INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Noise-robust Hidden Markov Models for limited training data for within-species bird phrase classification

Kantapon Kaewtip[1], *Charles Taylor*[2], *Abeer Alwan*[1]

[1]Department of Electrical Engineering, University of California, Los Angeles, USA
[2]Department of Ecology and Evolutionary Biology, University of California, Los Angeles, USA

kkaewtip@ucla.edu, alwan@ee.ucla.edu, taylor@biology.ucla.edu

Abstract

Hidden Markov Models (HMMs) have been studied and used extensively in speech and birdsong recognition, but they are not robust to limited training data and noise. This paper presents two novel approaches to training continuous and discrete HMMs with extremely limited data. First, the algorithm learns the global Gaussian Mixture Models (GMMs) for all training phrases available. GMM parameters are then used to initialize state parameters of each individual model. For the GMM-HMM framework, the number of states and the mixture components for each state are determined by the acoustic variation of each phrase type. The (high-energy) time-frequency prominent regions are used to compute the state emitting probability to increase noise-robustness. For the discrete HMM framework, the probability distribution of each state is initialized by the global GMMs in training. In testing, the probability of each codebook is estimated using the prominent regions of each state to increase noise-robustness. In Cassins Vireo phrase classification using 75 phrase types, the new GMM-HMM approach achieves 79.5% and 87% classification accuracy using 1 and 2 phrases, respectively, while HTK's GMM-HMM framework makes guess predictions resulting in 1.33% accuracy. The performance of the other algorithm is presented in the paper. [1]
Index Terms: Hidden Markov Models (HMMs), limited data, noise-robust, bird phrase classification

1. Introduction

Studies of birdsong syntax would benefit greatly from an ability to identify species and classify phrase types automatically [1, 2, 3]. Bird phrase classification is challenging due to within-class variability, limited training data, and noisy environments [4, 5]. This problem shares many features with speech processing, while presenting new challenges of its own [3].

Two spectrograms with same class labels may look different due to time misalignment and frequency variation [4]. Birdsongs become especially challenging when the song repertoire is diverse: some species have thousands of distinct phrases in their lexicons [6]. The frequency distribution of birdsong elements often resembles a Zipf distribution where some phrases appear many times, while others appear sparingly [7]. Thus, it is important to have an automatic classification system that can be trained with only a few samples per phrase. Furthermore, the amount of available training data may be limited by the logistics of the recording procedure. The lack of human annotation may also limit the amount of training labels even when more recordings are available. In real recording environments, the data can be corrupted by background interference such as rain, wind, other animals or even other birds vocalizing. Automatic birdsong systems may suffer from detecting non-target segments. Most systems are sensitive to noise and demand "a low-clutter, low noise environment" [8]. A noise-robust classifier needs to handle such adverse conditions that may be present in the actual deployment data.

Techniques such as support vector machines (SVMs), sparse representation, HMMs, and dynamic time-warping (DTW) have been used for birdsong classification [9, 10, 11, 12, 13, 14, 15]. A time alignment component (e.g. DTW and HMMs) is essential for birdsong classification. For example, SVMs and spare representation classifiers benefit greatly from integrating DTW into their frameworks [12]. For a recognition task where the signal is a continuous recording, DTW and HMMs can detect or recognize patterns without requiring a segmentation algorithm, which is prone to errors especially in a noisy environment [5]. For these reasons, DTW and HMMs are appealing frameworks. HMMs work well when there is sufficient training data but their performance suffer greatly when training data is limited due to the statistical nature of the algorithm. DTW, on the other hand, is robust to limited training data but its performance does not improve to the level of HMMs when more data become available. Both DTW and HMMs are susceptible to noise [14]. Some algorithms have been designed to reduce noise in bird songs based on signal enhancement techniques, such as spectral subtraction but the improvement is not dramatic [16, 17, 18, 4]. Prominent regions — time-frequency ranges expected to contain high energy for a particular phrase — have been successfully integrated with DTW and shown to be a noise robust component specially when non-target birds are singing in the background [5, 4].

The HMM framework is appealing since it has been studied extensively in speech recognition and other applications [19]. It can be readily used for a transcription task where the data is not pre-segmented and serveral well-developed HMM systems are available such as HTK and Kaldi [20, 21]. However, the HMM system degrades when training data is limited and the recording is corrupted by background noise. In this paper, we propose a methodology that compliments the existing HMM framework. For each phrase class, the proposed algorithm uses a phrase sample to obtain an initial model. This procedure is similar to the segment-based speech recognition framework where similar adjacent frames are grouped into a segment [22]. After that, the proposed algorithm uses information learned from other phrase types to derive model parameters. This procedure is inspired by shared-distribution HMMs but implemented in a different way [23, 24, 25]. The prominent region component is also integrated

[1]This work was supported in part by National Science Foundation Award Number 1125423.

http://dx.doi.org/10.21437/Interspeech.2016-1360

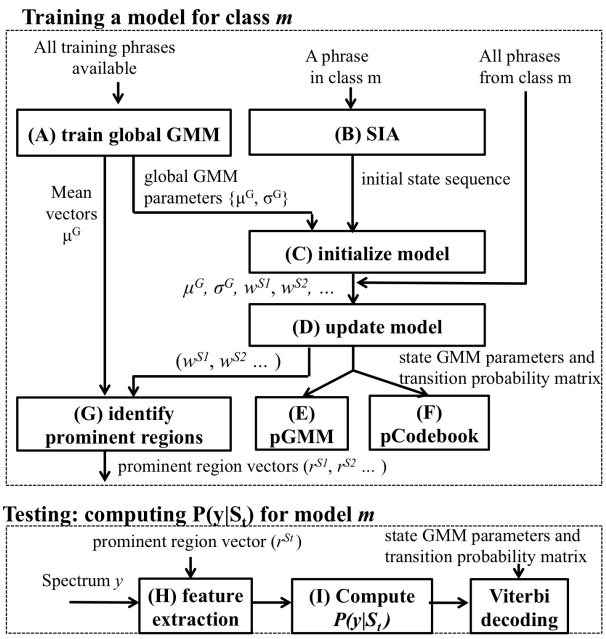

Figure 1: *System overview: training and testing*

into the system for noise robustness but in a statistical fashion.

2. Proposed Algorithm

Fig. 1 illustrates the proposed system. The training procedure derives GMM parameters to estimate the state emitting probabilities. Emitting probabilities can be determined by several methods such as using a GMM or a codebook.

2.1. Feature Extraction

For each sound file or phrase segment, a spectrogram is obtained in the same way as described in [4]. The DCT is then applied and the first 13 dimensions are retained. The first and second derivatives are also appended to the cepstral coefficients resulting in 39-dimensional feature vectors.

2.2. Global GMM

A global GMM is used to learn the distribution of the feature vectors of all phrases (Block A). First, a feature matrix is extracted from each file. Then each feature vector is treated as a sample point for the GMM. The number of mixture components is set to be 128, but it can be set to a higher number as well. For mixture component k, let μ_k^G, σ_k^G, w_k^G be its mean, covariance, mixture weight of the global GMM, respectively.

2.3. State Sequence Initialization Algorithm (SIA)

To determine an initial state sequence from single training file when there is no model available, we propose a State Sequence Initialization Algorithm (SIA) as shown in Block B. SIA groups similar adjacent frames into one state. As an initial step, SIA assigns State 1 to Frame 1 (now the current state is State 1). For a latter frame i, the algorithm considers the cosine similarity between Frame i and the first frame of the current state. Suppose State S_t is the current state and the first frame of State S_t is Frame q. If the cosine similarity between spectra y_i and y_q is greater than a threshold (here, 0.7), Frame i stays in the current state S_t. Otherwise, it starts a new state State S_{t+1} (in this case, Frame i is now the first frame of State S_{t+1}). This procedure carries on until the last frame is processed. Finally, the algorithm gives the initial state sequence and the total number of states. The number of states depends on the variation of the acoustic profile of the particular phrase type and the pre-defined similarity threshold.

2.4. Initial model parameters

The initial state sequence obtained from the SIA is used to derive initial model parameters (Block C). The transition probability matrix can be computed readily from the initial state sequence. The emission probability density function (PDF), however, can be difficult for HMM training to estimate especially if there is only 1 sample; there is no variance to fit a single Gaussian. Our algorithm utilizes the feature variation learned from the global GMM. Suppose SIA indicates that frame 1 to N are in State S_t. The membership weight $p_k^{(n)}$ of a particular frame n —the probability that frame n is generated by mixture component k —can be obtained from the Global GMM. The mixture weight of State S_t (w^{S_t}) is estimated by averaging the membership weights across N vectors (i.e., $w_k^{S_t} = \frac{1}{N}\sum_{n=1}^{N} p_k^{(n)}$). The PDF of State S_t can be then obtained by constructing a GMM whose mixture means and covariances are identical to those of the global GMM but the mixture weight vector changed to reflect the new distribution. In other words, the GMM parameters for State S_t are $\{\mu_k^G, \sigma_k^G, w_k^{S_t}\}$.

2.5. Parameter updates

One training sample is used to initialize model parameters. If more samples are available, model parameter estimates can further improve. If the training data size is sufficiently large, the EM algorithm is commonly used to re-estimate the parameters. Here, the Viterbi algorithm is used for simplicity. All training files belonging to a particular phrase type is used in this procedure (Block D). For each file, a feature matrix is extracted and aligned with the current model resulting in another state sequence. The state sequences of all files combined are then used to update the model for each state in the same fashion as in Section 2.4. Transition probabilities can be computed readily by considering the state sequences. After the final iteration, the PDF of each state is determined. The mixture mean and covariance parameters are the same as the global GMMs; only the mixture weight vector differs for each state.

2.6. Prominent Region Identification

Prominent regions are used to modify a given test spectrogram to compute the emitting probability of a given state. Prominent regions are used in testing (Block H) but first derived in training (Block G) as follows. First, the prominent regions for each mixture component is determined. An inverse DCT is applied to the first 13 coefficients (excluding the derivatives) resulting is a vector in the spectral domain. The algorithm selects frequency bins whose spectral amplitudes are higher than 20% of the maximum amplitude and expands to both higher and lower frequencies by 1 kHz on each side; the values for these bins are set to 1. The prominent regions for each state S_t (r^{S_t}) are obtained by a weighted summation of the prominent regions for those mixture components. The weighting function is simply the state mixture weight vector (w^{S_t}). The frequency bins that have the combination values greater than 0.5 are declared as

prominent, resulting in a prominent region vector of State S_t (r^{S_t}).

In testing, to compute the probability that Spectrum y is generated by State S_t whose prominent region vector is r^{S_t}, $y(b)$ is set to be zeros if $r^{S_t}(b) = 0$. The resulting spectrum is then used for feature extraction in testing (Block H). The decoding procedure is similar to traditional GMM-HMMs where the emitting probability $P(y|S_t)$ for each state S_t is first computed and the Viterbi decoding algorithm is used for classification. The difference of our framework in testing is that the prominent region is applied to feature vectors for computing $P(y|S_t)$ during Viterbi coding. Note that the prominent regions are an attribute of a state (just like its GMM parameters) that indicates which frequency bins are expected to contain relevant information. The process of deriving a prominent region vector is therefore performed only in training, and not during testing.

2.7. GMM-HMMs and Codebook-HMMs

Mixtures of Gaussians are used to estimate the PDF of each state in the GMM-HMM framework (Block E). After the parameter update, the GMM for each state $\{\mu_k^G, \sigma_k^G, w_k^{S_t}\}$ has 128 mixture components with a different mixture weight vector. The coefficients of these vectors are sparse so we simply drop any mixture component k whose weight is smaller than 0.0001 ($w_k^{S_t} < 0.0001$) for efficiency.

The codebook-HMM framework (Block F) differs from GMM-HMM in the sense that the emitting probability is estimated using a finite set of points (codes). If we treat each mixture mean vector of the global GMM (μ_k^G) as a codeword, we readily have $c_k = \mu_k^G$ where c_k is a code. The emission PDF for each code can be estimated using the mixture weight of State S_t (which has been derived from the parameter update procedure (Section 2.4)) or $P(c_k|S_t) = w_k^{S_t}$. We can assign a code to a feature vector y by selecting the component k with the highest membership weight $p_k^{(y)}$ or by leaving $p_k^{(y)}$ as the soft score as a code probability (the probability of being each code). For convenience, we refer to the first method as pCode1 and the probabilistic method as pCode2. In addition, we refer to pGMM as the proposed GMM-HMM framework.

3. Experimental setup

3.1. Database

The training set is obtained from Cassin's Verio songs recorded in 2013, while the test data is recorded in 2014. The most common 75 phrase types are selected. Sixteen samples are randomly selected for each phrase type from the training set (1200 samples in total) while the test data has 10 samples per phrase types (750 total samples). Each experiment was repeated three times and the results were averaged. The recordings and annotations for this study are available at http://taylor0.biology.ucla.edu/al/bioacoustics/.

3.2. Train and test conditions

This paper investigates two main factors that affect classification accuracies: the amount of training data and the level of background noise. Let N be the number of training samples. Under each training condition, a different number of samples was used to train each phrase class: N = 1, 2, 4, 8, and 16. Each experiment set was tested under 4 SNR conditions namely 10dB, 5dB, 0dB, and the clean condition. The background noise was recorded in the same environment when the target bird species was not singing [5, 4]. For a given test segment, each algorithm classifies which one of the 75 phrase types the segment belongs to. The average classification accuracy is observed from the 750 test samples for each train-test condition. Note that even though the proposed framework can also be used for phrase recognition, we chose to evaluate it on a classification task because the experiments can be better controlled (so that each phrase has the same N for training and each segment has the specified SNR as intended in testing, which can be difficult to achieve with continuous recordings.)

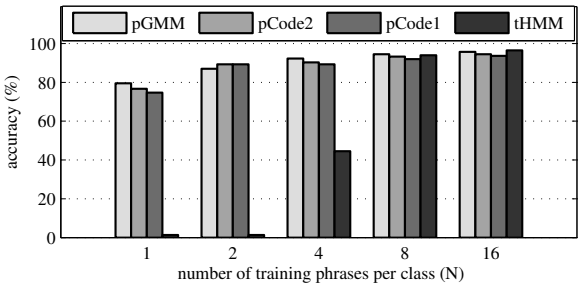

Figure 2: *Classification accuracies under different N (the number of phrase per class used for training)*

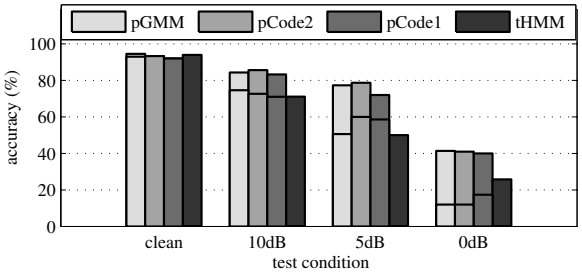

Figure 3: *Classification accuracies under different SNR test conditions when N = 8. The horizontal line in each bar represents the accuracy when the prominent regions are not used*

3.3. Baseline algorithm

The tHMM framework was executed using HTK-based backend [20, 26]. We modeled 75 phrase types with 17 states per left-to-right model, and each state is modeled using 1, 2 or 4 Gaussian mixtures (whichever gives the highest accuracy for each N) and diagonal covariance. MFCCs were used as front-end features for HTK with standard parameters. We refer to this traditional HMM framework as tHMM.

4. Results and discussion

4.1. Limited data

Fig. 2 shows the classification accuracies (Acc.) of each algorithm when tested in clean conditions but trained with different N. Across all experiments, the performance of each algorithm generally improves as more training data are available but with a different rate of improvement. In the clean train-test condition when N = 1, the proposed algorithms yield reasonable performance of 74.7% - 79.5% Acc. tHMM results in a pure guess (1/75 = 1.33 % Acc.) due to the limitation of the statistical na-

ture of the HMM algorithm. When N increases to 8 and above, the performance of tHMM increases significantly, while that of the proposed algorithms increases at a lower rate. Among all proposed algorithms, pGMM yields best performance in most cases. The performance of pCodebook2 (probabilistic codebook) is slightly better than that of PCodebook1 (deterministic codebook) for most cases.

4.2. Noise Robustness

Fig. 3 shows the accuracies of each algorithm when tested with different SNR conditions. The training condition is N = 8 where the performance of all algorithms are in the same range when tested in the clean condition (92.0 - 94.5 % Acc.). However, when there is noise present, tHMM underperforms the proposed algorithms by a large margin. In the 10dB-SNR condition, for example, the classification accuracy of all proposed algorithms is in the range of 83 - 86% Acc. while tHMM yields 71.07% Acc.

The horizontal line in each bar represents the accuracy when the prominent regions are not used in the system. For example, the first bar of the 10dB-SNR test condition has a horizontal line at 74.6% Acc. This means pGMM yields 74.6% Acc. when the system does not use the prominent regions to compute the emitting probabilities, indicating about 10% improvement when the prominent regions are utilized. The classification of all proposed algorithm increases significantly in the 0dB testing condition, a scenario where the the energy of background noise is equal to the target bird signal. This extreme scenario validates that the prominent regions are an essential component for a noise-robust system. In the clean-test condition, most algorithms do not benefit from prominent regions possibly because there is virtually no background noise under this condition.

4.3. Discussion

The proposed GMM-HMM and the traditional GMM-HMM work best for different Ns. One possible reason that tHMM outperforms the pGMM framework when there is sufficient training data (here, $N \geq 8$) is due to how Guassain mixtures fit training samples. For the proposed framework, the mixture parameters are also shared with other states to represent their sample points. This may compromise the mixture parameters, while tHMM can focus only on modeling a particular phrase type. However the trade-off is unavoidable because when the training data is not sufficient, we need to derive the model parameters by considering examples from similar phrase types.

The fact that pGMM performs well in limited training data condition is encouraging; we can use pGMM to compliment tHMM. The two training frameworks can be combined to create a better system. For example, a system may employ tHMM to train phrase classes that have at least N_0 samples and use pGMM to train the phrase classes with $N < N_o$. After training, all models will have the same format (i.e., each model has mixture means, mixture covariances, and mixture weights). If we do not employ pGMM, however, tHMM will simply provide guess prediction for the phrase classes whose model parameters can not be reliably estimated (e.g., 1.33% Acc. for N = 1 or 2 from our experiments). On the other hand, if pGMM is used to obtain a GMM-HMM when N = 1 or 2, the accuracy for this phrase type can go up to 80 - 90% Acc. Some novel components presented in this paper may be used to improve tHMM. For example, we can determine a suitable state number for each phrase type by using the SIA, and let the HMM system (e.g. HTK) work the rest as it normally does. The prominent regions will add noise robustness to the HMM framework.

The acoustic nature of birdsongs poses some challenges for bird phrase recognition, but also provides some advantages, allowing us to include the above components to birdsong recognition systems. For a certain species, phrase types can be numerous (e.g., 3000 phrase types) each of which may require 20 states, leaving us with 60,000 states to learn, some which can be quite similar. The advantage, however, is that most states share similar characteristics, enabling us to learn the distribution from phrase neighbors even though there is only a single example available for a given phrase type. In addition, birdsongs cover a wide range of frequencies, but a small region of frequency for a given phase and a given time instance, allowing us to extract prominent regions in order to exclude background noise. These characteristics may or may not directly apply to human speech. For example, the frequency coverage at a particular time of speech tends to be wide, and mostly in the low frequency range. The number of phonemes are relatively limited and most states do not share the same characteristics with others. However, parameter sharing is also used in estimate triphone models in speech but the conditions and implementations are different from our algorithm.

5. Summary and Conclusion

A noise-robust HMM framework for limited training data is proposed. This framework is subdivided into two algorithms depending upon how the emitting probability is estimated: GMMs and codebooks. Each phrase model generally has a different state number depending upon the dynamic acoustic variation of the phrase. These models are learned from not only their own phrase type (which can be limited) but also from the variation of other types. The prominent regions — an essential component for noise-robust classification of birdsongs — are used to modify the spectrum when extracting a feature vector, in order to compute the emitting probabilities. When all model parameters are obtained, the models can be used for both classification and recognition. However, we studied the relevant factors using a classification task. The proposed algorithm outperforms HMMs in most conditions. Out of 20 train-test conditions, the only scenario where tHMM outperforms the pGMM is when the system is trained with 16 samples and tested in the clean data set. When the number of training phrases is low, the proposed algorithms outperform HMMs by a large margin. When there are only 1 or 2 training phrases available per phrase type, the proposed GMM-HMM framework yields 79.5% and 88% classification accuracy, respectively, while HMMs make guess predictions (1.33% Acc). The experiments also validate that prominent regions are an essential component for noise-robustness, especially when the SNR is low.

We do not claim that the proposed system can replace the well-established HMM framework. Rather, we proposed a new algorithm, which can compliment the traditional HMMs. The integrated system can train a model effectively with limited data using the proposed GMM-HMMs. For phrase types that have sufficient training data, traditional HMMs can still be used to train their models. In future work, we plan to investigate if other components in the proposed system (such as the prominent regions or variable state numbers) benefits the traditional HMM framework and to integrate the two frameworks. Finally, we plan to apply the integrated system to fully automatic phrase recognition where pre-segmentation is not required.

6. References

[1] D. T. Blumstein, D. J. Mennill, P. Clemins, L. Girod, K. Yao, G. Patricelli, J. L. Deppe, A. H. Krakauer, C. Clark, K. A. Cortopassi *et al.*, "Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus," *Journal of Applied Ecology*, vol. 48, no. 3, pp. 758–767, 2011.

[2] A. Kershenbaum, D. T. Blumstein, M. A. Roch, Ç. Akçay, G. Backus, M. A. Bee, K. Bohn, Y. Cao, G. Carter, C. Cäsar *et al.*, "Acoustic sequences in non-human animals: a tutorial review and prospectus," *Biological Reviews*, 2014.

[3] 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.

[4] K. Kaewtip, L. N. Tan, A. Alwan, and C. E. Taylor, "A robust automatic bird phrase classifier using dynamic time-warping with prominent region identification," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 768–772.

[5] K. Kaewtip, L. N. Tan, C. E. Taylor, and A. Alwan, "Bird-phrase segmentation and verification: A noise-robust template-based approach," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 758–762.

[6] C. K. Catchpole and P. J. Slater, *Bird song: biological themes and variations*. Cambridge university press, 2003.

[7] L. N. Tan, K. Kaewtip, M. L. Cody, C. E. Taylor, and A. Alwan, "Evaluation of a sparse representation-based classifier for bird phrase classification under limited data conditions." in *Interspeech*, 2012, pp. 2522–2525.

[8] S. E. Anderson, A. S. Dave, and D. Margoliash, "Template-based automatic recognition of birdsong syllables from continuous recordings," *The Journal of the Acoustical Society of America*, vol. 100, no. 2, pp. 1209–1219, 1996.

[9] S. Fagerlund, "Bird species recognition using support vector machines," *EURASIP Journal on Applied Signal Processing*, vol. 2007, no. 1, pp. 64–64, 2007.

[10] M. A. Acevedo, C. J. Corrada-Bravo, H. Corrada-Bravo, L. J. Villanueva-Rivera, and T. M. Aide, "Automated classification of bird and amphibian calls using machine learning: A comparison of methods," *Ecological Informatics*, vol. 4, no. 4, pp. 206–214, 2009.

[11] L. N. Tan, K. Kaewtip, M. L. Cody, C. E. Taylor, and A. Alwan, "Evaluation of a sparse representation-based classifier for bird phrase classification under limited data conditions." in *INTERSPEECH*, 2012.

[12] L. N. Tan, A. Alwan, G. Kossan, M. L. Cody, and C. E. Taylor, "Dynamic time warping and sparse representation classification for birdsong phrase classification using limited training dataa)," *The Journal of the Acoustical Society of America*, vol. 137, no. 3, pp. 1069–1080, 2015.

[13] V. M. Trifa, A. N. Kirschel, C. E. Taylor, and E. E. Vallejo, "Automated species recognition of antbirds in a mexican rainforest using hidden markov models," *The Journal of the Acoustical Society of America*, vol. 123, no. 4, pp. 2424–2431, 2008.

[14] J. A. Kogan and D. Margoliash, "Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden markov models: A comparative study," *The Journal of the Acoustical Society of America*, vol. 103, no. 4, pp. 2185–2196, 1998.

[15] K. Ito, K. Mori, and S.-i. Iwasaki, "Application of dynamic programming matching to classification of budgerigar contact calls," *The Journal of the Acoustical Society of America*, vol. 100, no. 6, pp. 3947–3956, 1996.

[16] F. Briggs, X. Fern, and R. Raich, "Technical report (not peer reviewed): Acoustic classification of bird species from syllables: an empirical study."

[17] W. Chu and D. T. Blumstein, "Noise robust bird song detection using syllable pattern-based hidden markov models," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 345–348.

[18] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 27, no. 2, pp. 113–120, 1979.

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

[20] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, *The HTK book*. Entropic Cambridge Research Laboratory Cambridge, 1997, vol. 2.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[22] J. R. Glass, "A probabilistic framework for segment-based speech recognition," *Computer Speech & Language*, vol. 17, no. 2, pp. 137–152, 2003.

[23] M.-Y. Hwang and X. Huang, "Shared-distribution hidden markov models for speech recognition," *Speech and Audio Processing, IEEE Transactions on*, vol. 1, no. 4, pp. 414–420, 1993.

[24] S. J. Young and P. C. Woodland, "State clustering in hidden markov model-based continuous speech recognition," *Computer Speech & Language*, vol. 8, no. 4, pp. 369–383, 1994.

[25] D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. K. Goel, M. Karafiát, A. Rastrow *et al.*, "Subspace gaussian mixture models for speech recognition," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4330–4333.

[26] H.-G. Hirsch and D. Pearce, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.