Bird Call Identification using Dynamic Kernel based Support Vector Machines and Deep Neural Networks

Deep Chakraborty*, Paawan Mukker†, Padmanabhan Rajan† and A. D. Dileep†
*Department of Electronics and Communication Engineering
Manipal Institute of Technology, Manipal, Karnataka, India 576104
†School of Computing and Electrical Engineering
Indian Institute of Technology Mandi, Kamand, Himachal Pradesh, India 175001

***Abstract*—In this paper, we apply speech and audio processing techniques to bird vocalizations and for the classification of birds found in the lower Himalayan regions. Mel frequency cepstral coefficients (MFCC) are extracted from each recording. As a result, the recordings are now represented as varying length sets of feature vectors. Dynamic kernel based support vector machines (SVMs) and deep neural networks (DNNs) are popularly used for the classification of such varying length patterns obtained from speech signals. In this work, we propose to use dynamic kernel based SVMs and DNNs for classification of bird calls represented as sets of feature vectors. Results of our studies show that both approaches give comparable performance.**

I. Introduction

Birds serve as important indicators of ecosystem health. Traditional field techniques to track and identify different bird species have required much human effort. Automatic analysis of bird call recordings have recently become popular [1]. Reliable techniques for tasks such as species identification allow scientists and ecologists to analyze long recordings obtained from the field. Several techniques applied to the processing of speech and audio signals can be applied to bird calls. This paper focuses on identification of bird species from their calls. As in any pattern recognition task, the challenges in bird call identification is in the choice of the features and in the choice of the classifiers. Several works in the literature show that spectral features like Mel frequency cepstral coefficients (MFCC) are widely used to represent bird sounds [2], [3] and classifiers such as Gaussian mixture model (GMM) [4], [5] and support vector machines [6] are commonly used.

The main focus of this work is to explore state-of-the-art techniques applied to speech and audio signals for representing bird calls and for their classification. As in speech signal processing, short-time analysis of bird call signal involves performing spectral analysis on each frame of about 20 milliseconds duration and representing each frame by a real valued feature vector. The bird call signal of an utterance with T frames is represented as a sequential pattern $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t, \ldots, \mathbf{x}_T)$, where $\mathbf{x}_t$ is a feature vector for frame t. The duration of the calls (either of the same species or across different species) varies from one recording to another. Hence, the number of frames also differs from one recording to another. In the tasks such as classification of bird phrase [7] there is a need to model the temporal dynamics and correlations among the features in the sequence of feature vectors. However, in the task such as bird call identification, preserving sequence information is not critical. In such cases, the bird call signal is represented as a set of feature vectors.

In this regard, we utilise commonly used techniques for the classification of varying length patterns of speech and audio signals that are represented as sets of continuous valued feature vectors in bird call identification. Conventionally, Gaussian mixture models (GMMs) [8] are used for classification of varying length patterns represented as sets of feature vectors. The maximum likelihood (ML) based method is commonly used for the estimation of the parameters of the GMM for each class. When the amount of the training data available per class is limited, robust estimates of the model parameters can be obtained through maximum a posteriori adaptation (MAP) of the universal background model (UBM), to the training data of each class [9]. The UBM is a large GMM trained using the training data of all classes.

Classification of varying length sets of feature vectors using support vector machines (SVMs) requires design of a suitable kernel as a measure of similarity between a pair of sets of feature vectors. The kernels designed for varying length patterns are referred to as dynamic kernels [10]. Probabilistic sequence kernel [11], GMM supervector kernel [12], GMM-UBM mean interval kernel [13], GMM-based intermediate matching kernel [10] and GMM-based pyramid match kernel [14] are some of the state-of-the-art dynamic kernels for sets of feature vectors. Their effectiveness has been shown in the tasks such as speaker identification and speech emotion recognition [10], [12], [13]. In this work, we show the effectiveness of these dynamic kernel based SVMs for classifying bird calls.

In recent years, deep learning techniques are setting new standards in different tasks related to speech data. Recent works on fully connected deep neural networks (DNNs) [15], [16] are shown to outperform traditional baseline systems in the tasks such as speaker identification, language identification and speech recognition. In this work, we explore the effectiveness of fully connected DNNs for the classification of bird calls in similar lines as they are used for speaker identification and language identification tasks.

The paper makes following contributions. First, we explore Mel frequency cepstral coefficients (MFCCs) and logarithm of the Mel filterbank energy coefficients (log MFECs) to represent an audio recording as a set of feature vectors. Secondly, we explore the effectiveness of state-of-the-art techniques in speech technology such as GMMs, dynamic kernel based SVMs and DNNs for the identification of bird species from their calls.

The rest of this paper is organized as follows. In Section II, a review of dynamic kernels for sets of feature vectors is presented. Details of fully connected deep neural networks are presented in Section III. The database and features used in our experiments are given in Section IV. Studies on bird call identification are presented in Section V. We conclude in Section VI.

II. Dynamic kernels for sets of feature vectors

In this section, we review the approaches to design dynamic kernels for varying length patterns represented as sets of feature vectors. Different approaches to design dynamic kernels are broadly divided into explicit mapping based approaches and matching based approaches [10].

A. Explicit mapping based approaches

These approaches involve mapping a set of feature vectors onto a fixed-dimensional representation and then defining a kernel function in the space of that representation. In this work we propose to explore probabilistic sequence kernel (PSK) [11], GMM supervector (GMMSV) kernel [12] and GMM-UBM mean interval (GUMI) kernel [13] as the dynamic kernels for sets of feature vectors constructed using the explicit mapping based approaches.

1) Probabilistic sequence kernel: Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ be a set of feature vectors. Probabilistic sequence kernel (PSK) [11] maps a set of feature vectors onto a probabilistic feature vector obtained using GMMs. The PSK uses UBM with Q components [9] and the class-specific GMMs obtained by adapting the UBM. A feature vector $\mathbf{x}$ is represented in a higher dimensional feature space as a vector of responsibility terms of the $2Q$ components (Q from a class-specific adapted GMM and Q from UBM) as $\mathbf{\Psi}(\mathbf{x}) = [\gamma_1(\mathbf{x}), \gamma_2(\mathbf{x}), ..., \gamma_{2Q}(\mathbf{x})]^\top$. Since the element $\gamma_q(\mathbf{x})$ indicates the probabilistic alignment of $\mathbf{x}$ to the qth component, $\mathbf{\Psi}(\mathbf{x})$ is called the probabilistic alignment vector. A set of local feature vectors $\mathbf{X}$ is represented as a fixed dimensional vector $\mathbf{\Phi}_{\text{PSK}}(\mathbf{X})$, given by

$$\mathbf{\Phi}_{\text{PSK}}(\mathbf{X}) = \frac{1}{T}\sum_{t=1}^{T}\mathbf{\Psi}(\mathbf{x}_t) \tag{1}$$

The dimension of $\mathbf{\Phi}_{\text{PSK}}(\mathbf{X})$ is D=$2Q$. Then, the PSK between two examples $\mathbf{X}_m = \{\mathbf{x}_{m1}, \mathbf{x}_{m2}, ..., \mathbf{x}_{mT_m}\}$ and $\mathbf{X}_n = \{\mathbf{x}_{n1}, \mathbf{x}_{n2}, ..., \mathbf{x}_{nT_n}\}$ is given as

$$K_{\text{PSK}}(\mathbf{X}_m, \mathbf{X}_n) = \mathbf{\Phi}_{\text{PSK}}(\mathbf{X}_m)^\top \mathbf{S}^{-1} \mathbf{\Phi}_{\text{PSK}}(\mathbf{X}_n) \tag{2}$$

The correlation matrix $\mathbf{S}$ is a $D \times D$ matrix and is defined as follows:

$$\mathbf{S} = \frac{1}{M}\mathbf{R}^\top\mathbf{R} \tag{3}$$

where $\mathbf{R}$ is the $M \times D$ matrix whose rows are the probabilistic alignment vectors for the feature vectors of all examples in the training data set and M is the total number of feature vectors in the training data set.

2) GMM supervector kernel: The GMM supervector (GMMSV) kernel [12] performs a mapping of a set of feature vectors onto a higher dimensional vector corresponding to a GMM supervector. An example-specific adapted GMM is built for each example by adapting the means of the UBM using the data of that example. Let $\boldsymbol{\mu}_q^{(\mathbf{X})}$ be the mean vector of qth component in the example-specific adapted GMM for an example $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$. A GMM vector $\mathbf{\Psi}_q(\mathbf{X})$ for an example $\mathbf{X}$ corresponding to the qth component of GMM is obtained as follows:

$$\mathbf{\Psi}_q(\mathbf{X}) = \left[\sqrt{w_q}\mathbf{\Sigma}_q^{-\frac{1}{2}}\boldsymbol{\mu}_q^{(\mathbf{X})}\right]^\top \tag{4}$$

where, w_q and $\mathbf{\Sigma}_q$ are the mixture coefficient and covariance matrix of qth component in UBM. The GMM supervector for the example $\mathbf{X}$ is given by

$$\mathbf{\Phi}_{\text{GMMSV}}(\mathbf{X}) = [\mathbf{\Psi}_1(\mathbf{X})^\top, \mathbf{\Psi}_2(\mathbf{X})^\top, ..., \mathbf{\Psi}_Q(\mathbf{X})^\top]^\top \tag{5}$$

The dimension of GMM supervector is D=Qd, where Q is the number of components in UBM and d is the dimension of the feature vector. The GMMSV kernel between a pair of examples $\mathbf{X}_m$ and $\mathbf{X}_n$ is given by

$$K_{\text{GMMSV}}(\mathbf{X}_m, \mathbf{X}_n) = \mathbf{\Phi}_{\text{GMMSV}}(\mathbf{X}_m)^\top \mathbf{\Phi}_{\text{GMMSV}}(\mathbf{X}_n) \tag{6}$$

3) GMM-UBM mean interval kernel: The GMM-UBM mean interval (GUMI) kernel [13] performs a mapping of a set of local feature vectors onto a higher dimensional vector corresponding to a GUMI supervector. In GUMI kernel, an example-specific adapted GMM is built for each example by adapting the mean vectors and covariance matrices of the UBM using the data of that example. Let $\boldsymbol{\mu}_q^{(\mathbf{X})}$ and $\mathbf{\Sigma}_q^{(\mathbf{X})}$ be the mean vector and the covariance matrix of qth component in the example-specific adapted GMM for an example $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$. A GUMI vector $\mathbf{\Psi}_q(\mathbf{X})$ for an example $\mathbf{X}$ corresponding to the qth component of GMM is obtained as follows:

$$\mathbf{\Psi}_q(\mathbf{X}) = \left(\frac{\mathbf{\Sigma}_q^{(\mathbf{X})} + \mathbf{\Sigma}_q}{2}\right)^{-\frac{1}{2}} \left(\boldsymbol{\mu}_q^{(\mathbf{X})} - \boldsymbol{\mu}_q\right) \tag{7}$$

where, $\boldsymbol{\mu}_q$ and $\mathbf{\Sigma}_q$ are the mean vector and covariance matrix of qth component in UBM. The GUMI supervector is obtained by concatenating the GUMI vectors of different components as

$$\mathbf{\Phi}_{\text{GUMI}}(\mathbf{X}) = [\mathbf{\Psi}_1(\mathbf{X})^\top, \mathbf{\Psi}_2(\mathbf{X})^\top, ..., \mathbf{\Psi}_Q(\mathbf{X})^\top]^\top \tag{8}$$

The dimension of GUMI supervector is D=Qd. The GUMI kernel between a pair of examples $\mathbf{X}_m$ and $\mathbf{X}_n$ is given by

$$K_{\text{GUMI}}(\mathbf{X}_m, \mathbf{X}_n) = \mathbf{\Phi}_{\text{GUMI}}(\mathbf{X}_m)^\top \mathbf{\Phi}_{\text{GUMI}}(\mathbf{X}_n) \tag{9}$$

B. Matching based approaches

Given two sets of feature vectors, the matching based approach computes the kernel function by matching individual feature vectors from these sets.

In this work, we propose to use GMM-based intermediate matching kernel [10] and GMM-based pyramid match kernel [14] as the dynamic kernels designed using the matching based approaches.

1) GMM-based intermediate matching kernel: An intermediate matching kernel (IMK) [17] is constructed by matching the two sets of feature vectors using a set of virtual feature vectors. For every virtual feature vector, a feature vector is selected from each set of feature vectors and a base kernel for the two selected feature vectors is computed. The IMK for a pair of sets of feature vectors is computed as a combination of these base kernels. In [10], the set of virtual feature vectors considered are in the form of the components of the UBM. For every component of the UBM, a feature vector each from the two sets of feature vectors, that has the highest probability of belonging to that component (i.e., value of responsibility term) is selected. Then a base kernel is computed between the selected feature vectors. The responsibility of qth component for a feature vector $\mathbf{x}$, $\gamma_q(\mathbf{x})$, is given as

$$\gamma_q(\mathbf{x}) = \frac{w_q \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)}{\sum_{j=1}^{Q} w_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \tag{10}$$

where w_q is the mixture coefficient of the component q, and $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$ is the normal density for the component q with mean vector $\boldsymbol{\mu}_q$ and covariance matrix $\boldsymbol{\Sigma}_q$. The feature vectors $\mathbf{x}^*_{mq}$ and $\mathbf{x}^*_{nq}$ respectively in $\mathbf{X}_m$ and $\mathbf{X}_n$, are selected using the component q as

$$\mathbf{x}^*_{mq} = \arg\max_{\mathbf{x}\in\mathbf{X}_m} \gamma_q(\mathbf{x}) \quad \text{and} \quad \mathbf{x}^*_{nq} = \arg\max_{\mathbf{x}\in\mathbf{X}_n} \gamma_q(\mathbf{x}) \tag{11}$$

The GMM-based IMK is computed as the sum of the values of the base kernels computed for the Q pairs of selected feature vectors as follows:

$$K_{\text{GMMIMK}}(\mathbf{X}_m, \mathbf{X}_n) = \sum_{q=1}^{Q} k(\mathbf{x}^*_{mq}, \mathbf{x}^*_{nq}) \tag{12}$$

The Gaussian kernel $k(\mathbf{x}^*_{mq}, \mathbf{x}^*_{nq}) = \exp(-\delta\|\mathbf{x}^*_{mq} - \mathbf{x}^*_{nq}\|^2)$ is used as the base kernel. Here δ is the width parameter of the Gaussian kernel that is empirically chosen.

2) GMM-based pyramid match kernel: In the pyramid match kernel (PMK), a set of feature vectors is mapped onto a multi-resolution histogram pyramid. The kernel is computed between a pair of examples by matching the pyramids using a weighted histogram intersection match function at each level of the pyramid. In [14], the UBMs built with increasingly larger number of components are used to construct the histograms at the different levels in the pyramid. At level l, a UBM of $Q = b^l$ components is built using the feature vectors in the training examples of all the classes. Here, b is considered as branching factor. The histogram vectors $\mathbf{h}_l(\mathbf{X}_m) = [h_{l1}(\mathbf{X}_m), h_{l2}(\mathbf{X}_m), \ldots, h_{lQ}(\mathbf{X}_m)]^\top$ and $\mathbf{h}_l(\mathbf{X}_n) = [h_{l1}(\mathbf{X}_n), h_{l2}(\mathbf{X}_n), \ldots, h_{lQ}(\mathbf{X}_n)]^\top$ with Q-dimensions, corresponding to the sets of feature vectors $\mathbf{X}_m$ and $\mathbf{X}_n$, is then obtained by soft quantization. A histogram intersection kernel, $K^{(l)}_{\text{HIK}} = \sum_{q=1}^{Q} \min\left(h_{lq}(\mathbf{X}_m), h_{lq}(\mathbf{X}_n)\right)$ is then computed to obtain the number of matches between a pair of histogram vectors corresponding to a pair of examples $\mathbf{X}_m$ and $\mathbf{X}_n$ at each level, $l = 0, 1, \ldots, L$. Here, L is the total number of levels in the pyramid. The matching is a hierarchical process from the bottom of the pyramid to the top of the pyramid. The number of new matches at a level l is calculated by computing the difference between the number of matches at that level and the number of matches at its immediately higher level and is given by $K^{(l)}_{\text{HIK}}(\mathbf{X}_m, \mathbf{X}_n) - K^{(l+1)}_{\text{HIK}}(\mathbf{X}_m, \mathbf{X}_n)$. The number of new matches at each level is weighted according to the number of components of UBM at that level. The GMM-based PMK between a pair of examples is computed as a weighted sum of the number of new matches at different levels of the pyramid and is given as,

$$K_{\text{PMK}}(\mathbf{X}_m, \mathbf{X}_n) = \sum_{l=0}^{L-1} \frac{1}{b^{L-l}} (K^{(l)}_{\text{HIK}} - K^{(l+1)}_{\text{HIK}}) + K^{(L)}_{\text{HIK}} \tag{13}$$

In our experiments, we compare the performance of the SVM-based classifiers using the kernels reviewed in this section.

III. Deep neural networks

In this section, we briefly describe fully connected deep neural networks (DNNs) that are commonly used for the tasks on speech signals.

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ be a set of feature vectors corresponding to a speech signal and each $\mathbf{x}_t$ corresponding to a spectral feature extracted from tth frame. In the speech domain it is common that the input feature vector to the DNN corresponding to $\mathbf{x}_t$ is the supervector of contextual vectors around $\mathbf{x}_t$. The input feature vectors to the DNN is obtained by stacking every d-dimensional feature vectors $\mathbf{x}_t$ by l contextual vectors to the left and r contextual vectors to the right. Thus, the total number of stacked frames is $l + r + 1$. Therefore, the dimension of input feature vectors to the DNN is $D = d(l+r+1)$ corresponding to every frame $\mathbf{x}_t$. (For the initial and final few feature vectors, some form of padding or repetition is used.) Thus there are D visible units in the input layer of the DNN. There will be J hidden layers and each hidden layer contains k units with a rectified linear unit (ReLU) activation [15]. The output layer is the softmax output layer, with one output for each class. Typically, J varies from 2 to 3 and k varies from 256 to 1024 for typical speech tasks [15], [16]. For each input vector corresponding to $\mathbf{x}_t$, the DNN outputs a score $f_c(\mathbf{x}_t)$ for each class c. For any test example, the class label is decided as

$$\arg\max_c \sum_{t=1}^{T} f_c(\mathbf{x}_t) \tag{14}$$

Figure 1 shows the complete topology of the baseline fully connected DNN. Stochastic gradient descent with error back-

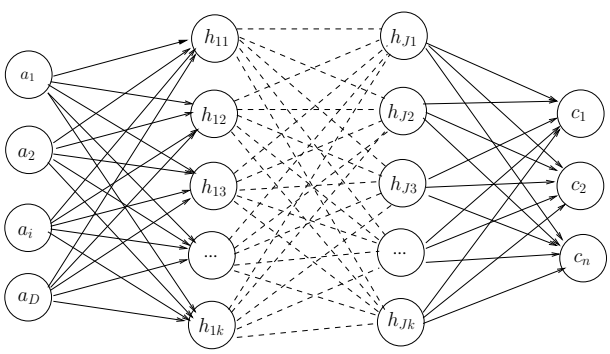

Fig. 1. DNN topology with D input units, J hidden layers with k units in each layer and n softmax output units.

propagation [15], [16] is used for training the DNN. Initialization of the weights have to be done with care [16]. In this work, we use supervised pre-training for this purpose [18]. The idea of pre-training is to learn one layer of weights at a time with the outputs in one layer acting as the input for training the next layer. After this pre-training, the multiple layers of weights can be used as a much better starting point for a fine-tuning phase during which back-propagation through the DNN slightly adjusts the weights found in pre-training.

IV. Databases and features used in the studies

The dataset used in our study was collected at the Great Himalayan National Park, situated in the lower Himalayas, in north India. The recordings were collected manually using a directional microphone and were labelled with the species by experienced birdwatchers. The same equipment was used in all the recordings, and the recordings have no overlapping calls.

Recordings from 26 different passerine species were used in this study. The durations of the recordings varied from 86 seconds to 15 seconds and the average duration was about 40 seconds per species. Long recordings were segmented into individual calls by analyzing the amplitude of the samples. These served as training and testing examples. In order to avoid biasing towards any class having larger number of training examples, we have considered about 14 seconds of data for the training of each class and the remaining data were used for testing. This leads to a total of 232 training examples and the test set includes a total of 329 examples. Table I shows the details of the database used for the study. The evaluation metric is the identification accuracy obtained on the test set.

39-dimensional Mel frequency cepstral coefficients (MFCCs) consisting of 12 base coefficients, 1 log-energy and their corresponding delta and acceleration coefficients were utilised as features. The MFCCs are extracted from 32 Mel filterbanks. A frame size of 20 ms and a shift of 10 ms are used. These features are used to build Gaussian mixture models (GMMs), dynamic kernel based SVMs and DNNs. Additionally, the logarithm of the Mel filterbank energy coefficients (log MFEC) have also been utilised as inputs to DNNs in speech tasks [15]. We also consider these 32-dimensional features from every frame for building the DNNs.

In this study, we consider 7 contextual vectors to the left and 7 contextual vectors to the right of every tth frame. This makes the total number of stacked frames in the DNN as 15 and hence, the dimension of the input feature vectors is $D = d * 15$ corresponding to every frame. The d here may be either 39 or 32, depending upon the features considered.

V. Experimental studies on bird call identification

In this section, we first study the effectiveness of the dynamic kernels for bird call identification using SVM-based classifiers.

We consider SVMTorch [19] tool to build the SVM-based classifiers. In this study, one-against-the-rest approach is considered for the 26-class bird call identification task. The value of trade-off parameter, C in the SVM is chosen empirically. In this work, the best results are observed for $C = 0.001$.

TABLE I. Database of bird calls. The total training data for each species is approximately 14 seconds long.

Bird Species	Number of samples for training	Number of samples for test
Lesser Cuckoo	7	3
Black Throated Tit	8	12
Black and Yellow Grosbeak	10	12
Blackcrested Tit	5	9
Chestnut-crowned Laughingthrush	8	11
Eurasian Treecreeper	9	5
Golden Bushrobin	10	14
Great Barbet	10	20
Grey Bellied Cuckoo	10	7
Grey Bushchat	8	9
Greyhooded Warbler	7	3
Greywinged Blackbird	4	6
Himalayan Monal	11	25
Large-billed Crow	7	4
Orange-flanked Bushrobin	8	10
Oriental Cuckoo	9	7
Pale-rumped Warbler	6	6
Rock Bunting	6	7
Rufous-gorgetted Flycatcher	9	8
Rufous-bellied Niltava	9	9
Russet-backed Sparrow	14	38
Spotted Nutcracker	19	31
Streaked Laughingthrush	9	4
Western Tragopan	8	5
White-cheeked Nuthatch	10	52
Yellow-bellied Fantail	11	12

Table II compares the accuracies for the bird call identification task obtained using the GMM-based classifiers and SVM-based classifiers using the state-of-the-art dynamic kernels mentioned in Section II. In this study, the GMMs whose parameters are estimated using the maximum likelihood (ML) method (MLGMM) or by adapting the parameters of the UBM to the data of a class (adapted GMM) [9] are considered for the GMM-based classifiers. The GMMs are built using diagonal covariance matrices. The accuracies presented in Table II are the best accuracies observed among the GMM-based classifiers and the SVM-based classifiers with dynamic kernels by varying the following parameters: Q, the number of components in the GMM or the UBM; δ, the width parameter of the Gaussian kernel used in GMM-based IMK; L, the number levels in the pyramid and b, the branch factor used in GMM-based PMK.

It is seen that the adapted GMM-based classifier gives better performance than the ML GMM-based classifier. The better performance of the adapted GMM-based system is mainly due to robust estimation of parameters using the limited amount of training data available for each class, as explained in [9]. It is also seen that performance of the SVM-based classifiers using the state-of-the-art dynamic kernels is comparable to that of the GMM-based classifiers. This is mainly because a GMM-based classifier is trained using the non-discriminative learning based technique, where as an SVM-based classifier using the dynamic kernels is built using a discriminative learning based technique. It is also seen that the GUMIK-based SVM performed better than other dynamic kernel based SVMs and

TABLE II. COMPARISON OF CLASSIFICATION ACCURACY (CA) (IN %) OF THE GMM-BASED CLASSIFIERS AND SVM-BASED CLASSIFIERS USING PSK, GMMSV KERNEL, GUMI KERNEL, GMM-BASED IMK AND GMM-BASED PMK FOR BIRD SPECIES RECOGNITION TASK. Q INDICATES THE NUMBER OF COMPONENTS CONSIDERED IN BUILDING GMM FOR EACH CLASS OR THE NUMBER OF COMPONENTS CONSIDERED IN BUILDING UBM OR THE NUMBER OF VIRTUAL FEATURE VECTORS CONSIDERED. THE PAIR (L, b) INDICATES VALUES OF L, NUMBER OF LEVELS AND b, BRANCHING FACTOR CONSIDERED IN CONSTRUCTING THE PYRAMID. C INDICATES THE TRADE-OFF PARAMETER IN SVM.

Classification Model		$Q/(L,b)$	C	CA
MLGMM		16	-	93.44
Adapted GMM		64	-	95.57
SVM using	PSK	1024	0.005	96.35
	GMMSV Kernel	512	1	96.05
	GUMI Kernel	128	0.1	**98.18**
	GMM-based PMK	(7,3)	1	97.57
	GMM-based IMK	128	0.1	97.87

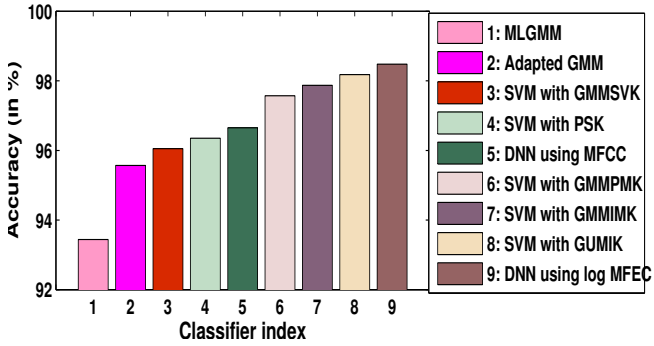

Fig. 2. Comparison of classification accuracy (in %) of the GMM-based classifiers, SVM-based classifiers using PSK, GMMSVK, GUMIK, GMMIMK & GMMPMK and fully connected DNN-based classifiers for bird call identification task.

the GMM-based classifiers.

Next, we study the effectiveness of DNNs for bird call identification. Experiments are carried out using different architectures i.e, different numbers of hidden layers (J) and different number of nodes (k) in each hidden layer. Table III presents the accuracies for the bird call identification task obtained using the different architectures while using MFCCs as well as log MFECs as features. It is seen that, in both cases a DNN with 3 hidden layers ($J = 3$) and 512 units ($k = 512$) in each hidden layer gives better accuracy (although not significantly so.) It is also observed that, in all cases, the DNN performed better while using log MFEC features than conventional MFCC features.

TABLE III. CLASSIFICATION ACCURACY OF DNN-BASED CLASSIFIERS FOR BIRD CALL IDENTIFICATION TASK WITH MFCC AND LOG MFEC FEATURES. HERE, J AND k INDICATES THE NUMBER OF HIDDEN LAYERS AND NUMBER OF UNITS IN EACH HIDDEN LAYER CONSIDERED RESPECTIVELY IN THE DNN.

J	k	MFCC	log MFEC
2	256	95.44	97.17
	512	96.05	98.18
	1024	95.74	98.18
3	256	96.35	97.42
	512	**96.65**	**98.48**
	1024	96.35	98.18

The observed best accuracies of SVM-based classifiers using dynamic kernels and DNN-based classifiers are compared in Figure 2 using bar charts. It is seen that the SVM classifiers with GUMIK perform better than the DNN using MFCC features. The classification accuracy of SVM with GUMIK is comparable with DNNs using log MFEC as features.

VI. CONCLUSIONS

In this paper, we explored speech and audio processing techniques for the identification of bird calls. We have explored dynamic kernel based support vector machine (SVM) and fully connected deep neural networks (DNNs) as classifiers for the identification of birds. This work focused on studying the calls of birds from the lower Himalayan regions. Twenty six bird species are considered for the study. MFCC features are extracted from each recording and is used to build dynamic kernel based SVM classifier and DNNs. The state-of-the-art dynamic kernels such as PSK, GMMSV kernel, GUMI kernel, GMM-based IMK and GMM-based PMK are considered to match the bird calls. Log Mel-filterbank energy coefficients are also considered as feature to build fully connected DNNs. The performance of dynamic kernel based SVMs in identifying the bird calls is compared with that of the DNN-based classifiers and results are found to be comparable.

The recordings of bird calls considered for this study are fairly clean. One of the future directions is to evaluate the above methods in noisy conditions. This study would give the robustness of the classifies considered in this work to noise. It is shown in the literature that the convolutional neural networks (CNNs) are more robust to the noise [20], [21]. Another direction to the future work is in exploring CNNs for the bird recognition task.

VII. ACKNOWLEDGEMENT

The authors wish to thank scientists from the National Center for Biological Sciences (NCBS), Bangalore, India for providing the recordings.

REFERENCES

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

[2] M. T. Lopes, C. N. Silla Junior, A. L. Koerich, and C. A. A. Kaestner, "Feature set comparison for automatic bird species identification," in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, October 2011, pp. 965–970.

[3] Dan Stowell and Mark D. Plumbley, "Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning," *PeerJ*, vol. abs/1405.6524, 2014.

[4] P. Somervuo, A. Harma, and S. Fagerlund, "Parametric representations of bird sounds for automatic species recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 2252–2263, November 2006.

[5] M. Graciarena, M. Delplanche, E. Shriberg, and A. Stolcke, "Bird species recognition combining acoustic and sequence modeling," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP 2011)*, Prague, Czech Republic, May 2011, pp. 341–344.

[6] Seppo Fagerlund, "Bird species recognition using support vector machines," *EURASIP ournal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 64–64, January 2007.

[7] Lee Ngee Tan, Kantapon Kaewtip, Martin L. Cody, Charles E. Taylor, and Abeer Alwan, "Evaluation of a sparse representation-based classifier for bird phrase classification under limited data conditions," in *Proceedings of INTERSPEECH*, Portland, Oregon, USA, September 2012, pp. 2522–2525.

[8] Douglas A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, pp. 91–108, August 1995.

[9] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, January 2000.

[10] A. D. Dileep and C. Chandra Sekhar, "GMM-based intermediate matching kernel for classification of varying length patterns of long duration speech using support vector machines," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 8, pp. 1421–1432, Aug 2014.

[11] K-A. Lee, C.H. You, H. Li, and T. Kinnunen, "A GMM-based probabilistic sequence kernel for speaker verification," in *Proceedings of INTERSPEECH*, Antwerp, Belgium, August 2007, pp. 294–297.

[12] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, April 2006.

[13] Chang Huai You, Kong Aik Lee, and Haizhou Li, "An SVM kernel with GMM-supervector based on the Bhattacharyya distance for speaker recognition," *IEEE Signal Processing Letters*, vol. 16, no. 1, pp. 49–52, January 2009.

[14] A. D. Dileep and C. Chandra Sekhar, "Speaker recognition using pyramid match kernel based support vector machines," *Internatiional Journal for Speech Technology.*, vol. 15, no. 3, pp. 365–379, September 2012.

[15] Yu-hsin Chen, Ignacio Lopez-Moreno, T Sainath, Mirkó Visontai, Raziel Alvarez, and Carolina Parada, "Locally connected and convolutional neural networks for small footprint speaker recognition," in *Proceedings of INTERSPEECH*, Dresden, Germany, September 2015, pp. 1136–1140.

[16] Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012.

[17] Sabri Boughorbel, Jean Philippe Tarel, and Nozha Boujemaa, "The intermediate matching kernel for image local features," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2005)*, Montreal, Canada, July 2005, pp. 889–894.

[18] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems (NIPS 2007)*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds., pp. 153–160. MIT Press, 2007.

[19] R. Collobert and S. Bengio, "SVMTorch: Support vector machines for large-scale regression problems," *Journal of Machine Learning Research*, pp. 143–160, 2001.

[20] Yun Lei, Luciana Ferrer, Aaron Lawson, Mitchell McLaren, and Nicolas Scheffer, "Application of convolutional neural networks to language identification in noisy conditions," in *Proceedings of Odyssey-2014*, Joensuu, Finland, June 2014, pp. 1–6.

[21] Mitchell McLaren, Yun Lei, Nicolas Scheffer, and Luciana Ferrer, "Application of convolutional neural networks to speaker recognition in noisy conditions," in *Proceedings of INTERSPEECH*, Singapore, September 2014, pp. 686–690.