BAYESIAN CLASSIFICATION OF FLIGHT CALLS WITH A NOVEL DYNAMIC TIME WARPING KERNEL

TECHNICAL REPORT

DEPARTMENT OF COMPUTER SCIENCE
INSTITUTE FOR COMPUTATIONAL SUSTAINABILITY
&
CORNELL LAB OF ORNITHOLOGY

Cornell University

Theodoros Damoulas, Samuel Henry,
Andrew Farnsworth, Michael Lanzone
and Carla P. Gomes.

November 9, 2010

Bayesian Classification of Flight Calls with a novel Dynamic Time Warping Kernel

Theodoros Damoulas
Department of Computer Science
Cornell University
Ithaca, 14853 NY, USA
damoulas@cs.cornell.edu

Samuel Henry
Department of Computer Science
Cornell University
Ithaca, 14853 NY, USA
sth56@cornell.edu

Andrew Farnsworth
Cornell Lab of Ornithology
Cornell University
Ithaca, 14853 NY, USA
af27@cornell.edu

Michael Lanzone
Biotechnology and Biomonitoring Lab
Carnegie Museum of Natural History
ARC, Rector, PA 15677, USA
mlanzone@gmail.com

Carla Gomes
Department of Computer Science
Cornell University
Ithaca, 14850 NY, USA
gomes@cs.cornell.edu

***Abstract*—In this paper we propose a probabilistic classification algorithm with a novel Dynamic Time Warping (DTW) kernel to automatically recognize flight calls of different species of birds. The performance of the method on a real world dataset of warbler (Parulidae) flight calls is competitive to human expert recognition levels and outperforms other classifiers trained on a variety of feature extraction approaches. In addition we offer a novel and intuitive DTW kernel formulation which is positive semi-definite in contrast with previous work. Finally we obtain promising results with a larger dataset of multiple species that we can handle efficiently due to the explicit multiclass probit likelihood of the proposed approach[1].**

***Index Terms*—Acoustic Signal Processing, Probabilistic Supervised Learning, Dynamic Time Warping, Kernel Machines**

I. Introduction

Birds are sensitive environmental indicators and among the first animals to respond to changes in local ecosystems and the global climate. Tracking bird populations in space and time is a central challenge for conservation science, particularly in light of the potential changes to the present climate of the planet, and the new field of computational sustainability provides novel insight into pursuing these challenges [1], [2]. The migratory patterns of birds can produce data of interest beyond the ornithological community, and therefore has been the focus of much recent research [3]–[9].

Determining the migration paths of birds at the species level is difficult but one of the most promising methods of tracking avian migrations is by flight call recording. Many species produce flight calls: species-specific vocalizations that vary in duration, frequency, and contour, and are frequently given during nocturnal migration by several hundred species in North America. These signals are the only source of information, at present, for reliably identifying passing nocturnal migrants. Recording stations that can capture such signals are relatively inexpensive and can be deployed remotely [5]. Additionally, these stations can be programmed to record autonomously, facilitating data collection for extended, nocturnal periods.

One of the primary limiting factors to expanding networks of recording stations from single station networks to thousands of microphones is the classification process. Classification of flight calls has traditionally been a labor-intensive and expensive manual process consisting of inspecting spectrograms by trained professionals. In this paper we present a method of automatically classifying such avian flight calls that allows for the deployment of large scale systems of recording stations, see Fig. 1, that have the ability to accurately track the migrations of birds around the world.

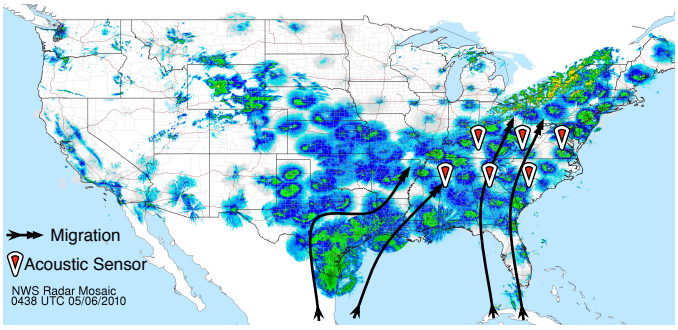

Fig. 1. Developing an efficient classification algorithm allows for large-scale networks of recording stations to be built. This image (United States Weather Surveillance Radar-88Doppler (WSR-88D) network) depicts a frontal boundary with scattered, intense precipitation over Lakes Erie and Ontario, spreading eastward into New York and New England. However, the image is dominated by biological targets, mostly birds, visible across the Great Plains south to Texas and east to the Southeastern Coastal Plain north along the Atlantic coast to Maine. The areas of intense, uniform green representing targets over the radars show bird densities close to 2.5×10^3 birds/km^3.

The contributions of this work are:

- Excellent recognition rates (97%) on automatic flight call detection of 5 within-family species (classes), and promising results on a larger dataset of 45 species within and across families. To put the results in perspective

[1]The data and codes are available at www.cis.cornell.edu/ics/projects.php and www.dcs.gla.ac.uk/inference/pMKL

experienced observers classify on average 69% - 91% correctly, while experts (of which there are very few) can classify between 88% - 100% correctly.
- A novel DTW positive semi-definite (p.s.d) kernel ($\text{DTW}_{\text{global}}$) that results in a very discriminative feature space for detection of acoustic signals.
- A successful application of the variational Bayes probabilistic classifier [10] to detect flight calls.
- Publicly available codes and real world datasets.

II. Previous Work

A. Flight Call Classification

There have been attempts to automate flight call classification in the past and the results were promising; but several factors, including robust call measurement and representation of intra- and inter-specific variation in calls, in a computational viable manner, were major unmet challenges. Successful methodologies included template matching schemes [11], [12] and statistical methods for classification [7], [8].

Flight calls must be extracted from the continuous audio streams produced by recording stations. This process is essential even for manual classification, so automated detection and extraction of flight calls from the raw data generated from individual recording stations has been the main focus of prior research [7], [13]. Automatic flight call detection has challenges, in particular the abundance of additional confounding signals in the frequency range of interest for flight calls, but works sufficiently well for the samples discussed in this paper. The detection algorithms are designed to identify high levels of energy with user-defined characteristics in specific frequency bands [5].

The microphones used to capture flight calls vary in type, but an inexpensive and effective design is Pressure Zone Microphones [8]. These microphones can be built with off-the-shelf parts, costing as little as $50 and can easily be deployed. For more information about the microphones used to capture the raw data, details can be found at [7] and http://www.oldbird.org.

B. Dynamic Time Warping Kernels

Dynamic Time Warping is a well-known dynamic programming method [14] that has been extensively applied to time-series and sequence-based problem domains. It operates by stretching a sequence[2] $\mathbf{x}_i \in \Re^{D_i}$ in order to match another sequence $\mathbf{x}_j \in \Re^{D_j}$ while calculating the cost of alignment $c_{\mathbf{x}_i,\mathbf{x}_j}$ based on an application-specific function or some standard distance measure derived from the warped path(s). A warping path π_{ij} of length $|\pi_{ij}| = p$ is a path in the $\mathbf{x}_i, \mathbf{x}_j$ graph, that can be seen [15] to define a pair of increasing p-tuples (ζ_i, ζ_j) as:

$$\begin{aligned} 1 = \zeta_i(1) \leq \cdots \leq \zeta_i(p) = D_i \\ 1 = \zeta_j(1) \leq \cdots \leq \zeta_j(p) = D_j \end{aligned} \quad (1)$$

[2]As usual x denotes scalar, $\mathbf{x}$ column vector and $\mathbf{X}$ matrix.

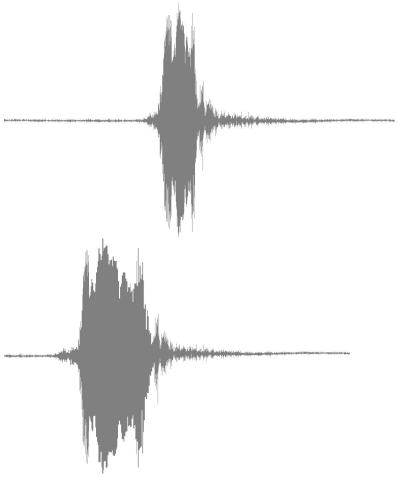

Fig. 2. An example of two flight call signals that are out of phase. The signals have similar structure but will appear very different if point to point comparisons are made.

where ζ_i is the warping transformation of the i^{th} sequence when mapped according to the path π_{ij} and intuitively describes a matching between elements of $\mathbf{x}_i$ and $\mathbf{x}_j$. D_i denotes the dimensionality of vector $\mathbf{x}_i$.

Recently there has been a great interest in constructing kernels based on DTW measures [15]–[17]. All the proposed approaches are developed within the Support Vector Machine (SVM) framework and hence cannot explicitly handle uncertainty due to their non-probabilistic nature. The previous kernels are summarized in Table I.

TABLE I
Previous DTW kernels

Work	Kernel function $k(\mathbf{x}_i, \mathbf{x}_j) =$
[16]	$\underset{\pi}{\text{argmax}} \frac{1}{\lvert\pi\rvert} \sum_{p=1}^{\lvert\pi\rvert} \exp(-\frac{1}{\sigma^2} \lvert\lvert \mathbf{x}_{i,\zeta_i(p)} - \mathbf{x}_{j,\zeta_j(p)} \rvert\rvert^2)$
[15]	$\sum_{\pi} \prod_{p=1}^{\lvert\pi\rvert} \exp(-\beta \lvert\lvert \mathbf{x}_{i,\zeta_i(p)} - \mathbf{x}_{j,\zeta_j(p)} \rvert\rvert^2)$

The kernel function proposed in [16] is not a p.s.d kernel in general [17], and the kernel in [15] is p.s.d "under favorable conditions" but can be diagonally dominant hence requiring additional smoothing. Related past work [18], [19] on constructing kernels for sequence-based problems has employed Hidden Markov Models (HMMs) as the *generative* (in contrast with this work that does not model parametrically the generating distribution) underlying model and have been applied to both supervised and unsupervised learning settings.

III. Method

In order to best present the approach, we first describe the proposed DTW kernel in comparison with previous formulations and then proceed to the adopted implementation for our specific flight call application.

A. A Dynamic Time Warping Kernel from Global Alignment

Both previous approaches construct a DTW kernel by effectively exploiting the path(s) constructed by the dynamic

programming steps within a single warping of sequences $\mathbf{x}_i, \mathbf{x}_j$. In [16] this results in a non-p.s.d kernel and in [15] it requires the consideration of all possible paths within the $D_i \times D_j$ matrix.

We instead construct a global alignment DTW kernel from the minimum-cost alignment scores $c^*_{\mathbf{x}_i\mathbf{x}_j}$ of the optimal paths $\{\pi^*_{in}\}_{n=1}^N$ directly (and not the paths themselves), taking into account the optimal N alignments of a signal $\mathbf{x}_i$ *with all* the available sequences in the training set $\{\mathbf{x}_n, t_n\}_{n=1}^N$. Hence we can employ kernel functions that result in a p.s.d kernel, since there is a common metric of alignment to N sequences, such as the Gaussian:

$$\text{DTW}_{\text{global}} \text{ Kernel} \quad k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{||\mathbf{c}^*_{\mathbf{x}_i} - \mathbf{c}^*_{\mathbf{x}_j}||^2}{\theta}\right) \quad (2)$$

where $\mathbf{c}^*_{\mathbf{x}_i} \in \Re^N$ is the vector of minimum-cost alignments through the optimal N warping paths $\{\pi^*_{in}\}_{n=1}^N$ and $\theta \in \Re$ the bandwidth (isotropic case). In the next section we describe explicitly how this cost is calculated in our implementation for flight call detection. It is worth noting that the proposed approach can exploit existing DTW implementations [20] with different cost functions in analogy to the *String* kernels proposed in bioinformatics and the availability of scoring matrices [21].

B. Minimum-cost alignment for Flight Calls

Given a training set $\{\mathbf{x}_i, t_i\}_{i=1}^N$, with sequence $\mathbf{x}_i \in \Re^{D_i}$ belonging to class t_i we first construct the spectrogram matrices $\mathbf{S}_{\mathbf{x}_i} \in \Re^{F\times W}$ of the signals where W is the number of windows and F the number of frequency bands from the short time Fourier transformation (STFT).

Having obtained the N spectrograms, we can construct a dissimilarity matrix $\mathcal{D}^{ij} \in \Re^{W\times W}$ for sequences $\mathbf{x}_i$ and $\mathbf{x}_j$ by one minus their (normalized) inner product:

$$\mathcal{D}^{ij}(w,v) = 1 - \frac{\mathbf{S}_{\mathbf{x}_i}(:,w)^\top \mathbf{S}_{\mathbf{x}_j}(:,v)}{\sqrt{\mathbf{S}_{\mathbf{x}_i}(:,w)^\top \mathbf{S}_{\mathbf{x}_i}(:,w)\mathbf{S}_{\mathbf{x}_j}(:,v)^\top \mathbf{S}_{\mathbf{x}_j}(:,v)}}, \quad (3)$$

where $\mathbf{S}_{\mathbf{x}_i}(:,w)$ denotes the w^{th} column of the $\mathcal{D}^{ij}$ matrix and $w, v \in \{1,\ldots,W\}$. The dynamic programming operations of standard DTW procedures can now operate on the dissimilarity matrix $\mathcal{D}^{ij}$ in order to obtain the optimal warping path and the minimum-cost alignment between sequences $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$\pi^*_{ij} = \underset{\pi}{\text{argmax}} \frac{1}{|\pi|} c_{\mathbf{x}_i,\mathbf{x}_j}(\pi)$$
$$c_{\mathbf{x}_i,\mathbf{x}_j}(\pi) = \sum_{p=1}^{|\pi|} \mathcal{D}^{ij}_{\pi(p)}. \quad (4)$$

These are collected to complete the description:

$$\mathbf{c}^*_{\mathbf{x}_i} = \left[c_{\mathbf{x}_i,\mathbf{x}_1}(\pi^*_{i1}), \cdots, c_{\mathbf{x}_i,\mathbf{x}_N}(\pi^*_{iN})\right]^\top. \quad (5)$$

Eq. 5 can be directly used as an "unkernelized" feature construction ($\text{DTW}_{\text{global}}$) and it is worth noting that the cost is symmetric, i.e. given optimal alignment paths π^*_{ij} and π^*_{ji}, the cost is the same: $c_{\mathbf{x}_i,\mathbf{x}_j}(\pi^*_{ij}) = c_{\mathbf{x}_j,\mathbf{x}_i}(\pi^*_{ji}) \; \forall \, i,j \in \{1,\ldots,N\}$.

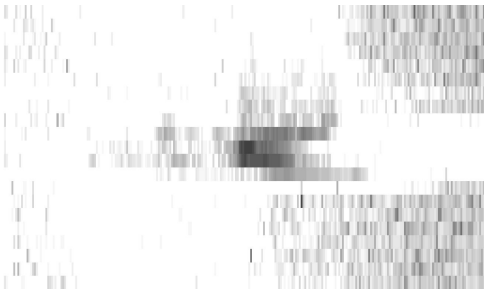

Fig. 3. Example of a spectrogram of the top signal shown in Fig. 2. The x-axis shows the normalized frequency. The y-axis shows time. Each interval on the y-axis represents a window. Shades of gray show the presence of a frequency. Dark gray and black indicate strong presence of the frequency, while light gray to white represents little to no presence of the frequency.

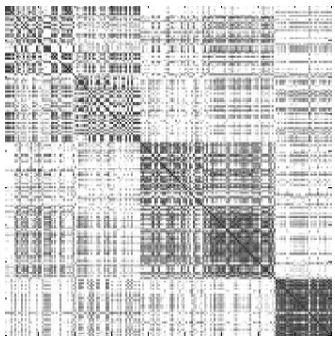

Fig. 4. The $\text{DTW}_{\text{global}}$ kernel from the primary flight call data (N=200, class sorted). The block structure captures the similarity of flight calls within and across species.

C. Probabilistic Multiple Kernel Learning

Having described the underlying DTW kernel representation we now turn to the probabilistic classifier that is adopted for the flight call detection. We employ the variational Bayes (VBpMKL) classifier recently proposed in [10] in order to take advantage of the uncertainty quantification measures (posterior variance) that are explicitly provided and can assist in decision making (costs and risks of misclassification). Furthermore, the methodology allows for inclusion of prior knowledge, which is vital for the spatio-temporal migration patterns of species of birds and their prior probability of being present and hence detected by such flight call systems.

Finally, the multiple kernel learning capabilities of the method provide a promising tool for integrating other sources of information for flight call detection such as spatio-temporal (time stamp and station that flight call is detected), radar traces, and even weather based information.

The approach is a kernel formulation of a generalized linear model (GLM) with an explicit multiclass probit likelihood that can handle multiple classes with a single underlying model. The likelihood is given by:

$$P\Big(t_i = m|\mathbf{W}, \mathbf{k}_i^{\boldsymbol{\beta\theta}}\Big) = \mathcal{E}_{p(u)}\left\{\prod_{l\neq m}\Phi\Big(u + (\mathbf{w}_m - \mathbf{w}_l)\,\mathbf{k}_i^{\boldsymbol{\beta\theta}}\Big)\right\}, \tag{6}$$

where $\mathbf{w}$ are the regression coefficients, $\boldsymbol{\beta}$ specifies the kernel combination parameters for MKL that are not employed in this work, $\boldsymbol{\Phi}$ is the normal cdf function, m, l denote classes, and $u \sim \mathcal{N}(0,1)$. Approximate Bayesian inference is performed via a variational treatment for the posterior distribution which is described in detail in [10].

IV. Pseudo-Code

The pseudo-code implementation of the proposed classification method is listed in Algorithm 1. Standard cross-validation procedures need to be adopted as usual.

Algorithm 1 Probabilistic DTW kernel classifier

```
Initialization and pre-processing
# STFT Spectrograms
for all X ∈ ℜ^{N×D_{x_i}} (Flight call sequences) do
    S_{x_i} ← spectrogram(x_i)
end for
# Dissimilarity (Cost) Matrix and DP
for i = 1 to N do
    for j = i to N do
        D^{ij} ← Eq. (3)
        c*_{x_i} ← Eq. (4,5)
    end for
end for
# Create the DTW Kernel
for i = 1 to N do
    for j = i to N do
        k(x_i, x_j) ← Eq. (2)
    end for
end for
# Classifier (VBpMKL) [10]
P(t_i = m|W, k_i) ← Eq. (6)
```

V. Experimental Setup and Feature Construction

The performance of the proposed approach is compared to several other classification methods and feature constructions on real world flight call datasets. All the reported results are for 10 replications of 10-fold Cross-Validation (10×10CV) unless otherwise stated and *baseline* denotes performance by assignment to largest populated class.

A. Data

Two flight call datasets are presented in this work. All the samples were collected via extraction from recording stations, recording captive birds, or recording and labeling via direct observation. The primary dataset consists of 5 classes (species of bird) with 40 samples (flight calls) from each class for a total of 200 samples. The five bird species recorded for the primary dataset are listed in Table II.

TABLE II
Bird Species

Common Name	Scientific Name	No. Samples
Magnolia Warbler	Dendroica magnoliae	40
Nashville Warbler	Vermivora ruficapilla	40
Chestnut-sided Warbler	Dendroica pennsylvanica	40
American Redstart	Setophaga ruticlla	40
Yellow-rumped Warbler	Dendroica coronata	40

The data is digitized and stored as .wav files. The calls range from between 1916 and 6037 features (sampling every 4.5×10^{-5} seconds); each call was padded with zeros to create a uniform length dataset. As an extension to the primary data set, an auxiliary data set consisting of 42 classes (species information available from our data repository), 1180 samples, and 15075 features. Samples per class range from 10 to 40, and call lengths range from 1175 to 15075. *Both datasets are publicly available at www.cis.cornell.edu/ics/projects.php*. One of the challenges of the primary data set is that most of the samples are warblers, which tend to have structurally similar flight calls. The auxiliary data set contains a larger variety of species, and therefore greatly varying samples of flight calls.

B. Global Features

When classifying flight calls, humans tend to look for a few global features of the calls. These attributes were extracted as an alternative feature construction to the proposed $\text{DTW}_{\text{global}}$ features and kernel: average energy value of the call, length of the flight call, maximum amplitude of the call and number of peaks.

C. Down Sampling

Down sampling, by averaging the signal over fixed-length intervals, can capture the general structure of the signal while reducing the total number of features. Frequency based feature extractions are based on standard Fast Fourier Transformations (FFT) with best performing settings.

D. Competing Classifiers

In order to examine the performance of VBpMKL and the various feature extraction methods, classification was performed with several other Weka classifiers [22] and an SVM Multiclass implementation [23]. The classifiers employed are given below with their implementation details and cover a range of popular classification techniques.

VI. Results

A. Primary Data Set

The technique described in this paper produces excellent results with a 97.6% average percent correct classification for the primary dataset under consideration and promising results with the 45 class flight call data. Furthermore, the underlying $\text{DTW}_{\text{global}}$ feature extraction method (kernelized or not) proves to be very discriminative for most classifiers employed. For Tables IV to VI we report only the mean of the 10×10CV as

TABLE III
CLASSIFIERS USED FOR COMPARISON

Classifier Type	*Implementation*	*Reference*
C4.5 Decision Tree	J48	[24]
Nearest Neighbors	Kstar	[25]
Bayesian Network	BayesNet	[26]
Regression	Simple Logistic	[27], [28]
Decision Table	Decision Table	[29]
Ensemble Decision Tree	Random Forest	[30]
Boosting Decision Stumps	Logit Boost	[31]
Ensemble Decision Tree	Rotation Forest	[32]
Support Vector Machine	$\text{SVM}^{\text{multiclass}}$	[23]

TABLE IV
AMPLITUDES AND DOWN SAMPLING, CORRECT CLASSIFICATION

Classifier	10	25	50	100	250	Raw
J48	57.5	**66.0**	61.0	60.0	53.0	55.0
Kstar	60.0	**67.5**	66.5	65.0	66.5	20.0
BayesNet	46.5	52.5	55.0	49.0	50.5	**67.5**
Simple Logistic	54.5	59.5	63.0	58.5	**64.5**	46.0
Decision Table	46.0	54.5	**52.5**	52.0	47.0	**52.5**
Random Forest	60.0	64.5	67.0	**73.0**	66.5	58.0
Logit Boost	58.0	59.0	62.5	60.0	55.5	**66.5**
Rotation Forest	61.5	67.0	68.5	66.5	66	**69.5**

TABLE V
FREQUENCY (FFT) AND DOWN SAMPLING, CORRECT CLASSIFICATION

Classifier	10	25	50	100	All
J48	73.35	**74.65**	73.65	73.65	75.2
Kstar	78.85	81.9	**83.05**	84.2	19.25
BayesNet	67.15	70.6	70.6	**72.2**	74.1
Simple Logistic	70.3	74.35	77.45	75.0	**84.4**
Decision Table	62.5	66.1	64.2	64.5	**67.85**
Random Forest	82.05	83.05	83.2	**83.55**	80.6
Logit Boost	80.25	80.9	**82.1**	80.75	82.0
Rotation Forest	83.35	84.9	85.6	84.8	**86.7**

TABLE VI
GLOBAL FEATURES, CORRECT CLASSIFICATION

Classifier	10	25	50	100	250
J48	62.5	**71.0**	63.5	60	61
Kstar	62.5	66.5	**70.5**	69.5	69.0
BayesNet	53.5	**56.5**	55.0	51.5	51.0
Simple Logistic	58.0	61.5	**63.0**	43.0	**63.0**
Decision Table	49.5	**55**	53.5	51.0	45.5
Random Forest	69.0	**74.0**	69.5	68.5	66.5
Logit Boost	64.5	65.5	**69.5**	64.0	59.5
Rotation Forest	70.5	**74.5**	70.5	72.5	70.5

the feature extraction methods are underperforming compared to the proposed DTW based constructions.

In Table IV we report the recognition rates with varying levels of down sampling and conversion to amplitudes. Each column shows the results of different levels of down sampling (varying the number of bins within which averaging takes place). The column headers indicate the length of the feature vector after down sampling occurs (e.g. 25 indicates a feature vector of length 25) and the *Raw* column shows the results when applying the various classifiers on the original data. Bold fonts indicate best performance of a classifier within specific feature extraction.

Table V presents results after FFT has been performed on either the original signal ("All") or in segmented versions (bins) with varying levels of down sampling and conversion to frequency. Again, the column headers indicate the length of the feature vector after down sampling occurs. The final alternative feature extraction in Table VI considers the aforementioned "global features" of the signals. Combining these features with FFT based extractions or amplitude information does not result in statistically significant improvements.

Table VII shows the performance of the proposed method and the best results for each classifier across feature extraction approaches for the primary dataset. The best window size for the $\text{DTW}_{\text{global}}$ was found by grid-search and cross validation to be 450 with an associated bandwidth of 0.4. All methods perform best with the $\text{DTW}_{\text{global}}$ feature construction and the adopted VBpMKL method achieves above 97% average recognition rate on the 5 class problem with the additional probabilistic benefits for prior knowledge inclusion, uncertainty quantification and data fusion. Considering the aforementioned recognition levels of human experts this is a significant step towards the development of automated flight call detection systems.

The second dataset that we consider has flight calls from a greater variety of species and hence poses a larger multi-class problem. In fact the inter and intra family structure of the flight calls can be exploited in a hierarchical manner that is ongoing work. In Table VIII we report preliminary recognition rates of VBpMKL with the proposed $\text{DTW}_{\text{global}}$ kernel. We achieve an average of 74% accuracy which, considering the 42 different classes of flight calls in the data, is a very good classification level and a promising benchmark to improve upon.

VII. COMPUTATIONAL COMPLEXITY

The DTW routine has complexity $\mathcal{O}\left(|\mathbf{x}_i||\mathbf{x}_j|\right)$ and the classifier a dominant term of $\mathcal{O}\left(CN^3\right)$ where $C, N, \mathbf{x}_i$ are the number of classes, samples and the length of sequence i. Both complexities can be improved via sparsity (sparsity inducing priors or regularization) and faster implementations.

VIII. INDIVIDUALS

The experiments and results discussed thus far only consider a sample's species when splitting datasets for testing and training. The true effectiveness of classification is revealed when individual specimens are accounted for. Both the primary and auxiliary datasets contain multiple flight call recordings produced by a single bird (an individual). An individual's flight calls contain less variation than that of flight calls from multiple individuals of the same class. Fig. 6 and Fig. 7 illustrates this point. They show four samples from the

TABLE VII
BEST RESULTS FOR EACH CLASSIFIER (BASELINE = 20%)

Classifier	*Feature Extraction Method*	10×10CV %
J48	DTW_{global}	87.1 ± 1.14
Kstar	DTW_{global}	**96.6 ± 0.65**
BayesNet	DTW_{global}	93.2 ± 0.27
Simple Logistic	DTW_{global}	94.9 ± 0.55
Decision Table	DTW_{global}	72.8 ± 3.82
Random Forest	DTW_{global}	93.2 ± 0.84
Logit Boost	DTW_{global}	91.7 ± 1.64
Rotation Forest	DTW_{global}	94.5 ± 1.06
$SVM^{multiclass}$	DTW_{global} Kernel	95 ± 0.43
VBpMKL	DTW_{global} Kernel	**97.6 ± 0.68**

TABLE VIII
VBPMKL & DTW_{GLOBAL} ON AUXILIARY DATA SET (BASELINE = 3.4%)

Window Size	Bandwidth	5CV %
450	0.00004	72.45 ± 12.39
450	0.00005	74.07 ± 13.64
650	0.00001	74.49 ± 12.50

same individual and four samples from separate individuals of the same species. There is noticeably more variation among individuals as is shown in Fig. 7. Datasets containing the same individual in both testing and training do not accurately represent a "real world" scenario and can produce unrealistically high correct classification rates, therefore test and training sets must be designed to ensure that the same individual does not occur in both.

All results discussed below are generated with individuals split; all samples from a single individual form a test set and samples from all other individuals form a training set. The primary dataset is well balanced consisting of four individuals per class, and 10 samples per individual for a total of 20 individuals and therefore 20 test sets. Results shown in Table XI show the result of 10 repetitions of 20-fold cross validation (10x20CV). The auxiliary dataset does not contain an even number of individuals per class or samples per individual and in total consists of 795 individuals. Table XXIII shows the individuals for the auxiliary dataset. Due to the computation time required to perform classification, one fold per individual is not practical for the large dataset as it would require 795 test sets and therefore 795 folds. Instead the data is split into test sets containing one individual per class until all individuals of each class have been tested. The result is 40 test sets and therefore 40 fold cross-validation. The results shown in Table XII are therefore result of a single repetition of 40-fold cross validation (1x40CV). The introduction of splitting for individuals produces two noticeable results, a drop in correct classification and an increase in standard deviation. The decrease of correct classification is a result of the increased difficulty, and the increase in standard deviation is due to the way in which test sets are created. Individual classification rates tend to be at one of the two extremes, all correct or all incorrect, and since each test set contains a single individual (or a set of individuals) the standard deviation increases. This is illustrated in Fig. 5, where it shows the correct classification for each individual of the auxiliary dataset. The x-axis indicates individual number and the y-axis indicates correct classification rate. Notice how most individuals are either at the top or the bottom. Few individuals are in the middle indicating that few individuals have classification rates between the two extremes.

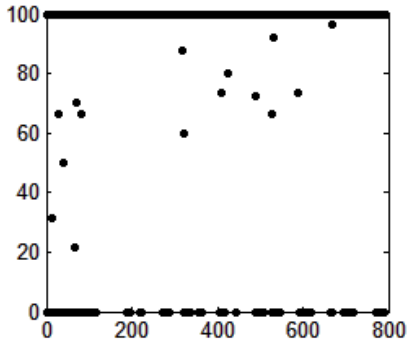

Fig. 5. The correct classification rates for each individual of the large dataset. The x-axis shows the individual number and the y-axis shows the correct classification rate.

Due to the introduction of individuals as well as new information on how to identify classes the statistics of the auxiliary dataset have changed slightly. Information about the auxiliary dataset is summarized in Table X and more detailed information can be seen in Table XXIII. The primary dataset remains unchanged, see Tables IX and XXIV for information about the primary dataset.

TABLE IX
PRIMARY DATASET INFORMATION

Number of Samples	200
Number of Classes	5
Number of Individuals	20
Signal Length Range	2241 - 6037
Average Signal Length	3265.52
Average Spectrogram Size (windowSize = 250)	129x25

TABLE X
AUXILIARY DATASET INFORMATION

Number of Samples	1178
Number of Classes	47
Number of Individuals	795
Signal Length Range	2559 - 15075
Average Signal Length	3496.6084
Average Spectrogram Size (windowSize = 250)	129x26

IX. CONSTRAINED DYNAMIC TIME WARPING

To improve correct classification rates the Dynamic Time Warping (DTW) process can be modified, in particular the warp paths can be restricted. Traditionally restricting warp paths (applying DTW constraints) has several advantages. First

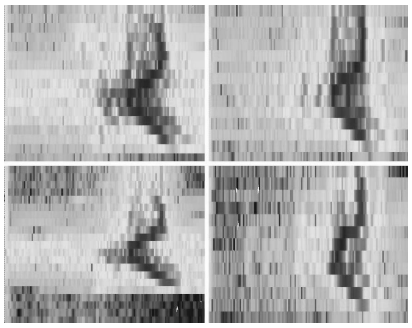

Fig. 6. An example of four flight call signals produced by the same individual. Each signal is structurally more similar than the signals shown in 7.

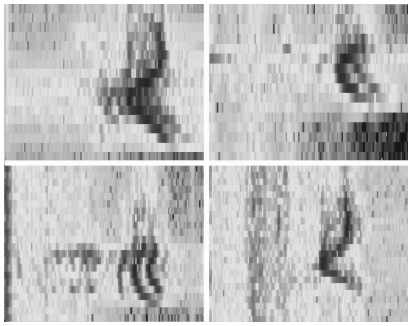

Fig. 7. Examples of a signals produced by different individuals of the same species. The variation among flight calls of different individuals is greater than the variation among the flight calls produced by the same individual as shown in 6.

TABLE XI
PRIMARY DATASET INDIVIDUALS SPLIT

Window Size	Bandwidth	10x20CV %
100	0.00005	77.00 ± 34.50
200	0.00005	77.50 ± 35.37
221	0.00005	**77.85 ± 34.49**
221	0.00004	76.80 ± 35.15
300	0.00005	76.50 ± 31.99
400	0.00005	75.50 ± 32.84
500	0.00005	71.00 ± 35.23

TABLE XII
AUXILIARY DATASET INDIVIDUALS SPLIT

Window Size	Bandwidth	1x40CV %
50	0.00005	54.82 ± 19.05
100	0.00005	53.17 ± 16.77
150	0.00005	56.41 ± 15.28
175	0.00005	55.84 ± 17.53
200	0.00005	**57.72 ± 19.04**
225	0.00005	53.34 ± 17.11
250	0.00005	56.31 ± 19.79
275	0.00005	54.47 ± 19.66
300	0.00005	53.19 ± 19.25
400	0.00005	49.9 ± 21.26
500	0.00005	45.31 ± 17.66
600	0.00005	44.94 ± 20.08

constraints can produce better results. "Pathological warping" (warping one signal to the point where comparisons are meaningless) can be prevented and more informative warp paths are explored [33]. Secondly DTW constraints can improve the run time of DTW; restricting warp paths to follow the diagonal decreases the computational complexity from $\mathcal{O}(|\mathbf{x}_i||\mathbf{x}_j|)$ to $\mathcal{O}(n)$, where *n* is the size of the diagonal of the dissimilarity matrix, $\mathcal{D}^{ij}$ [33]. Fig. 8 shows an example of the area where warp paths are allowed with constrained DTW. Warping is restricted to the light area, and as the area is restricted to the diagonal the allowable warp area shrinks. As the warp area shrinks so does the computational complexity since less paths need to be considered and less point-to-point comparisons need to be made.

Constraining DTW requires modifying two steps in the DTW routine. First the paths considered are restricted. This is accomplished by modifying the dynamic programming routine to only consider non-restricted paths. More precisely, let $\mathbf{A}$ define the area where warping is allowed. $\mathbf{A}$ is defined as a set containing pairs of values (ζ_i, ζ_j) that define points within $\mathcal{D}^{ij}(w, v)$. Eq. 4 should be modified as show in Eq. 7.

$$\pi^*_{ij} = \underset{\pi}{\operatorname{argmax}} \frac{1}{|\pi|} c_{\mathbf{x}_i, \mathbf{x}_j}(\pi) \; s.t. \; \forall (\zeta_i, \zeta_j)_k \in \pi, (\zeta_i, \zeta_j)_k \in \mathbf{A} \tag{7}$$

The implementation of Eq. 7 will vary depending on the constraint used, but the paths considered must be limited in order properly determine the minimum cost warp path, π^*.

Second, although not essential the construction of the dissimilarity Matrix $\mathcal{D}^{ij}(w, v)$ should be modified from Eq. 3 to the procedure shown in algorithm 2. Constructing $\mathcal{D}^{ij}(w, v)$ in this way reduces the computations required and produces a significant speed increase of the DTW routine.

Algorithm 2 Modified Dissimilarity Matrix Construction

1: **if** $(w, v) \in \mathbf{A}$ **then**
2: $\quad \mathcal{D}^{ij}(w, v) \leftarrow Eq.3$
3: **else**
4: $\quad \mathcal{D}^{ij}(w, v) \leftarrow \infty$
5: **end if**

A. Sakoe-Chiba Band

A Sakoe-Chiba (S-C) Band [34] is one of the simplest DTW constraints. S-C bands restrict the warp path to stay within a distance *k* of the diagonal. Fig. 9 shows all possible comparisons necessary to calculate all possible warp paths for a single point with varying S-C band sizes. As *k* decreases bands become tighter and less warping is allowed causing a decrease in the number of warp paths and necessary comparisons. The possible widths of an S-C band ranges *k=0* to *k=n/2* (where *n* is the length of the signal). When *k=0* the Euclidean distance between two signals is calculated and when *k=n/2* we have unconstrained warping with no restrictions. Fig. 8 shows S-C bands of varying sizes, warping is only allowed in light

areas. The results of S-C bands with individuals split is show in Table XIII.

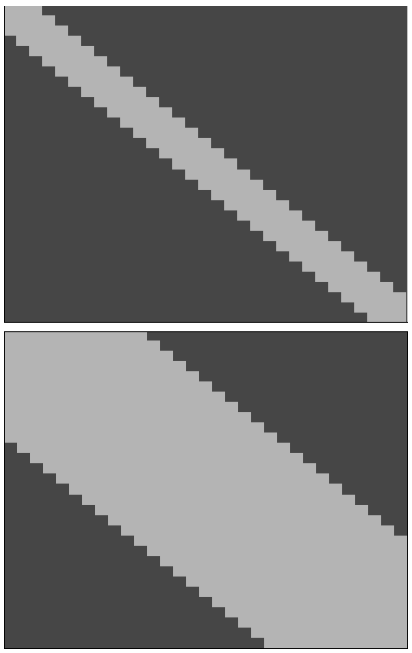

Fig. 8. An Example of two Sakoe-Chiba (S-C) bands of varying sizes. The top SK Band is size 2, the bottom is size 10. Warp paths are restricted to the light gray areas.

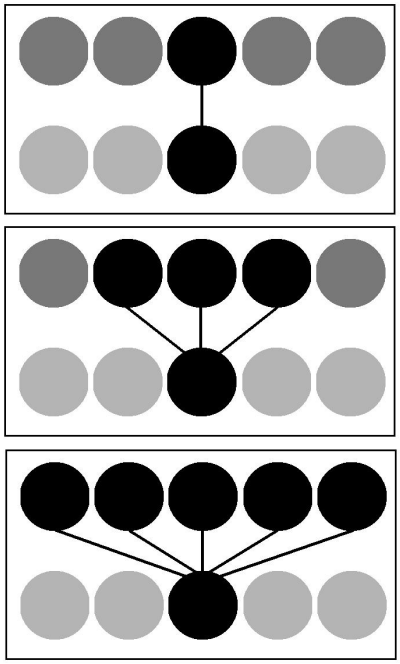

Fig. 9. As the size of an Sakoe-Chiba band increases more warping and therefore more comparisons can be made among elements of each signal. The effect of increased band size and number of necessary comparisons is shown above. As the band size increases so does the number of necessary comparisons.

TABLE XIII
SAKOE-CHIBA BAND RESULTS

Window Size	Bandwidth	Band Size	10x20CV %
200	0.00007	4	75.00 ± 36.49
200	0.00007	10	77.5 ± 34.01
200	0.00007	16	76.50 ± 34.07
100	0.00007	10	72.50 ± 38.09
300	0.00007	10	76.50 ± 33.29
200	0.000001	10	17.50 ± 33.70
200	0.001	10	73.00 ± 34.96

B. Ratanamahatana-Keogh Band

S-C bands are the simplest of DTW constraints and only allow for a small subset of available bands to be explored. Band shapes can vary much more than S-C bands, and the tractability of finding an optimal band shape becomes questionable for large datasets containing long signals (such as the auxiliary dataset). In such cases as with many large optimization problems a heuristic search can be used to find a solution. Ratanamahatana-Keogh (R-K) bands [34] allow arbitrarily shaped bands to be learned with a heuristic search. R-K Bands like S-C bands follow the diagonal, but rather than defining the band shape with a single value, k, it is defined by a vector of values, $\mathbf{b}$, where $\mathbf{b}$ is the same length as the diagonal and each element, $\mathbf{b}_i$ defines the width of the band at that point in the diagonal. Niennattrakul et al. describes the learning technique in detail, but essentially an optimal band shaped is learned via a binary search of the solution space (possible band shapes). The cost of each band shape is determined by the correct classification rate achieved with that band shape. The end result of the heuristic search is a DTW constraint that restricts some parts of the signal more than other parts allowing flexibility in parts of the signal where more variation occurs and forcing conformity where little variation occurs. An example of an R-K Band is show in Fig. 10, where warp paths are restricted to the light gray area.

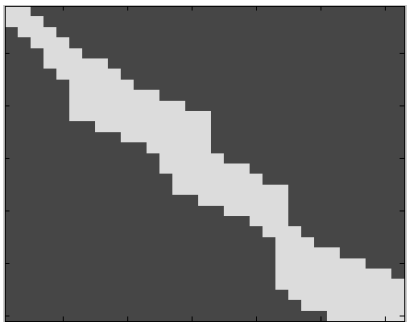

Fig. 10. An example Ratanamahatana-Keogh (R-K) Band, warping is restricted to the light gray area. An R-K band can take many shapes and is defined by the band size at each point in the diagonal.

Our implementation of R-K bands varies slightly from the technique described by Niennattrakul et al. A modified hillclimbing heuristic is used to overcome plateaus (areas where cost stays the same as band size increases) in the

solution space, and a single band shape is learned for the entire dataset as opposed to learning a different band shape for each class as Niennattrakul et al. and Ratanamahatana et al. propose. The implementation of the learning algorithm for R-K bands is outlined in algorithm 3. These modifications were necessary because the solution space contains many plateaus and the large number of classes and computation time limit the practicality of finding R-K Bands for each class. Plateaus often occur because the optimal warp path may not change until certain parts of the signal are made allowable for warping. This causes the optimal warp path and therefore the correct classification rate to jump from one value to another with plateaus in between jumps. Our search ends when costs begin to decrease as opposed to ending when costs stop increasing allowing the search to continue past plateaus, and as with Niennattrakul et al. and Ratanamahatana et al. we favor tight bands by selecting band shapes that occur at the start of plateaus. Using R-K Bands achieve a correct classification rate of 24.00 % on the primary dataset.

Algorithm 3 Ratanamahatana-Keogh Band Learning Technique

Initialization and pre-processing
# STFT Spectrograms
for all $\mathbf{X} \in \Re^{N \times D_{\mathbf{x}_i}}$ (Flight call sequences) **do**
  $\mathbf{S}_{\mathbf{x}_i} \leftarrow \text{spectrogram}(\mathbf{x}_i)$
end for
# Initialize R-K Bands
$\mathbf{b}^* \leftarrow 0$
$tree \leftarrow$ createBinarySearchTree
enqueue($tree.root$))
# Find Global Best R-K Band
while notEmpty($queue$) **do**
  $current \leftarrow$ pop($queue$)
  $[start, end] \leftarrow current.range$
  $\mathbf{b}_{temp} \leftarrow \mathbf{b}^*$
  # Find Best Band Within Range
  while $\mathbf{c} \geq \mathbf{c}^*$ **do**
    for $i = start$ to end **do**
      $\mathbf{b}_{temp}[i]$ ++
    end for
    c = evaluate($\mathbf{b}_{temp}$)
  end while
  #Update Global Best Bands
  if $\mathbf{c} > \mathbf{c}^*$ **then**
    $\mathbf{c}^* \leftarrow \mathbf{c}$
    $\mathbf{b}^* \leftarrow \mathbf{b}_{temp}$
  end if
  enqueue($current.children$)
end while
# Constrained DTW
$\mathbf{k} \leftarrow$ perform DTW with $\mathbf{b}^*$
Classifier (VBpMKL) [10]
$P(t_i = m|\mathbf{W}, \mathbf{k}_i) \leftarrow$ Eq. (6)

C. Box Constraint

Both S-C and R-K bands fail to significantly improve correct classification rates because both techniques fail to address the root of the problem: paths closely following the diagonal are often not the most informative paths. Both banding techniques assume that a diagonal path is favorable and restrain the warp path around the diagonal. Diagonal warp paths are often the lowest cost but not the most informative. This is caused by noise bands, manifestations of background noise that was not removed during feature extraction. An example of a noise band is shown in Fig. 11.

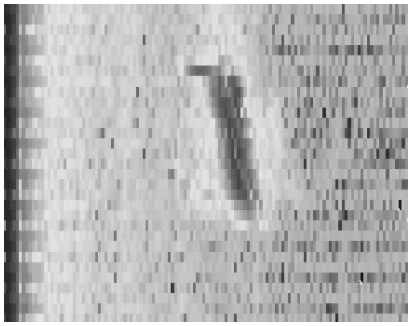

Fig. 11. An Example of a noise band. The noise band is shown as the dark band located at the far left of the spectrogram, and the frequency band is shown as the dark shape in the center of the spectrogram.

Due to the implementation (inner product) of DTW noise bands are indistinguishable from frequency bands (the most informative aspect of signals). A more informative warp path must be found that encourages comparisons between frequency bands and ignores noise bands. To accomplish this we propose a box constraint. A box constraint blocks some diagonal paths and therefore promotes non-diagonal warp paths. A sub-optimal warp path stresses comparisons between frequency bands because near-diagonal warp paths are often the result of noise bands corrupting the data. Fig. 12 shows the results of a box constraint. The sample is incorrectly classified using unconstrained DTW, but it is correctly classified with a box constraint. The lines show how the frequency band is warped and what comparisons are made using each DTW technique. Without a box constraint the frequency band of the first signal is warped and compared to a non-informative part of the signal, but with a box constraint the frequency bands are compared directly.

Fig. 13 shows an example of a box constraint; warp paths are allowed between all points on a signal except for pairs of points within the dark box. Box shapes are defined by the area outside of the box because this easily translates to new datasets with different signal lengths. Also optimal boxes for both the primary and auxiliary datasets have roughly the same number of data-points outside the box even though optimal box dimensions vary. Box constraints are defined by box top and box right where box top defines the area above a box and box right defines the area to the right of the box. All boxes are assumed to begin at the lower left of a dissimilarity matrix (Eq. 3).

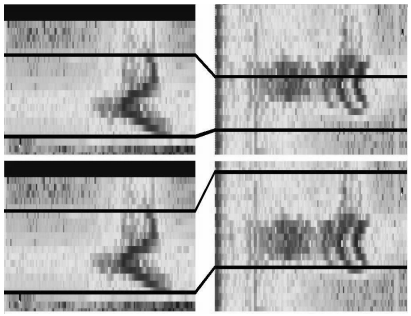

Fig. 12. An example showing the effect of a box constraint. The top shows the warping of the frequency bands when unconstrained DTW is performed, and the bottom shows warping with a box constraint. The frequency bands are correctly compared when a box constraint implemented.

Fig. 13. An example of a box Constraint. Warping is not allowed within the dark box, so comparisons between pairs of points falling within the box are not made forcing a non-diagonal warp path.

Determining the values of box top and box right is problem dependant but generally a box that covers a small portion of the diagonal for the majority of signals is favorable. This is accomplished by choosing a box right that is about the average length of a non-zero-padded spectrogram and a box right that allows some warping at the end of signals while still creating a box of sufficient size to cover the diagonal. We found that small values of box right and a box top of around 29 worked well.

Box Constraints produce excellent results. Results generated with a box constraint are shown in Tables XIV and XVI. The correct classification rate is high and certainty (probability it belongs to the classified class) of both correctly and incorrectly classified samples are generally favorable. Fig. 14 shows the discretized count of certainty for each sample of the primary dataset. Each bar indicates the number of samples classified with a certainty range of 10%. The indexes shown correspond to the ranges: [0,10), [10,20), [20,30), [30,40), [40,50) [50,60), [60,70),[70-80),[80,90), [90-100] respectively. The top graph shows certainty of samples that were classified correctly, and the bottom shows certainty for incorrectly classified samples. Certainty of misclassified samples centers around 50-60% indicating that there is little correlation between the sample and the classified class. This indicates that the classifier is unsure of its prediction and is essentially guessing. Low certainties for incorrectly classified samples is favorable, since with the addition of more data-sources the certainty for the true class can be increased allowing the sample to then be correctly classified. Certainty for correctly classified samples is generally high with the majority of samples falling in the 90-100% certainty range. This is also favorable since samples with high certainty are also less likely to be misclassified with the inclusion of more, possibly misleading data. The auxiliary dataset has similar certainties as is shown in Fig. 15.

TABLE XIV
PRIMARY DATASET WITH BOX CONSTRAINT

Box Top	Box Right	Window Size	10x20 %
24	5	250	85.50 ± 23.28
29	1	250	91.00 ± 17.75
29	5	250	**93.00 ± 12.61**
29	10	250	88.50 ± 18.430
39	5	250	77.00 ± 33.10
29	5	150	75.00 ± 35.31
29	5	350	80.50 ± 32.20

TABLE XV
AUXILIARY DATASET WITH BOX CONSTRAINT

Box Top	Box Right	Window Size	1x40 %
32	5	250	88.66 ± 10.45
30	5	250	89.51 ± 9.78
29	5	250	90.4 ± 9.47
28	5	250	89.63 ± 10.43
26	5	250	89.26 ± 11.64
24	5	250	88.83 ± 10.67
29	5	150	88.08 ± 11.67
29	5	350	84.33 ± 13.53
29	5	450	74.81 ± 17.48
29	5	550	68.43 ± 21.78
29	2	250	90.28 ± 8.81
29	4	250	89.96 ± 8.91
29	6	250	90.18 ± 11.85
29	8	250	90.43 ± 10.53
29	10	250	90.29 ± 9.72
29	15	250	**90.55 ± 9.43**
29	20	250	89.55 ± 10.21
29	25	250	88.71 ± 12.02
29	30	250	87.01 ± 14.18
29	35	250	85.69 ± 12.80

TABLE XVI
AUXILIARY DATASET WITH BOX CONSTRAINT

Box Top	Box Right	Window Size	1x795 %
31	5	250	84.27 ± 36.00
29	5	250	**86.93 ± 33.22**
27	5	250	84.60 ± 35.72
29	5	200	86.55 ± 33.38.72
29	5	300	84.60 ± 35.72

Tables XVII and XVIII display precision, recall, and F_1 scores. The scores for for table XVII were generated using the primary dataset with 10 repetitions of 20 fold cross-validation, and the scores for table XVIII were generated using the auxiliary dataset with 1 repetition of 795 fold cross-validation.

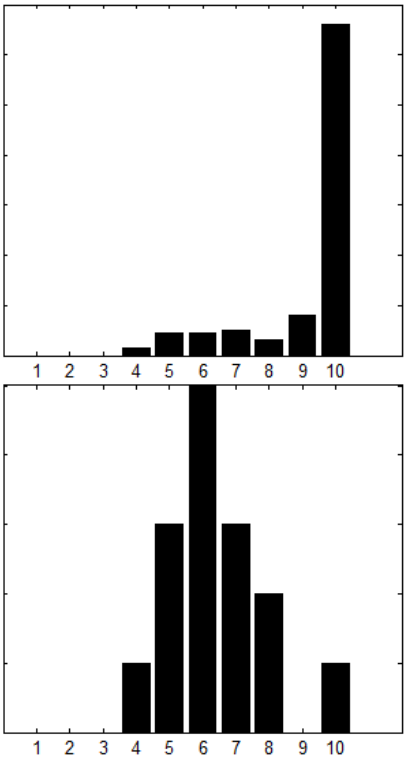

Fig. 14. Certainty of classification for the primary dataset. The top graph shows certainty for correctly classified samples and the bottom graph shows certainty for incorrectly classified samples.

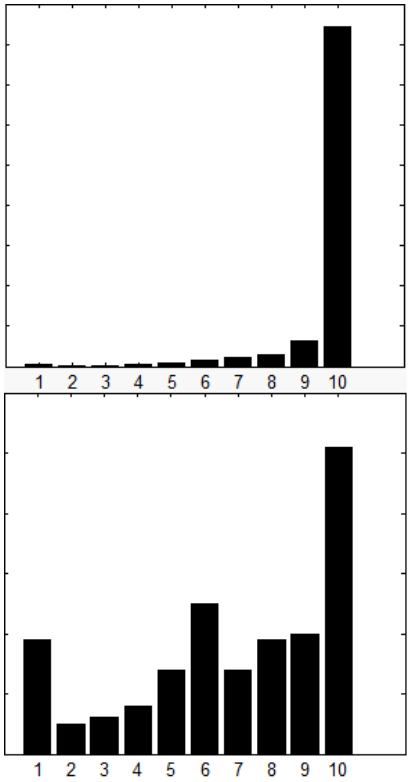

Fig. 15. Certainty of classification for the auxiliary dataset. The top graph shows certainty for correctly classified samples and the bottom graph shows certainty for incorrectly classified samples.

For both tables default parameters (shown in Table XIX) were used.

TABLE XVII
PRIMARY DATASET RESULT STATISTICS WITH DEFAULT PARAMETERS

Class Label	Recall	Precision	F_1 Score
AMRE	0.83	0.92	0.87
CSWA	0.93	0.95	0.94
MAWA	0.95	0.81	0.87
NAWA	0.93	1.00	0.96
YRWA	1.00	0.98	0.99

Fig. 16 shows the misclassifications for each class. Each row indicates the misclassifications for a single class with the column indicating the predicted class. Dark boxes indicate many misclassifications as that class and light boxes indicate few misclassifications as that class. For example in Fig. 16 the black box on the top row is at location (1,3), this indicates that class 1 (American Redstart) had many samples misclassified as class 3 (Magnolia Warbler). Table XXIV can be used to determine the species indicated by class labels of each row and column. Similarly Fig. 17 shows misclassification for the auxiliary dataset and Table XXIII can be used to determine species from class number and label. More detailed misclassification information for several classes of the auxiliary dataset with high rates of misclassification are shown in Fig. 18. The figures show the classes and exact count for which misclassified samples were classified.

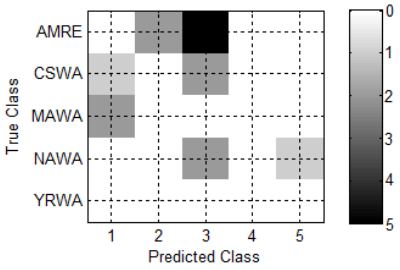

Fig. 16. Misclassified samples for each class and a count of the number of samples misclassified as each class for the primary dataset. Light boxes indicate few misclassifications, dark boxes indicate many misclassifications.

X. DEFAULT PARAMETERS

The results for both datasets converge to a maximum correct classification with similar parameters: box sizes, window sizes, and bandwidths. An optimal window size of around 250 creates spectrograms that can accurately represent the frequencies present without being redundant. A bandwidth of around 0.00007 separates the sample enough to distinguish between classes while still creating strong correlation between similar signals. The optimal boxes found both have around 29 data-points above them and 5 to the right (an optimal shape of 29x15 was found for the auxiliary dataset but the results are not significantly different than that of a 29x5 box). The default

TABLE XVIII
AUXILIARY DATASET RESULT STATISTICS WITH DEFAULT PARAMETERS

Class Label	Recall	Precision	F_1 Score
LOWA	0.72	0.42	0.53
GRWA	0.74	0.89	0.81
BTYW	0.00	NaN	NaN
CONW	0.90	0.84	0.87
MGWA	0.80	0.33	0.47
TEWA	0.26	0.75	0.39
OVEN	0.51	0.62	0.56
AMRE	0.83	0.63	0.71
AUWA	0.76	0.87	0.81
BAWW	0.73	0.80	0.76
BBWA	0.98	0.87	0.92
BGHE	1.00	0.97	0.99
BLBW	0.42	1.00	0.59
BLPW	0.90	0.84	0.87
BTBW	0.95	0.83	0.88
BTNW	0.84	0.94	0.89
BWWA	0.78	0.94	0.85
CAWA	0.93	0.76	0.83
CERW	1.00	1.00	1.00
CMWA	0.93	0.77	0.84
COLW	1.00	1.00	0.94
COYE	0.76	0.94	1.00
CSWA	0.95	0.93	0.77
GCWA	1.00	1.00	0.67
GWWA	0.89	1.00	0.94
HEWA	1.00	1.00	1.00
HOWA	0.73	0.83	0.77
KEWA	0.80	0.57	0.67
KIWA	0.00	NaN	NaN
LUWA	1.00	0.93	0.96
MAWA	0.89	0.82	0.85
MOWA	0.60	0.69	0.64
MYWA	1.00	0.87	0.93
NAWA	0.88	0.88	0.88
NOPA	0.46	0.75	0.57
NOWA	0.95	0.88	0.92
OCWA	1.00	0.86	0.92
PAWA	0.93	1.00	0.96
PAWH	1.00	1.00	1.00
PIWA	0.90	0.95	0.93
PRAW	0.00	0.00	NaN
PROW	0.57	0.44	0.50
SWWA	1.00	0.80	0.89
TOWA	1.00	1.00	1.00
VIWA	1.00	1.00	1.00
WIWA	0.75	0.60	0.67
YEWA	0.54	0.78	0.64
YRWA	0.94	0.89	0.92

parameters found to work well with both dataset are shown in table XIX and can serve as a starting point for additional datasets.

TABLE XIX
DEFAULT PARAMETERS FOR FEATURE EXTRACTION

*P*arameter	*V*alue
Box Top	29
Box Right	5
Window Size	250
Bandwidth	0.00007

XI. DIFFERENT SIGNAL LENGTHS

The feature extraction method proposed in this paper begins by padding signals with zeros to create a uniform length dataset. Alternative feature extraction methods are proposed by Ratanamahatana et al. A uniform length dataset is unnecessary as DTW can operate on different length signals, and a uniform length dataset can be created by"stretching" the signals via linear interpolation. To evaluate the effectiveness of zero-padding we consider both keeping the signals the same length and stretching the signals as alternate feature extraction procedures. The results of unconstrained DTW using these three techniques are shown in Table XX. These three techniques were also applied to DTW constrained by S-C bands and box constraints, results of which are shown in Tables XXI, XXII respectively. R-K Bands with different lengths was not performed since it requires uniform length datasets.

TABLE XX
DIFFERENT LENGTH TECHNIQUES WITH UNCONSTRAINED DTW

Length Technique	WindowSize	10x20CV %
different Lengths	300	75.5 $\pm$ 33.16
interpolation	100	68.5 $\pm$ 35.88
zero padding	221	**77.85 $\pm$ 34.49**

TABLE XXI
DIFFERENT LENGTH TECHNIQUES WITH SAKOE-CHIBA BANDS

Length Technique	Window Size	S-C Band Size	Best Result
different Lengths	300	4	26.50 $\pm$ 35.43
interpolation	100	12	37.00 $\pm$ 41.81
zero padding	221	16	**76.5 $\pm$ 33.92**

XII. DISCUSSION AND FUTURE DIRECTIONS

The immediate extensions of this work are to deal with the streaming nature of the flight call sensor data and to integrate spatio-temporal and ecological (GIS) information into the model. Scalability becomes an important aspect of an online system, and sparsity can reduce the computational complexity of kernel based methods via only retaining a few representative samples (e.g. relevance vectors) [35]. In the larger picture, interesting computational problems come into play as (near) optimal sensor placements for flight calls becomes an exciting direction for research that can couple predictive models of species distributions to flight call detection; and human observation locations (e.g. eBird: http://ebird.org) to acoustic sensor placements.

The development of a successful feature construction and classification methodology for flight call detection is an important step towards the overall goal of understanding bird migration. Increasing automation of the flight call analysis work flow for detection and classification will allow for the processing and reporting of increasing amounts of audio data in increasingly rapid and efficient analyses. This, in turn, will lead to more efficient use of trained experts' time in interpreting the acoustic record; and it follows that more timely and relevant analysis and interpretation of nocturnal migration is possible.

The proposed method, and specifically the $\text{DTW}_{\text{global}}$ kernel construction, appears to be generalizable to a larger class of sequence-based problems. The flexibility of being able to use different scoring functions within DTW routines is a further benefit for other applications where different scoring approaches can prove to be more beneficial.

Finally, recent developments in DTW banding and time-series research such as the LB-Keogh indexing [36] and the MCFF motif discovery approach [37] are very promising directions for improving computational complexity and recognition capabilities.

XIII. Acknowledgment

T.D. and C.G. acknowledge funding from NSF Expeditions in Computing grant on Computational Sustainability (Award Number 0832782). The authors would like to thank T.G. Dietterich and S. Kelling for feedback on drafts of the paper.

References

[1] C. P. Gomes, "Computational sustainability," *The Bridge, National Academy of Engineering*, vol. 39, no. 4, 2009.

[2] T. G. Dietterich, "Machine learning in ecosystem informatics and sustainability," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence. Pasadena, Calif.: IJCAI*, 2009, pp. 8–13.

[3] D. Sheldon, M. Elmohamed, and D. Kozen, "Collective Inference on Markov Models for Modeling Bird Migration," in *NIPS*, vol. 20. MIT Press, 2007, pp. 1321–1328.

[4] J. Yu, W.-K. Wong, and R. Hutchinson, "Modeling experts and novices in citizen science data for species distribution modeling," in *ICDM*. IEEE, 2010.

[5] A. Farnsworth and R. W. Russell, "Monitoring flight calls of migrating birds from an oil platform in northern gulf of Mexico," *Journal of Field Ornithology*, vol. 78, pp. 279–289, 2007.

[6] S. Kelling, W. Hochachka, D. Fink, M. Riedewald, R. Caruana, G. Ballard, and G. Hooker, "Data-intensive science: a new paradigm for biodiversity studies," *BioScience*, vol. 59, no. 7, pp. 613–620, 2009.

[7] W. R. Evans and D. K. Mellinger, "Monitoring grassland birds in nocturnal migration," *Studies in Avian Biology*, vol. 19, no. 219-229, 1999.

[8] A. Farnsworth and I. J. Lovette, "Phylogenetic and ecological effects on interspecific variation in structurally simple avian vocalizations," *Biological Journal of the Linnean Society*, vol. 94, 2008.

[9] ——, "Evolution of nocturnal flight calls in migrating wood-warblers: apparent lack of morphological constraints," *Journal of Avian Biology*, vol. 36, pp. 337–347, 2005.

[10] T. Damoulas and M. A. Girolami, "Pattern recognition with a Bayesian kernel combination machine," *Pattern Recognition*, vol. 30, no. 1, pp. 46–54, 2009.

[11] H. Figueroa and D. Michael, "Opening the BARN, a bioacoustic resource network," Presented at the XXII Meeting of the International Bioacoustics Council, 2009.

[12] D. Mellinger, D. Gillespie, H. Figueroa, K. Stafford, and T. Yack, "Software for bioacoustic analysis of passive acoustic data." *The Journal of the Acoustical Society of America*, vol. 125, p. 2547, 2009.

[13] R. R. Graber and W. W. Cochran, "Evaluation of an aural record of nocturnal migration," *Wilson Bull.*, vol. 72, pp. 252–273, 1960.

[14] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoustic Speech Signal Processing*, vol. 26, pp. 43–49, 1978.

[15] M. Cuturi, J.-P. Vert, óystein Birkenes, and T. Matsui, "A kernel for time series based on global alignments," in *ICASSP*. IEEE, 2007.

[16] H. Shimodaira, K. ichi Noma, M. Nakai, and S. Sagayama, "Dynamic time-alignment kernel in support vector machine," in *NIPS*. MIT Press, 2002.

[17] H. Lei and B. Sun, "A study on the dynamic time warping in kernel machines," in *IEEE Conference on Signal-Image Techonologies and Internet-Based Systems*, 2007.

[18] T. Jebara, Y. Song, and K. Thadani, "Spectral clustering and embedding with hidden Markov models," in *ECML*. Springer, 2007, pp. 164–175.

[19] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *NIPS*, 1999, pp. 487–493.

[20] D. Ellis, "Dynamic time warping (dtw) in matlab," 2003, available: http://www.ee.columbia.edu/ dpwe/resources/matlab/dtw/.

[21] L. Liao and W. S. Noble, "Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships," *Journal of Computational Biology*, vol. 10, no. 6, pp. 857–868, 2003.

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explorations*, vol. 11, no. 1, 2009.

[23] T. Joachims, "Multi-class support vector machine," Online, 2008, available: http://svmlight.joachims.org/svm_multiclass.html.

[24] R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA.: Morgan Kaufmann Publishers, 1993.

[25] J. G. Cleary and L. E. Trigg, "K*: An instance-based learner using an entropic distance measure," *ICML*, pp. 108–114, 1995.

[26] R. R. Bouckaert, "Bayesian network classifiers in Weka for version 3-5-7," http://www.cs.waikato.ac.nz/ remco/weka.bn.pdf, May 2008.

[27] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," 2005.

[28] M. Sumner, E. Frank, and M. Hall, "Speeding up logistic model tree induction," in *PKDD*, 2005, pp. 675–683.

[29] R. Kohavi, "Power of decision tables," in *8th ECML*, 1995, pp. 174–189.

[30] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[31] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," Master's thesis, Stanford University, 1998.

[32] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: A new classifier ensemble method," *IEEE Transactions PAMI*, vol. 28, no. 10, pp. 1619–1630, 2006.

[33] E. K. Chotirat Ann Ratanamahatana, "Three myths about dynamic time warping data mining."

[34] ——, "Making time-series more accurate using learnined constraints."

[35] I. Psorakis, T. Damoulas, and M. A. Girolami, "Multiclass relevance vector machines: Sparsity and accuracy," *IEEE Transactions on Neural Networks*, 2010, in Press.

[36] E. Keogh and C. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, 2005.

[37] A. Mueen and E. Keogh, "Online discovery and maintenance of time series motifs," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 1089–1098.

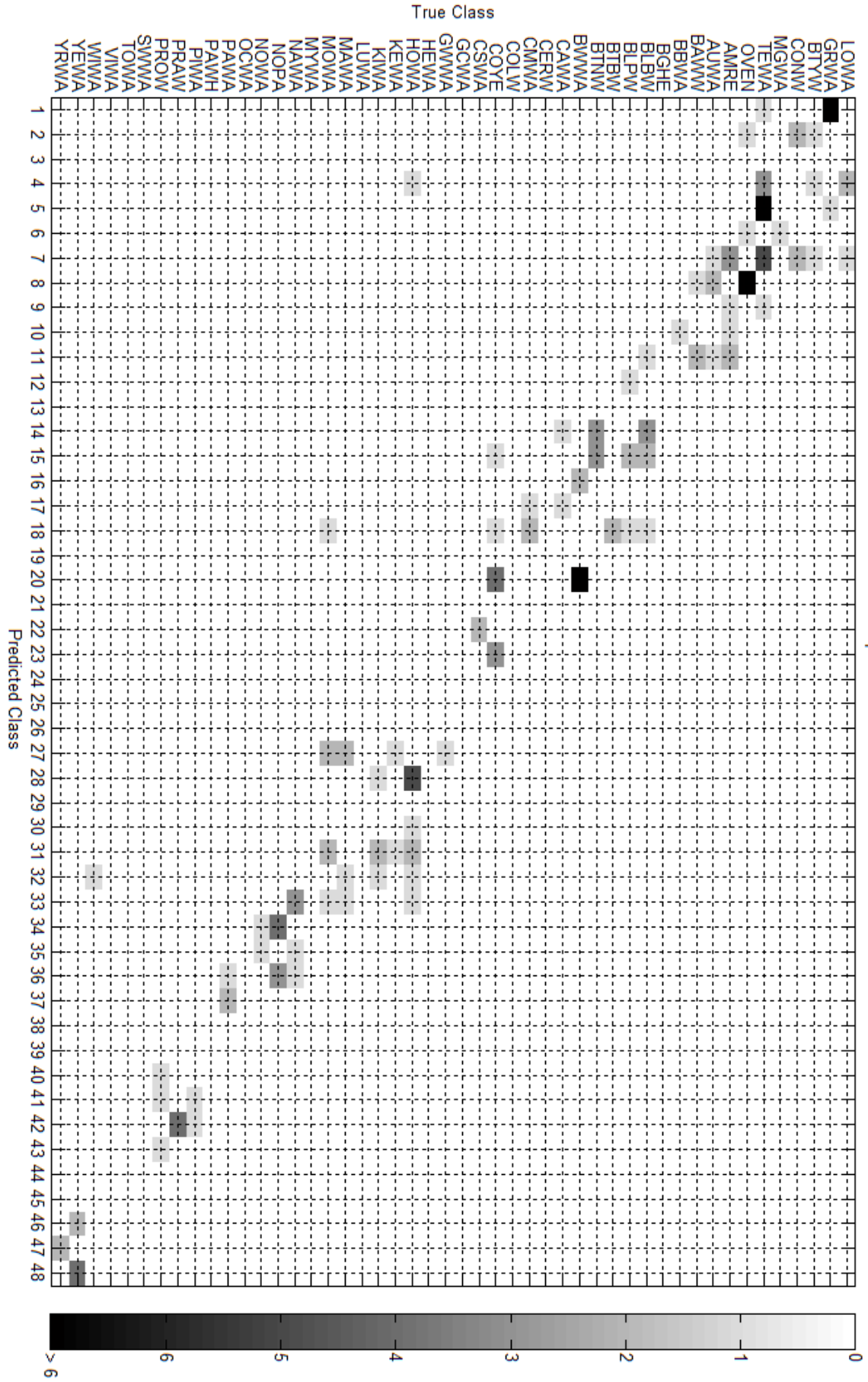

Fig. 17. Misclassified samples for each class and a count of the number of samples misclassified as each class for the auxiliary dataset. Light boxes indicate few misclassifications, Dark boxes indicate many misclassifications.

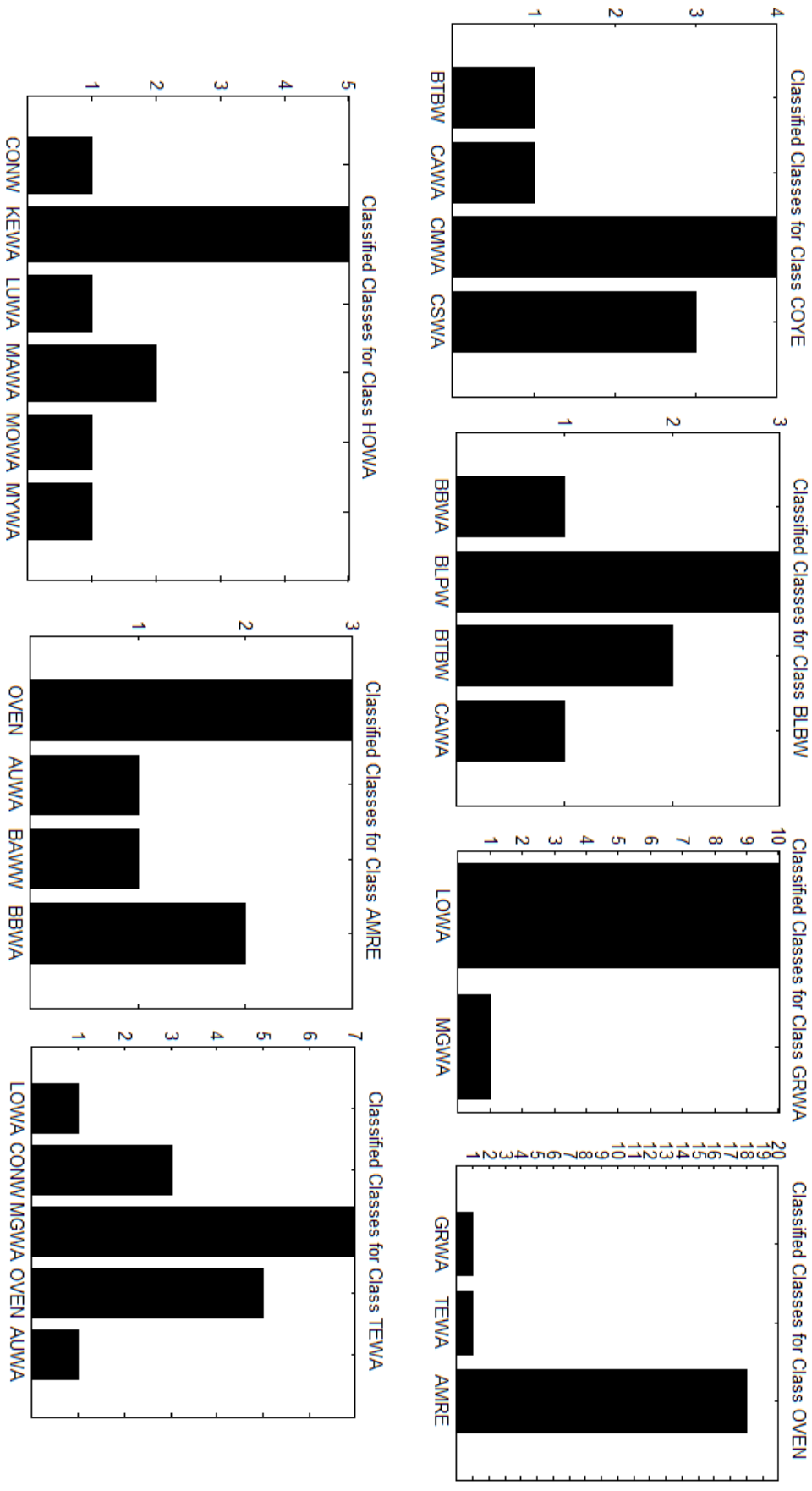

Fig. 18. Misclassified samples for a few highly misclassified classes and their predicted classes.

TABLE XXII
DIFFERENT LENGTH TECHNIQUES WITH BOX CONSTRAINTS

Length Technique	Window Size	Box Top	Box Right	Best Result
different Lengths	200	31	5	85.50 $\pm$ 27.43
interpolation	100	19	2	64.5 $\pm$ 36.02
zero padding	250	29	5	**93.00 $\pm$ 12.61**

TABLE XXIII
AUXILIARY DATASET

Class Label	*Species Name*	Class Number	*# Samples*	*# Individuals*
AMRE	American Redstart	8	42	15
AUWA	Yellow-rumped "Audubon's" Warbler	9	17	17
BAWW	Black-and-white Warbler	10	11	11
BBWA	Bay-breasted Warbler	11	40	40
BGHE	"Dendroica virens" group warbler	12	34	34
BLBW	Blackburnian Warbler	13	12	9
BLPW	Blackpoll Warbler	14	40	40
BTBW	Black-throated Blue Warbler	15	40	40
BTNW	Black-throated Green Warbler	16	38	37
BTYW	Black-throated Gray Warbler	3	3	3
BWWA	Blue-winged Warbler	17	40	13
CAWA	Canada Warbler	18	27	27
CERW	Cerulean Warbler	19	2	2
CMWA	Cape May Warbler	20	40	40
COLW	Colima Warbler	21	9	9
CONW	Connecticut Warbler	4	41	11
COYE	Common Yellowthroat	22	38	15
CSWA	Chestnut-sided Warbler	23	40	10
GCWA	Golden-cheeked Warble	24	10	10
GRWA	Grace's Warbler	2	42	6
GWWA	Golden-winged Warbler	25	9	9
HEWA	Hermit Warbler	26	70	36
HOWA	Hooded Warbler	27	40	12
KEWA	Kentucky Warbler	28	10	10
KIWA	Kirtland's Warbler	29	4	4
LOWA	Louisiana Waterthrush	1	11	11
LUWA	Lucy's Warbler	30	12	12
MAWA	Magnolia Warbler	31	36	9
MGWA	MacGillivray's Warbler	5	5	5
MOWA	Mourning Warbler	32	15	15
MYWA	Yellow-rumped "Myrtle" Warbler	33	12	40
NAWA	Nashville Warbler	34	40	11
NOPA	Northern Parula	35	13	11
NOWA	Northern Waterthrush	36	40	40
OCWA	Orange-crowned Warbler	37	12	12
OVEN	Ovenbird	7	41	19
PAWA	Palm Warbler	38	40	13
PAWH	Painted Redstart (Whitestart)	39	13	13
PIWA	Pine Warbler	40	21	21
PRAW	Prairie Warbler	41	4	4
PROW	Prothonotary Warbler	42	7	7
SWWA	Swainson's Warbler	43	4	4
TEWA	Tennessee Warbler	6	23	14
TOWA	Townsend's Warbler	44	13	13
VIWA	Virginia's Warbler	45	36	36
WIWA	Wilson's Warbler	46	4	4
YEWA	Yellow Warbler	47	13	13
YRWA	Yellow Rumped Warbler	48	36	8

TABLE XXIV
AUXILIARY DATASET

Class Label	*Species Name*	Class Number	*# Samples*	*# Individuals*
AMRE	American Redstart	1	40	10
CSWA	Chestnut-sided Warbler	2	40	10
MAWA	Magnolia Warbler	3	40	10
NAWA	Nashville Warbler	4	40	10
YRWA	Yellow-rumped Warbler	5	40	10