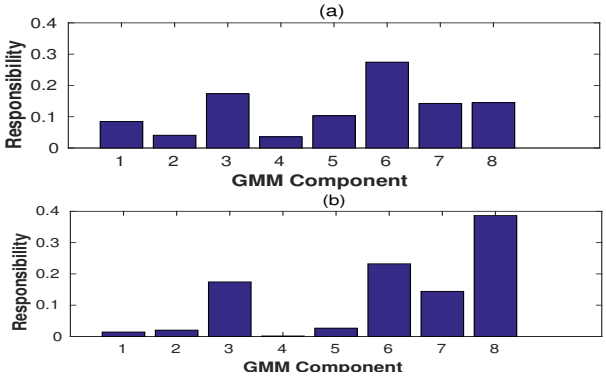

Fig. 3. Bar plots of $\mathbf{\Phi}_{\text{PSK}}$ representations calculated using $P = 8$ for (a) a bird recording (b) a non-bird recording.

V. Experiments and Results

A. Datasets Used

The proposed BAD framework using all Q GMM components and using only the top P components are evaluated on data that was released as part of the BAD challenge [4]. The data is from two sources: Freefield and Warblr. Freefield recordings are collected by the Freesound project [13]. The data consists of 1935 and 5755 recordings labeled as bird and non-bird respectively. Warblr [14] is UK-based bird sound crowd-sourcing research project. A subset of Warblr having 6045 bird and 1955 non-bird recordings is provided. Both

ISBN 978-0-9928626-7-1 © EURASIP 2017

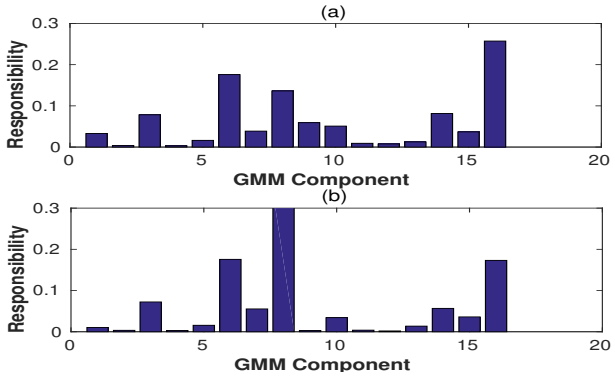

Fig. 4. Bar plots of $\mathbf{\Phi}_{\text{PSK}}$ representations calculated using $P = 16$ for (a) a bird recording (b) a non-bird recording.

datasets are collected in various environments and exhibits different background sounds. Each audio recording is 10 seconds long and has a sampling rate of 44.1 kHz.

B. Experimental setup

To evaluate the generalization of the proposed BAD system, training and testing is done on different datasets. In other words, when Warblr is used for training, Freefield is used as test, and vice versa. For feature extraction, a frame size of 20 ms with no overlap is used. This is done to reduce the number of frames for processing. The GMM is built using 100 randomly chosen examples from the bird class. The number of components in the GMM, Q is set to 128. These parameters are determined by utilising a small test set of 2000 examples. Varying these parameters did not result in major performance gains (see Table I). In general, the more the data used for building the GMM, the better is the estimate of the probabilistic alignment vectors. Moreover, the non-application of CMVN and STG resulted in a performance degradation ranging from 5 to 11%.

The SVM is trained using $\mathbf{\Phi}_{\text{PSK}}$ vectors derived from 200 examples each of the bird and the non-bird classes. LIBSVM [15] is used for SVM implementation and Voicebox [16] is used for MFCC extraction. Accuracy i.e. the percentage of correctly classified examples is used as the performance metric.

The performance of the proposed BAD system is compared with a random forest classifier with 128 trees. Random forest based approach is a baseline method considered in the BAD challenge. In this work, the random forest is trained on $\mathbf{\Phi}_{\text{PSK}}$ vectors derived from the GMM. The accuracy of this method is compared with that of the proposed approach in Table II.

The results demonstrate that the proposed PSK-based BAD system discriminates recordings having bird sounds with those that do not. Since the GMM is built using only recordings labeled as bird, examples from this class align better with most of the components.

TABLE I
PERFORMANCE OF THE PROPOSED BAD FRAMEWORK ON 2000 TEST EXAMPLES FOR DIFFERENT GMM COMPONENTS (Q) AND DIFFERENT NUMBER OF FILES FOR BUILDING THE GMM.

Training dataset	Testing dataset	Files used for building GMM	Components Used (Q)	Accuracy (%)
Warblr	Freefield	100	64	71.95
			128	**73.9**
			256	73.87
		500	64	73.3
			128	73.6
			256	**73.96**
		1000	64	74.01
			128	74.25
			256	**74.75**
Freefield	Warblr	100	64	75.4
			128	**76.1**
			256	75.9
		500	64	73.15
			128	75.1
			256	**75.4**
		1000	64	74.3
			128	**75.2**
			256	74.89

TABLE II
COMPARISON OF PERFORMANCES OF THE PROPOSED FRAMEWORK WITH RANDOM FOREST CLASSIFIER AND SVM WITH LINEAR KERNEL

Training	Testing	Random Forest (%)	SVM with proposed PSK with Q components (%)	Proposed PSK with P=32 components (%)
Warblr	Freefield	79.35	**85.01**	84.85
Freefield	Warblr	72.14	**77.15**	76.9

C. Using top P components

By choosing the top P scoring components, the computation requirement of GMM-based PSK is further decreased. The P components having high probability of corresponding to bird calls are chosen using algorithm 1. We use different values for P to find a configuration which provides comparable accuracy but takes significantly less computational time as compared to using all the Q GMM components.

To evaluate the performance and computation time trade-off, we use Warblr dataset for training and Freefield dataset for testing. Figure 5 depicts the accuracy and running time comparison for different values of P i.e. 8, 16, 32 and 64. The running time for both is measured on a computer having Intel i7 5th generation quad core processor and 16 GB of RAM. The running time shown in Figure 5 is the average time taken for ten runs on the complete test dataset.

From Figure 5, it is evident that the classification accuracies for $P = 32$, 64 and 128 components are essentially equivalent. However, it is clear that the average running times for 32 GMM components is 1593 seconds, for 64 components is 2836 seconds and for 128 components is 5694 seconds. Therefore, the average running time using 32 components is

ISBN 978-0-9928626-7-1 © EURASIP 2017

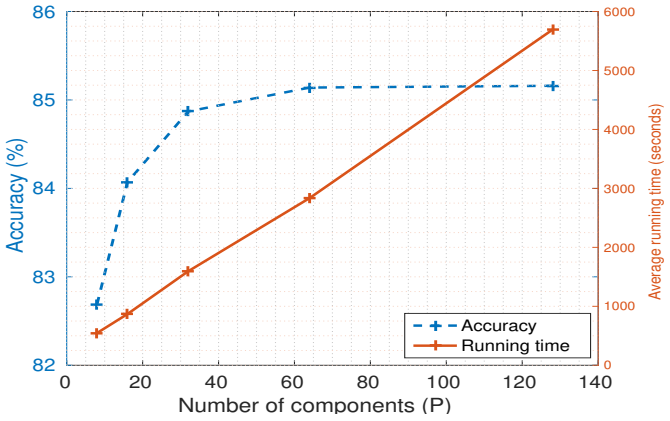

Fig. 5. Comparison of classification accuracies and running time (seconds) for different number of chosen components, P

almost 43% less than 64 components and 70% less than the 128 components. Hence, using only P components improves running time significantly, with a small drop in accuracy for lower values of P. This is useful in the context of searching through a large volume of recordings.

VI. CONCLUSION

This paper described a bird activity detector using a variant of the probabilistic sequence kernel. By utilising probabilistic alignment vectors derived from recordings that contain birdcalls and from ones which do not, the SVM is able to distinguish the two classes effectively. Moreover, by using only a subset of the components of the probabilistic alignment vector, considerable speedup was obtained, with a very small drop in accuracy. The method illustrates how converting a set of feature vectors into a fixed-length representation can be effective in discriminating classes. The method can also be applied to discriminate recordings of different durations.

Although this paper utilised only the probabilistic sequence kernel, several other dynamic kernels can be utilised [11]. Future work will investigate the use of these kernels.

VII. ACKNOWLEDGEMENT

This work is partially supported by IIT Mandi under the project IITM/SG/PR/39 and Science and Engineering Research Board, Govt. of India under the project SERB/F/7229/2016-2017.

REFERENCES

[1] T. S. Brandes, "Automated sound recording and analysis techniques for bird surveys and conservation," *Bird Conservation International*, vol. 18, no. S1, pp. S163–S173, 2008.

[2] A. L. Borker, M. W. McKown, J. T. Ackerman, C. A. EAGLES-SMITH, B. R. Tershy, and D. A. Croll, "Vocal activity as a low cost and scalable index of seabird colony size," *Conservation biology*, vol. 28, no. 4, pp. 1100–1108, 2014.

[3] B. J. Furnas and R. L. Callas, "Using automated recorders and occupancy models to monitor common forest birds across a large geographic region," *The Journal of Wildlife Management*, vol. 79, no. 2, pp. 325–337, 2015.

[4] "BAD challenge," http://machine-listening.eecs.qmul.ac.uk/bird-audio-detection-challenge/, accessed: 2017-2-1.

[5] D. Stowell, M. Wood, Y. Stylianou, and H. Glotin, "Bird detection in audio: a survey and a challenge," in *IEEE Int. Workshop Mach. Learn. Sig. Process.*, 2016, pp. 1–6.

[6] A. D. Dileep and C. C. Sekhar, "GMM-based intermediate matching kernel for classification of varying length patterns of long duration speech using support vector machines," *IEEE Trans. Neural net. learn. sys.*, vol. 25, no. 8, pp. 1421–1432, 2014.

[7] K.-A. Lee, C. You, H. Li, and T. Kinnunen, "A gmm-based probabilistic sequence kernel for speaker verification." in *Proc. Interspeech*, 2007.

[8] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 29, no. 2, pp. 254–272, 1981.

[9] N. V. Prasad and S. Umesh, "Improved cepstral mean and variance normalization using bayesian framework," in *Proc. IEEE Workshop Auto. Speech Recogn. Understand.*, 2013.

[10] B. Xiang, U. Chaudhari, J. Navrátil, G. N. Ramaswamy, and R. A. Gopinath, "Short-time Gaussianization for robust speaker verification," in *Proc. Int. Conf. Acoust. Speech, Signal Process*, 2002.

[11] D. Chakraborty, P. Mukker, P. Rajan, and A. Dileep, "Bird call identification using dynamic kernel based support vector machines and deep neural networks," in *Proc. Int. Conf. Mach. Learn. App.*, 2016.

[12] A. Thakur and P. Rajan, "Model-based unsupervised segmentation of birdcalls from field recordings," in *Proc. Int. Conf. Signal Process. Commun. Syst.*, 2016.

[13] "Freesound," http://freesound.org/, accessed: 2016-07-10.

[14] "Warblr," https://warblr.net/, accessed: 2017-2-1.

[15] "LIBSVM," https://www.csie.ntu.edu.tw/cjlin/libsvm/, accessed: 2017-2-1.

[16] "Voicebox," http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html, accessed: 2017-2-1.