Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach

Forrest Briggs,[a] Balaji Lakshminarayanan, Lawrence Neal, Xiaoli Z. Fern, and Raviv Raich
Department of Electrical Engineering & Computer Science, Oregon State University, Corvallis, Oregon 97331

Sarah J. K. Hadley, Adam S. Hadley, and Matthew G. Betts
Department of Forest Ecosystems & Society, Oregon State University, Corvallis, Oregon 97331

(Received 5 June 2011; revised 2 February 2012; accepted 25 March 2012)

Although field-collected recordings typically contain multiple simultaneously vocalizing birds of different species, acoustic species classification in this setting has received little study so far. This work formulates the problem of classifying the set of species present in an audio recording using the multi-instance multi-label (MIML) framework for machine learning, and proposes a MIML bag generator for audio, i.e., an algorithm which transforms an input audio signal into a bag-of-instances representation suitable for use with MIML classifiers. The proposed representation uses a 2D time-frequency segmentation of the audio signal, which can separate bird sounds that overlap in time. Experiments using audio data containing 13 species collected with unattended omnidirectional microphones in the H. J. Andrews Experimental Forest demonstrate that the proposed methods achieve high accuracy (96.1% true positives/negatives). Automated detection of bird species occurrence using MIML has many potential applications, particularly in long-term monitoring of remote sites, species distribution modeling, and conservation planning.
© *2012 Acoustical Society of America*. [http://dx.doi.org/10.1121/1.4707424]

PACS number(s): 43.60.Bf, 43.80.Ev, 43.60.Np, 43.60.Vx [MAH] Pages: 4640–4650

I. INTRODUCTION

Current and projected declines in biodiversity as a function of habitat loss[1] and climate change[2] necessitate the development of efficient and accurate estimates of species' diversity, habitats, and phenology. Birds have been used widely as indicators of biodiversity because they provide critical ecosystem services, respond rapidly to change, are relatively easy to detect, and may reflect changes at lower trophic levels (e.g., insects, plants).[3] Birds have thus been proposed as "canaries in the coal mine" with respect to anthropogenic environmental changes at both local and global scales.

Unfortunately, collection of data on trends in birds populations has been plagued by problems of poor sample representation in remote regions, observer bias,[4] imperfect detectability,[5] and, particularly, the prohibitive costs of sampling over large spatial and temporal scales at sufficiently fine resolutions.[6] These problems could be ameliorated to some degree with the use of automated acoustic surveys. However, the complexity of bird song, the noise present in most habitats, and the simultaneous song that occurs in many bird communities[7,8] make automated species identification a challenging task.

Many authors have proposed methods for acoustic bird species classification, but more work is needed to address the problem of identifying all species present in noisy recordings containing multiple simultaneously vocalizing birds.[9] It is common to classify species under the assumption that there is a single bird species present in a recording.[10–12] This assumption is reasonable for audio collected with hand-held directional microphones aimed at a target individual,[13–15] or for audio collected from birds in captivity,[16] but not for audio collected by unattended omnidirectional microphones.[17] A related problem is detection of one or a few specific species[17,18] (possibly amidst other sources of noise, including other birds), or detection of birds that make a particular type of call (e.g., tonal sounds[19]).

Unlike prior work in automatic bird sound detection and classification, we consider the following problem: given an audio recording (e.g., 10 s), predict the *set* of all species present in that recording.

We formulate this problem in the multi-instance multi-label (MIML) framework for supervised classification.[20] The main idea of MIML is that the objects to be classified are represented as a collection of parts (referred to as a "bag-of-instances"), and associated with multiple class labels. In this application, the objects to be classified are recordings, the parts are segments of the spectrogram corresponding to syllables of bird sound described by a feature vector of acoustic properties, and the labels are the species present. All supervised classification algorithms require some labeled training data to build a predictive model. A major advantage of the MIML formulation is that the only training data required is a list of the species present in a recording, rather than a detailed annotation of each segment, or training recordings containing only a single species (which is required in most prior work).[21–24] For recordings containing multiple simultaneously vocalizing species of bird, it is less labor intensive to construct the former type of labels.

[a]Author to whom correspondence should be addressed. Electronic mail: briggsf@eecs.oregonstate.edu

 0001-4966/2012/131(6)/4640/11/$30.00 © 2012 Acoustical Society of America

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

In order to apply MIML classification algorithms, it is necessary to transform the data from its original representation into a suitable bag-of-instances representation. An algorithm to do this is called a "bag generator." In prior work, MIML bag generators for images and text have been proposed, but MIML has not previously been applied to audio. We propose a MIML bag generator for audio, which makes it possible to apply existing MIML classifiers to the species set prediction problem.

We experimentally evaluate MIML acoustic species classification on 548 10-s recordings containing 13 species.[25] These experiments demonstrate that our methods accurately predict the set of species present in noisy, multi-bird recordings collected in the field by unattended omnidirectional microphones.

II. BACKGROUND AND RELATED WORK

In this section, we discuss segmentation, features and classifiers used in prior work on acoustic species classification. Then we review the multi-instance multi-label framework for supervised classification.

A. Acoustic bird species classification

Brandes provides a survey of methods for acoustic bird species classification.[9] There are three main stages in most bird species classification systems: segmentation, feature construction, and supervised classification.

A syllable is a single short utterance by a bird, which may be a call, or part of a song. Methods for acoustic classification of bird species can broadly be grouped into those that classify individual syllables, and those that classify recordings containing multiple syllables. In both cases, segmenting audio into distinct syllables is a crucial step. The accuracy of any classifier that relies on segmentation is sensitive to the quality of the segmentation.[26]

Most algorithms for segmentation operate in the time domain, and are based on energy. It is common to compute the energy of the signal in each frame, then consider intervals with high energy to be syllables.[12,22,23,27]

Energy-based, time-domain segmentation is not well suited for audio with high-noise or multiple simultaneous birds. Energy-based segmentation accuracy degrades in high-noise recordings (e.g., from wind, stream noise, or motor vehicles), and also cannot differentiate other loud non-bird sounds. Vocalizations from multiple birds may overlap in time, making time-domain segmentation ineffective. Further work is needed to extract measurements from syllables that overlap in time but not frequency, and in high-noise environments.[9]

There has been some prior work on 2D time-frequency segmentation, which is better suited to audio with multiple simultaneous birds. Mellinger and Bradbury[28] used 2D segmentation in the form of bounding boxes for vocalizations of marine mammals, but this algorithm requires a human to provide a rough box first. Brandes divides the frequency range of a recording into several automatically determined bands, then applies a 2D energy threshold within each band.[29] We showed in earlier work that a random forest[30] classifier applied to each pixel of a spectrogram achieves higher segmentation accuracy than a 2D energy threshold on field-collected recordings.[31]

After running a segmentation algorithm to identify syllables, systems for bird species recognition extract acoustic features to characterize the syllables in a way that can be used with machine learning algorithms for classification. Linear predictive coding (LPC)[11,24,27] and Mel-frequency cepstral coefficients (MFCCs),[32,33] are common in analysis of speech and music and are amongst the most widely used features to describe bird sound.[12,21,34,35] Features such as LPCs and MFCCs describe individual frames of sound; to characterize a syllable as a whole, a common approach is to average the frame-level features.[12,21,22] Other features that have been used to characterize syllables include spectral peak tracking,[22,36,37] analysis-by-synthesis/overlap-add,[24] wavelets,[23] and "descriptive parameters" such as bandwidth, zero-crossing rate and spectral flux.[12,22]

The algorithms that have been applied to acoustic bird species classification either at the syllable or interval level (or both) all follow the standard single-instance, single-label framework (SISL) in machine learning, i.e., they associate a single feature vector with a single class label. SISL algorithms that have been applied to bird species classification include nearest-neighbor and distance based classifiers,[21,22,24,36] neural nets,[11,23,27,35] self-organizing maps,[23] decision trees,[13] support vector machines,[12] hidden Markov models,[10,15,22,29] and Gaussian mixture models.[22] We elaborate on the differences between SISL and MIML in the following section.

B. Multi-instance multi-label learning

In traditional supervised classification, we are given a collection of training examples, each of which consists of a feature vector and a class label. The goal is to learn from the training examples how to assign a class label to a previously unseen feature vector. However, in some applications, it is natural for the objects of interest to be represented as a collection of parts (referred to as a bag-of-instances), where each part is described by a fixed-length feature vector. Multi-instance learning[38] incorporates such structure into the classification model. For example, in multi-instance image classification, an image is a bag, and the instances are features describing pixels, patches or regions;[39] in multi-instance text classification, a document is a bag, and the instances correspond to paragraphs or sub-windows of text.[20,40] In this study, a short audio recording is a bag, and the instances correspond to 2D segments in the time-frequency domain described by a vector of their acoustic properties (these segments roughly correspond to syllables).

The original formulation of multi-instance learning[38] concerns problems where bags have single binary labels. Zhou[41] and Foulds and Frank[42] provide surveys on multi-instance learning, mainly focussing on the binary label case. Recently, Zhou and Zhang[20] proposed multi-instance multi-label (MIML) learning, where there a multiple classes and bags have a set of multiple labels.

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

Numerous algorithms for MIML have been proposed, and achieve superior accuracy in image and text domains to prior approaches that do not model the multi-instance or multi-label aspects of a problem explicitly.[39,43–45] Practical applications include labeling anatomical structures in images of *Drosophila* embryogenesis,[44] and predicting tags for web pages on a social bookmarking site.[45]

MIML has not previously been applied to audio, but Mandel and Ellis[46] recently applied multi-instance learning to classify music clips. Our proposed representation is at a different temporal scale, and is designed for bird sound rather than music.

A major advantage of MIML is that it is often easier or less costly to obtain labels at the bag-level. To the best of our knowledge, Brandes[29] is the only prior author to address acoustic species classification with multiple simultaneous species. In his work, the goal is to classify individual bird, frog, or cricket calls; this requires training data in the form of individually labeled calls. In contrast, because we use a MIML formulation, we predict a set of species present rather than the species for each vocalization. However, we only require a list of the species in each recording (bag) for training data.

The MIML framework is formalized as follows: Suppose we have a feature space $\mathcal{X}$ (usually $\mathcal{X} = \mathbb{R}^d$), and set of labels $\mathcal{Y} = \{1, \ldots, c\}$. In SISL learning, the training dataset is $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, where $\mathbf{x}_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$. A SISL classifier is a function $f : \mathcal{X} \to \mathcal{Y}$, i.e., it maps feature vectors to single class labels. In MIML, the dataset is $(X_1, Y_1), \ldots, (X_m, Y_m)$, where $X_i = \{\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,n_i}\}$ is a bag of n_i instances (i.e., feature vectors), and $Y_i \subseteq \mathcal{Y}$ is a set of labels. A bag can be considered a subset of the feature space, i.e., $X_i \subset \mathcal{X}$. A MIML classifier is a function $f : 2^{\mathcal{X}} \to 2^{\mathcal{Y}}$, i.e., it maps sets of feature vectors (bags) to sets of labels.

III. METHODS

We formulate the species identification problem in the MIML framework as follows: audio recordings are bags, segments in the spectrogram are instances, and the set of species in a recording are a bag's label set. We propose a bag generator to convert an audio recording into a bag-of-instances representation, then use a MIML classifier to predict the set of species present in the recording. The bag generator transforms an audio signal into a spectrogram, applies noise reduction, segments the spectrogram into 2D regions, then associates each region with a feature vector. After applying the bag generator, any MIML classifier can be used.

A. Bag generator

This section describes the noise reduction, segmentation, and features in our proposed bag generator.

1. Preprocessing and noise reduction

Starting from a 10-s recording sampled at 16 kHz, we transform it into a spectrogram by dividing the input signal into frames of 512 samples with 50% overlap, then computing the 256-element magnitude spectrum of each frame using the fast Fourier transform (FFT) with a Hamming window. We will denote the elements of the spectrogram as $S(t, f)$, where t indexes a frame and f corresponds to frequency (note f indexes an element of the discrete spectrum, i.e., $f \in \{1, 2, \ldots, f_{\max}\}$, where $f_{\max} = 256$; it is not in units of Hz).

To reduce noise and improve the contrast of bird sound, we first normalize $S(t, f)$ to the range $[0, 1]$, then compute $S_1(t, f) = \sqrt{S(t, f)}$ for all elements of the spectrogram. Then we apply two iterations of a whitening filter. The main idea is to estimate the frequency profile of noise from low energy frames, and then attenuate each row of the spectrogram according to this profile. The filter is:

(1) Compute a quantity similar to the energy of each frame t, as $E(t) = (1/f_{\max}) \sum_{f=1}^{f_{\max}} S_1(t, f)^2$. Sort the frames by E. Let the noise frames $N = \{t :$ frame t is one of the lowest 20% energy frames$\}$.
(2) For each frequency $f \in \{1, \ldots, f_{\max}\}$, compute $P(f) = \sqrt{\epsilon + \sum_{t \in N} S(t, f)^2}$, where $\epsilon = 10^{-10}$ (we add ϵ to avoid dividing by 0).
(3) For all (t, f) compute the noised reduced spectrogram as $S_2(t, f) = S_1(t, f)/P(f)$.

We will refer to the spectrogram resulting from two iterations of this process as $\hat{S}(t, f)$. Figure 1 shows a spectrogram before and after noise reduction.

There are some differences in how we compute $\hat{S}(t, f)$ for segmentation and feature construction; for segmentation, we apply the whitening filter once, define $P(f)$ as $(1/|N|) \sum_{t \in N} S(t, f)$, and do not apply the square root in S_1.

2. Segmentation

We use 2D time-frequency segmentation to separate syllables which may overlap in time. Rather than a 2D energy

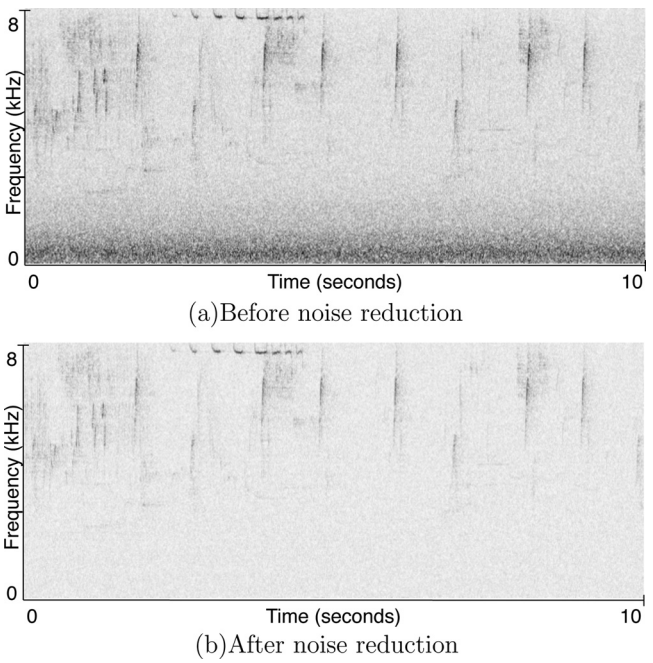
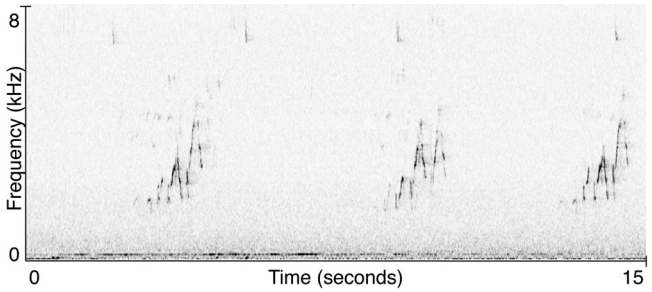
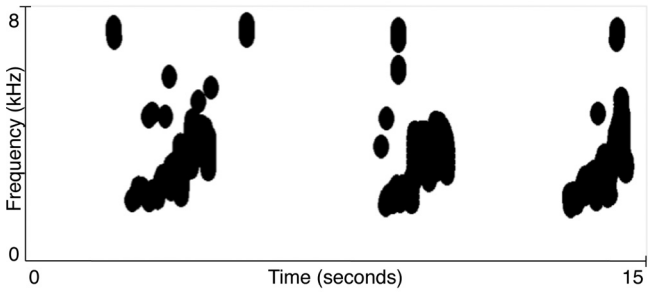


FIG. 1. An example showing noise reduction in a recording wind and stream noise.

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms



(a)A spectrogram in the segmentation training set.



(b)The corresponding manually generated time-frequency mask.

FIG. 2. An example of the manual segmentation that is used to train our supervised segmentation algorithm.

threshold,[29] we use a supervised SISL classifier to label each pixel in a spectrogram as bird sound or noise.[31] To do so, we associate each pixel in a spectrogram with a feature vector that describes a rectangular patch surrounding it. So for a particular (t, f) in the spectrogram, we compute its feature vector $\mathbf{x}(t, f)$ as follows.

(1) The spectrum-bin index f.
(2) The value of the elements of the spectrogram in a rectangle surrounding (t, f), i.e., $\hat{S}(i, j)$, $i \in [t - t_w, t + t_w]$, $j \in [f - f_w, f + f_w]$, where in our setup $t_w = 6$ and $f_w = 12$ (these values are manually tuned for the sampling frequency and window size used in our study).
(3) The variance of $\hat{S}$ in the same rectangle as above.

In order to train the classifier used for segmentation, we manually annotate a collection of spectrograms as examples of correct segmentation (Fig. 2). The mask $M(t, f)$ for spectrogram $\hat{S}(t, f)$ is defined as $M(t, f) = 0$ (white) if element (t, f) is background noise and $M(t, f) = 1$ (black) if it is bird sound. Recall that a SISL classifier (such as a random forest) takes as training data a list of pairs $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$. We form these pairs by selecting 500 000 points (t_i, f_i) at random within the manually annotated spectrograms. These points are sampled so there are 90% negative examples and 10% positive examples. For each point we compute the feature vector as described above, $\mathbf{x}_i = \mathbf{x}(t_i, f_i)$. The label for each training example is $y_i = M(t_i, f_i)$ (i.e., we have a two-class problem with labels 0 and 1). Then we train a random forest classifier[30] with 40 trees on this data (a random forest is an ensemble of decision trees).

Given an input $\mathbf{x}$, a random forest generates a probability $P(y|\mathbf{x})$ for the instance to belong to each class y, which is the fraction of trees in the forest that vote for label y given input $\mathbf{x}$. We use the random forest to compute the probability for each pixel (t, f) in the spectrogram to be bird sound, i.e., $P(y = 1|\mathbf{x}(t, f))$. Then we smooth these probabilities by convolving with a Gaussian kernel to obtain $g(t, f) = P(y = 1|\mathbf{x}(t, f)) * K$, where K is Gaussian kernel with $\sigma = 3$ over a 17×17 box. Finally, we obtain a predicted segmentation mask $M(t, f)$ for a spectrogram by applying a threshold of $\theta = 0.2$ to $g(t, f)$ (chosen by visual inspection of results with varying θ). Figure 3(b) shows an example of the predicted segmentation for one recording.

The random forest classifier discussed in this section is only used for segmentation; it is not directly involved in

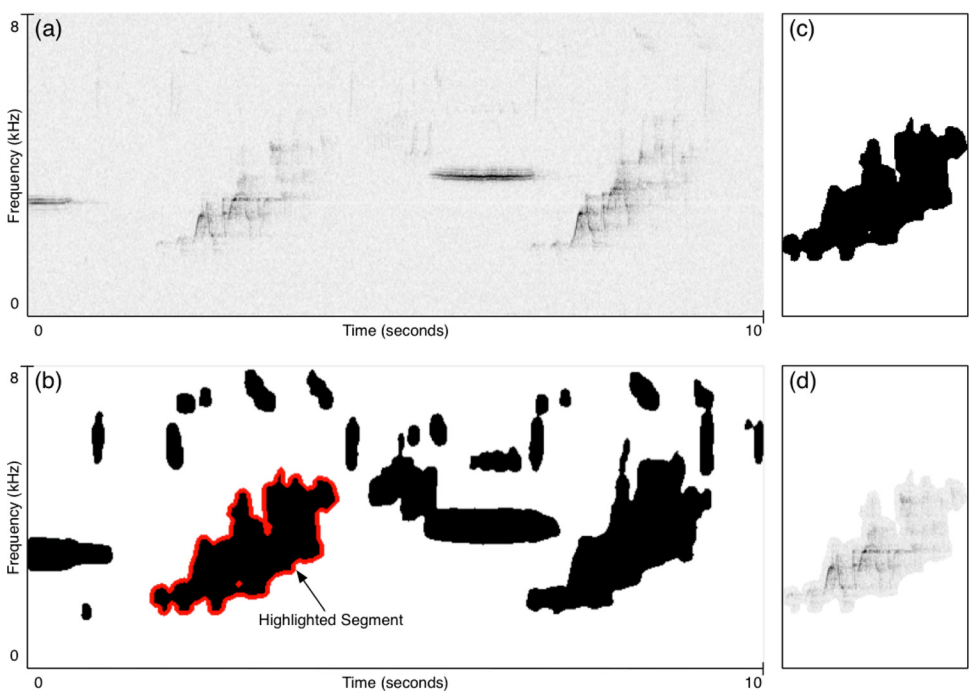


FIG. 3. (Color online) Extracting a syllable from the segmentation results. (a) The original spectrogram, (b) the binary mask generated by our segmentation algorithm. The highlighted segment will be further processed in this example. Note that several other segments overlap in time. (c) A cropped mask of the highlighted segment. (d) The masked and cropped spectrogram corresponding to the highlighted segment.

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

predicting the set of species in a recording (a MIML classifier does that instead). The collection of manually annotated spectrograms used to train the segmentation algorithm is disjoint from recordings for which we predict sets of species, i.e., training and testing data are separate.

3. Features

To compute features for each segment, we first crop the mask and spectrogram to contain just that segment. Figure 3 shows how one segment is cropped. Figure 3(a) shows the original spectrogram, and Fig. 3(b) shows the binary mask produced by our segmentation algorithm. For the sake of illustration, we highlight one segment. Figures 3(c) and 3(d) show a cropped image of the mask and spectrogram based on the highlighted segment.

The mask and spectrogram are cropped to the minimum number of frames to contain the whole segment in time, but not cropped at all in frequency. Note in Fig. 3(b), several other segments overlap in time with the highlighted segment. These overlapping segments are removed from the cropped mask [Fig. 3(c)]. The portions of the cropped spectrogram that are outside the mask are set to 0 [Fig. 3(d)]. The purpose of these two changes is to eliminate any contribution in the segment features from other segments that overlap in time, or noise which is outside of the segment mask.

We use the following notation in describing the segment features: Let $M_c(t,f)$ be the cropped, binary mask for a segment and let $\hat{S}_c(t,f)$ be the cropped, noise-reduced spectrogram. Note that t ranges from 1 to the duration of the segment in frames, T.

Three types of features describe a segment: mask descriptors, profile statistics, and histogram of gradients (HOG).[47] We depart from more commonly used audio features such as MFCCs because we are using 2D segmentation. The shape of the segment alone provides a lot of useful information, which the mask-based features capture. The profile statistics are similar to features that have previously been used for bioacoustics in noisy environments based on 2D segmentation.[28]

a. Mask descriptors. The first set of features that we compute for a segment are based on only the mask (i.e., not the contents of the spectrogram), and describe the shape of the segment. These features are

(1) $\textit{min-frequency} = \min\{f : M_c(t,f) = 1\}$.
(2) $\textit{max-frequency} = \max\{f : M_c(t,f) = 1\}$.
(3) $\textit{bandwidth} = \textit{max-frequency} - \textit{min-frequency}$.
(4) $\textit{duration} = T$.
(5) $\textit{area} = \sum_{tf} M_c(t,f)$.
(6) $\textit{perimeter} = \frac{1}{2} \times$ (# of pixels in M_c such that at least one pixel in the surrounding 3×3 box is 1 and at least one pixel is 0).
(7) $\textit{non-compactness} = \textit{perimeter}^2/\textit{area}$.
(8) $\textit{rectangularity} = \textit{area}/(\textit{bandwidth} \times \textit{duration})$.

b. Profile statistics. The next set of features that describe segments are based on statistical properties of the time and frequency profiles of the segment. To compute the time or frequency profile, we sum the columns or rows of the spectrogram. The time profile is $p_t(t) = \sum_f \hat{S}_c(t,f)$ and the frequency profile is $p_f(f) = \sum_t \hat{S}_c(t,f)$. We normalize the profiles to sum to 1, so they can be interpreted as probability mass functions. The normalized profile densities are $\hat{p}_t$ and $\hat{p}_f$. Two features measure the uniformity of these densities according the Gini index.[48]

(1) $\textit{freq-gini} = 1 - \sum_f \hat{p}_f(f)^2$.
(2) $\textit{time-gini} = 1 - \sum_t \hat{p}_t(t)^2$.

We obtain several more features by computing the kth central moments of the time and frequency profiles. However, because each segment may have a different duration, we compute these features in a re-scaled coordinate system where time goes from 0 to 1 over the duration of the segment, and frequency goes from 0 to 1.

(1) $\textit{freq-mean} = \mu_f = \sum_{f=1}^{f_{\max}} \hat{p}_f(f)(f/f_{\max})$.
(2) $\textit{freq-variance} = \sum_{f=1}^{f_{\max}} \hat{p}_f(f)(\mu_f - f/f_{\max})^2$.
(3) $\textit{freq-skewness} = \sum_{f=1}^{f_{\max}} \hat{p}_f(f)(\mu_f - f/f_{\max})^3$.
(4) $\textit{freq-kurtosis} = \sum_{f=1}^{f_{\max}} \hat{p}_f(f)(\mu_f - f/f_{\max})^4$.
(5) $\textit{time-mean} = \mu_t = \sum_{t=1}^{T} \hat{p}_t(t)(t/T)$.
(6) $\textit{time-variance} = \sum_{t=1}^{T} \hat{p}_t(t)(\mu_t - t/T)^2$.
(7) $\textit{time-skewness} = \sum_{t=1}^{T} \hat{p}_t(t)(\mu_t - t/T)^3$.
(8) $\textit{time-kurtosis} = \sum_{t=1}^{T} \hat{p}_t(t)(\mu_t - t/T)^4$.

In the same relative coordinate system, we compute the maxima of the time and frequency profiles.

(1) $\textit{freq-max} = \left(\arg\max \hat{p}_f(f)\right)/f_{\max}$.
(2) $\textit{time-max} = \left(\arg\max \hat{p}_f(t)\right)/T$.

We also include the mean and standard deviation of the spectrogram within the masked region.

(1) $\textit{mask-mean} = \mu_{tf} = (1/\textit{area}) \sum_{tf} \hat{S}_c(t,f)$.
(2) $\textit{mask-stddev} = \sqrt{(1/\textit{area}) \sum_{tf} (\mu_{tf} - \hat{S}_c(t,f))^2}$.

c. Histogram of gradients. To further characterize the shape and texture of each segment, we include a HOG feature similar to the work of Dalal and Triggs.[47] As input, we take the cropped spectrogram and mask for a segment $\hat{S}_c(t,f)$ and $M_c(t,f)$. First, the spectrogram is blurred by convolving with a 7×7 Gaussian kernel G with $\sigma^2 = 4$ to obtain $S_b(t,f) = \hat{S}_c(t,f) * G$. The gradients at a point (t,f) are computed by convolving a Sobel kernel with with S_b, i.e., $(d/dx)S_b(t,f) = S_b(t,f) * D_x$ and $(d/dy)S_b(t,f) = S_b(t,f) * D_y$, where

$$D_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

and $D_y = D_x^T$. Then, for each pixel of the spectrogram that is in the mask [i.e., $M_c(t,f) = 1$], we compute $\nabla S_b(t,f) = ((d/dx)S_b(t,f), (d/dy)S_b(t,f))$. Only pixels such that $||\nabla S_b(t,f)||^2 \geq 0.01$ contribute to the histogram. The histogram consists of 16 bins evenly spaced over the range of angles $[0, 2\pi]$. The feature vector for a segment consists of the normalized count, for each bin, of the number of gradients belonging to that bin. Hence we obtain a 16 dimensional HOG feature for each segment.

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

d. Feature rescaling. All of the features described above are concatenated to form a single feature vector describing each segment. These features differ widely in the range of values they can have. This property of the features can bias distance-based classifiers such as MIML-*k*NN to place more weight on features with larger magnitudes. To prevent this bias, we rescale each feature independently to the range [0,1].

B. MIML classifiers

Using our bag generator, we experimentally evaluate three MIML algorithms: MIMLSVM,[20] MIMLRBF,[49] and MIML-*k*NN.[50] These algorithms reduce the MIML problem to a single-instance multi-label problem by associating each bag with a bag-level feature, which aggregates information from the instances in the bag. Hence the MIML dataset $\{(X_i, Y_i)\}_{i=1}^m$ is transformed into a single-instance multi-label dataset $\{(\mathbf{z}_i, Y_i)\}_{i=1}^m$ where $\mathbf{z}_i \in \mathbb{R}^d$ is the bag-level feature for bag i. Each algorithm constructs a different bag-level feature, but all use some form of bag-level distance measure. The maximal and average Hausdorff distances between two bags $X = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ and $X' = \{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$ are defined as

$$D_H^{\max}(X, X') = \max\{\max_{\mathbf{a}\in X} \min_{\mathbf{b}\in X'} ||\mathbf{a} - \mathbf{b}||, \max_{\mathbf{b}\in X'} \min_{\mathbf{a}\in X} ||\mathbf{b} - \mathbf{a}||\},$$

$$D_H^{\text{avg}}(X, X') = \left(\sum_{\mathbf{a}\in X} \min_{\mathbf{b}\in X'} ||\mathbf{a} - \mathbf{b}|| + \sum_{\mathbf{b}\in X'} \min_{\mathbf{a}\in X} ||\mathbf{b} - \mathbf{a}||\right) \Big/ (|X| + |X'|).$$

MIMLSVM applies k-medoids clustering to the training dataset of bags using $D_H^{\max}$. This clustering produces k medoid bags $M_1, \ldots, M_k$. For each bag (X_i, Y_i), a bag-level feature is computed as $\mathbf{z}_i = (D_H^{\max}(X_i, M_1), \ldots, D_H^{\max}(X_i, M_k))$. The resulting multi-label classification problem is solved using the MLSVM algorithm, which consists of building one support vector machine (SVM) for each class.

MIMLRBF runs k-medoids clustering once for each class using D_H^{avg} on the set of bags including that class as a label (the parameter k is different for each class). Concatenating the medoids obtained in each clustering, there are q medoid bags, $B_1, \ldots, B_q$. Then each bag X_i is associated with a feature $\mathbf{z}_i = (1, K(X_i, B_1), \ldots, K(X_i, B_q))$, where $K(X, X') = \exp(-D_H^{\text{avg}}(X, X'^2/2\sigma^2))$. The resulting multi-label classification problem is solved using one linear model per class, trained by minimizing sum squared error.

MIML-*k*NN also assigns bag-level features, but does so using an approach inspired by nearest-neighbors rather than clustering. For each training bag X_i, MIML-*k*NN finds its k nearest neighbors, and k' citers (other bags that consider X_i to be one of their k' nearest neighbors), using D_H^{avg}. Then each bag X_i is associated with a bag-level feature vector $\mathbf{z}_i = (t_1, \ldots, t_c)$, where t_j is the number of bags in the neighbors and citers of X_i that include class j in their label set. The resulting multi-label problem is solved using the same approach as MIMLRBF.

IV. EXPERIMENTS

We apply the proposed methods to field-collected audio from the H. J. Andrews (HJA) Experimental Forest. These experiments demonstrate that our methods accurately predict the set of species present in an unattended acoustic monitoring scenario.

A. Data collection and labeling

To collect audio in HJA, we use 13 Wildlife Acoustics Song Meter SM1 recording devices. These devices have two omnidirectional microphones enclosed in wind shields protruding from a weather resistant enclosure that houses batteries, a computer, and 32 Gb flash-memory for data storage.

The audio is recorded at 16 kHz. The result of applying the FFT is a spectrogram with frequencies from 0–8 kHz. This range is sufficient to capture most bird sounds in HJA. For example, the Hermit Warbler is one of the highest pitched species in HJA, and is generally below 8 kHz.[51] It is possible that some bird sounds are omitted due to this sampling frequency, but the proposed methods still work well for the species that we identified.

In order to train and evaluate algorithms to predict which species of birds are present in a recording, it is necessary to have some labeled examples. We have months of audio in total, so it would not be feasible to manually label all of it. Accordingly, we focus on a representative sample of 548 10-s recordings from six sites, all within the range of 5:00 am to 5:20 am (birds are highly active at this time of day), on 5/31/2009. Many of the recordings include multiple bird species vocalizing simultaneously. We manually identified the set of species that are present in each 10-s recording.

There are 13 bird species in the recordings examined (Table I). Each recording contains between 1 and 5 species. There are 2.144 species per recording on average.

For the purpose of MIML experiments, we assume that recordings that do not contain any bird sounds can be detected during segmentation, hence we only include recordings that contain at least 1 species vocalizing. We evaluated segmentation on recordings that do not contain bird sound in prior work.[31]

TABLE I. The number of ten-second recordings containing each species in our labeled dataset.

Code	Name	#
PSFL	Pacific-slope Flycatcher (*Empidonax difficilis*)	165
HAFL	Hammond's Flycatcher (*Empidonax hammondii*)	103
OSFL	Olive-sided Flycatcher (*Contopus cooperi*)	90
HETH	Hermit Thrush (*Catharus guttatus*)	15
VATH	Varied Thrush (*Ixoreus naevius*)	89
SWTH	Swainson's Thrush (*Catharus ustulatus*)	79
GCKI	Golden-crowned Kinglet (*Regulus satrapa*)	197
PAWR	Pacific Wren (*Troglodytes pacificus*)	109
RBNU	Red-breasted Nuthatch (*Sitta canadensis*)	82
DEJU	Dark-eyed Junco (*Junco hyemalis*)	20
CBCH	Chestnut-backed Chickadee (*Poecile rufescens*)	117
HEWA	Hermit Warbler (*Setophaga occidentalis*)	63
WETA	Western Tanager (*Piranga ludoviciana*)	46

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

Note that a small fraction of the recordings contain only a single species (but still multiple syllables). Two out of the 13 species have recordings of this kind. It is not necessary for MIML to have multiple labels associated with every bag in the training set. Bags with a single label provide less ambiguous information, and should therefore be expected to improve accuracy.

There are 10 232 instances (audio segments) in our dataset. It is considerably more laborious to label the instances than the 548 bags. For the purpose of comparison to SISL methods, we have manually labeled 4998 of these instances. Of the remaining 5234 unlabeled instances, a substantial fraction are segmentation errors or noise, or faint sounds that are very difficult for a human to identify. These instance labels are only used for evaluation of SISL; they are not used by the MIML algorithms.

In addition to the 548 10-s recordings that we labeled with species sets, we also manually segmented 625 disjoint 15-s recordings that are used as examples to train the segmentation algorithm. These recordings are selected from 13 sites, over a period from 5/2/2009 to 7/4/2009, with some examples from each hour of the day. Within these 625 examples, 334 contain some bird sound, and 291 contain only noise.

It is not crucial that the training recordings be 10 or 15 s. We can provide some intuition for the choice of duration. Increasing the duration of the recordings used for training the segmentation algorithm reduces the number of syllables in the training data that are cut off by the boundary of the recording. However, as the duration of recordings used for MIML is increased, it becomes more likely for a bag to include all of the species at the recording site. In the extreme, a bag is labeled with every species, in which case no learning is possible because the labels are completely ambiguous.

All of the data collection sites are within 1 km of a stream. Hence, all recordings contain some stream noise, as well as wind or insects in some cases.

B. Evaluation

1. Cross-validation

We use five-fold cross-validation to evaluate each MIML algorithm on the collection of 548 species-labeled recordings. The recordings/bags are randomly partitioned into five disjoint sets (each set contains some examples from every species). For each fold, four of the sets are used as training data for the MIML algorithms, and the remaining set is used for testing. Test performance is aggregated over the five folds.

Because we use data from six sites over a 20-min interval of time, it is likely that the MIML classifier is trained and tested on vocalizations from the same individual birds. We expect that prediction accuracy would decrease in an experiment where the classifier is applied to individuals that do not appear in the training set.

2. Accuracy measures

Several measures common in multi-label and MIML experiments characterize the accuracy of each algorithm, namely, Hamming loss, rank loss, one-error, coverage,[39] and micro-AUC.[52] A MIML classifier outputs a set of classes, but many implementations first output a score for each class, which is compared to a threshold to obtain the set. Several of the accuracy measures use these scores. We denote the score for class j given by a MIML classifier f on input bag X as $f_j(X)$. The set of predicted labels which is obtained from the scores is denoted $f(X)$. Also let $I[\cdot]$ denote the indicator function. Recall that the number of classes (species) is c, and the number of bags is n. The accuracy measures are defined as follows.

Hamming loss does not rely on the scores for each class, but instead directly evaluates the predicted set. It is the number of false positives and false negatives, averaged over the number of classes and bags,

$$\frac{1}{nc}\sum_{i=1}^{n}\sum_{j=1}^{c} I[j \in f(X_i),\ j \notin Y_i] + I[j \notin f(X_i),\ j \in Y_i].$$

Rank loss captures the number of label pairs that are incorrectly ordered by the scores of the MIML classifier (i.e., classes that are in the true label set should receive higher scores than classes that are not). Let $\bar{Y}$ denote the complement of Y. Rank loss is defined as

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|Y_i||\bar{Y}_i|}\sum_{j\in Y_i,k\in\bar{Y}_i} I[f_j(X_i) \le f_k(X_i)].$$

One-error is the fraction of bags for which the top scoring label is not in the true label set,

$$\frac{1}{n}\sum_{i=1}^{n} I[(\arg\max_{j\in y} f_j(X_i)) \notin Y_i].$$

The scores for all classes can be ranked, so that rank 1 is the most likely to be present (highest score), rank 2 is the next most likely, and rank c is the least likely. We denote the rank of class j given input bag X as $\mathrm{rank}(X, j)$. Coverage measures the how far down the ranking one must go to get all of the true labels,

$$\frac{1}{n}\sum_{i=1}^{n}\max_{j\in Y_i}\{\mathrm{rank}(X_i, j) - 1\}.$$

MIMLSVM, MIMLRBF, and MIML-kNN output signed scores for each class, with a positive score indicating a class is present, and a negative score indicating it is absent. To compute Hamming loss, we use a threshold of 0. However, varying this threshold can be used to control the tradeoff between predicting species which are not present (i.e., false positives), or failing to detect species which are present (i.e., false negatives). The receiver operating characteristic (ROC) curve captures this tradeoff. Let the predicted label set for a bag X_i using a threshold t be $f(X_i, t)$. Define true/false positives/negatives as

$$\mathrm{TP} = \sum_{i=1,\ldots,n, j\in Y_i} I[j \in f(X_i,t)],\ \mathrm{FP} = \sum_{i=1,\ldots,n, j\in\bar{Y}_i} I[j \in f(X_i,t)],$$
$$\mathrm{TN} = \sum_{i=1,\ldots,n, j\in\bar{Y}_i} I[j \notin f(X_i,t)],\ \mathrm{FN} = \sum_{i=1,\ldots,n, j\in Y_i} I[j \notin f(X_i,t)].$$

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

The true positive rate is $\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$ and the false positive rate is $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$. Each point on the ROC curve (Fig. 1) corresponds to a pair (TPR,FPR) for one threshold. The area under this ROC curve is called micro-AUC (in contrast with macro-AUC, which is the average AUC of the separate ROC curves for each class).[53]

3. Parameter tuning

Each MIML algorithm has several parameters that can be tuned to improve accuracy. We evaluate each algorithm over all combinations of parameter values in a range, and report results corresponding to the parameter setting of each algorithm that minimizes Hamming loss. The parameters and ranges are as follows.

(1) For MIMLSVM, the parameters are (C, γ, r). The parameter C controls SVM regularization. The SVM uses a Gaussian kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma||\mathbf{x}-\mathbf{y}||^2)$. The parameter k for k-medoids clustering is set to nr, where n is the number of bags. We evaluate all combinations of $(C, \gamma, r) \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}^2 \times \{0.2, 0.4, 0.6, 0.8\}$.
(2) For MIMLRBF, the parameters are (r, μ). k-medoids clustering repeats once for each class; in the clustering for class i, k is set to $\nu_i r$, where ν_i is the number of bags including label i. MIMLRBF constructs bag-level features by applying a function $K(X, X') = \exp(-D_H^{\mathrm{avg}}(X, X')^2/2\sigma^2)$. The parameter σ in this expression is set to μ times the average Hausdroff distance between clusters.[49] We evaluate all combinations of $(r, \mu) \in \{10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}^2$.
(3) For MIML-kNN, we vary the number of neighbors k and the number of citers k' over $(k, k') \in \{5, 10, 20, 30\}^2$.

V. RESULTS

Table II lists the accuracy measures for each MIML algorithm. To better interpret these results, we discuss the ranges of values the accuracy measures can take, and compare to values for baseline classifiers.

Hamming loss, rank loss, and one-error have values in the range $[0, 1]$ with 0 corresponding to perfect prediction. Let the average number of labels per bag be $m = (1/n)\sum_{i=1}^{n}|Y_i|$. Then the best possible coverage is $m-1$, and the worst possible coverage is $c-1$ (for our dataset, this gives the range $[1.144, 12]$). However, in order to achieve the worst possible values for these measures, it is necessary to make predictions that are worse than random.

To obtain a baseline for Hamming loss, consider a non-informative classifier that always predicts the empty set. The Hamming loss is $m/c = 0.1649$, because each label in the true label set is a false-negative. The other measures are based on class ranks, so consider a non-informative classifier that outputs uniformly random scores for each class, or equivalently, ranks classes in a random order. The probability that the top-scoring class will be one of the $|Y_i|$ labels for bag i is $|Y_i|/c$, so the expected one-error is $1-(1/n)\sum_{i=1}^{n}(|Y_i|/c) = 1-(m/c) = 0.8351$. Because $P(f_j(X_i) \le f_k(X_i)) = \frac{1}{2}$, the expected rank loss is $\frac{1}{2}$. The AUC for a random classifier is $\frac{1}{2}$ as well.[54] We approximate the expected coverage for a non-informative classifier by averaging the coverage for 10 000 random orders.

We also compute the rank loss, one-error, and coverage for a classifier that ignores its input, and outputs the ranking from most frequent class to least frequent, which is a stronger baseline than random ranking (Table II).

All of the MIML classifiers are closer to perfect prediction than to non-informative or frequency order baselines. For example, the Hamming loss for MIML-kNN is 4.23 times lower than non-informative. With a rank loss of 0.019, MIML-kNN is 26.31 times less likely to incorrectly rank a present/absent species pair than a random classifier. A one-error of 0.034 means that if we only predict the highest scoring species in each recording, it will truly be present 96.6% of the time. MIML-kNN achieves a Hamming loss of 0.039, which is equivalent to a true positive/negative rate of 96.1% (the fraction of true positive/negatives is 1 − Hamming loss). To give a concrete view of the predictions, we show results for 20 randomly selected recordings using MIML-kNN in Table III.

Recent work[5] has highlighted the importance of accounting for imperfect detectability of species in wildlife surveys. Due to the massive amount of survey time enabled by continuous recordings, our proposed methods can help to reduce false negatives typical of manual bird surveys. The ROC curves (Fig. 4) show that we can set a threshold which achieves a low false positive rate, while still retaining a relatively high true positive rate, thus meeting critical assumptions for occupancy analysis.

A. Comparison to SISL

It is difficult to make a direct comparison between MIML and SISL, because MIML and SISL algorithms make different types of predictions, and are evaluated according to

TABLE II. Accuracy measures for MIML classifiers and baselines (— indicates the result cannot be calculated).

Algorithm	Hamming loss ↓	Rank loss ↓	One-error ↓	Coverage ↓	Micro-AUC ↑
MIMLSVM	0.054	0.033	0.067	1.844	0.966
MIML-kNN	**0.039**	**0.019**	0.036	**1.589**	0.962
MIMLRBF	0.049	0.022	0.034	1.632	**0.978**
Non-informative	0.165	0.5	0.8351	8.068	0.5
Frequency order	—	0.318	0.698	5.901	—
SISL random forest	0.125	0.050	0.084	2.201	0.949
SISL random forest filtered	0.049	0.023	**0.022**	1.708	0.974

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

TABLE III. Example predictions with MIML-*k*NN.

Ground truth	Predicted labels
PAWR, PSFL	GCKI, PAWR, PSFL
VATH, SWTH	VATH, HEWA, SWTH
OSFL, CBCH	GCKI, OSFL, CBCH
CBCH	GCKI, CBCH
HAFL	HAFL
VATH, HEWA	VATH, HEWA
GCKI, PSFL, RBNU, DEJU	GCKI, PSFL
GCKI, PAWR, PSFL	GCKI
GCKI, OSFL	GCKI, OSFL
GCKI, PAWR, PSFL	GCKI, PAWR, PSFL
SWTH	SWTH
VATH, HEWA, SWTH	VATH, HEWA, SWTH
GCKI, OSFL, HETH	GCKI
GCKI, OSFL	GCKI, OSFL
GCKI, PAWR, PSFL	GCKI, PAWR
SWTH	
GCKI, PAWR, PSFL	GCKI, PSFL
GCKI, PSFL, OSFL	GCKI, PSFL
HAFL	HAFL
CBCH, SWTH	CBCH, SWTH

different performance measures. We compare to a model based on a random forest SISL classifier because the random forest achieves high accuracy in many domains, and has only one parameter (the number of trees), to which it is not very sensitive. Using the 4998 labeled instances in the dataset, we train a SISL random forest with 100 trees. We use the same folds for five-fold cross validation as the MIML algorithms. For each fold, the labeled instances in four of the sets are used to train a random forest, then the instances in the remaining set are used to compute MIML performance measures.

We need to compute bag-level outputs to evaluate the random forest using MIML performance measures. To do so, we compute the probabilities for every instance in a bag to belong to each class, then define the bag-level score for each class as the maximum instance probability for that class, i.e., the bag-level score for class j given input bag X is $f_j(X) = \max_{\mathbf{x}\in X} P(j|\mathbf{x})$. This formulation of the bag-level model is similar to MIML algorithms including M^{3}MIML,[55] D-MIMLSVM,[39] and TMIML.[56] We compute the Hamming loss for the random forest using a threshold of 0.5 on the bag-level scores. The other performance measures are computed directly from the scores.

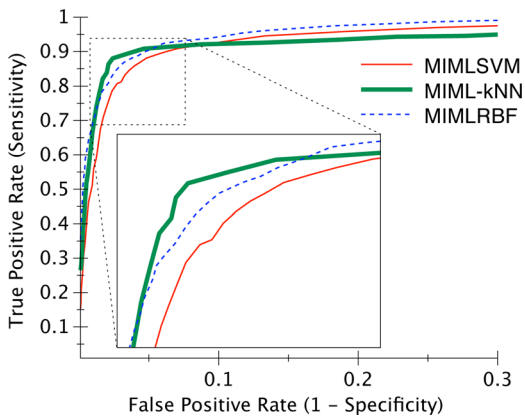


FIG. 4. (Color online) ROC curves for each algorithm.

This SISL-trained model is worse in every performance measure than the MIML algorithms (Table II; SISL random forest). The bag-level scores are computed using all of the instances in the bag, including the unlabeled instances which are more likely to correspond to noise that is mislabeled as bird sound in the segmentation stage. Such instances bias the bag-level scores to generate many false positives (with a threshold of 0.5 there are 740 false positives and 148 false negatives). We can improve the results for the random forest by filtering out all of the unlabeled instances so they do not influence the bag-level scores (Table II; SISL random forest filtered). The MIML algorithms in this study do not require this filtering to produce accurate results because they do not depend on instance-level predictions (this is not true of all MIML algorithms). However filtering would improve the performance of the MIML algorithms.

Even when we give the SISL model the advantages of having instance labels, and of filtering out the unlabeled instances, the best MIML results are better than the filtered SISL results in all measures except for one-error. This result suggests that it is non-trivial to incorporate instance labels into a bag-level model.

VI. CONCLUSION AND FUTURE WORK

We formulate the problem of detecting the set of bird species present in an audio recording using the MIML framework, and propose a method to transform an audio recording into a representation suitable for using with MIML algorithms. Using data collected in the field with omnidirectional microphones, we showed that the proposed methods achieve high accuracy.

This work is a step toward automatic unattended acoustic surveys of bird populations. In future work, we seek to classify all bird sounds in a much larger collection of audio (over 4 TB), representing two years of recordings in the field. This will effectively generate a presence/absence population survey at the sites where we have deployed recording devices. In contrast with manual surveys, automated acoustic surveys can provide high temporal resolution over the long term. For example, it would not be reasonable for a person to count birds once per minute, 24 h a day, for three months, but we aim to obtain similar results with acoustic surveys. Such data is likely to provide new insights into bird behavior, and their interaction with the environment.[57]

MIML classifiers can only predict labels that appear in their training data, and cannot detect when something does not belong to one of the training classes. Hence it is not clear how to handle unexpected sounds. Due to the high-noise environment, and birds vocalizing far from the microphone, it is often difficult for a human labeler to determine all of the species present in a recording. Consequently, some of the segments/instances may come from species that are not present in the training label set. Furthermore, some of the instances are segmentation errors capturing noise rather than bird

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

sound. Further work is needed on classes not present in the training data, incomplete label sets, and noise instances.

One may wish to predict the species of each individual segment/instance, rather than just the set of species in a recording. If individually labeled segments are available for training data, this is the standard SISL supervised classification problem. However, we instead focus on the situation where it is difficult or expensive to obtain such labels. Learning to predict instance labels from MIML training data is a different problem, known as instance annotation, which has received little study so far. One might also wonder if the accuracy of MIML predictions could be improved by including individually labeled instances in the training data (e.g., recordings containing only a single species and syllable). This problem of mixed-granularity training has also received little study. In prior work on MIML, this issue has been handled by using an unmodified MIML algorithm with a bag containing a single instance.[58]

Although we focus on birds, MIML may also be applicable to analysis of other bioacoustic signals from animals including grasshoppers,[14] crickets, frogs,[29] and marine mammals,[28] and computational acoustic scene analysis in general. Aside from its ecological applications, this work broadens the scope of MIML domains from text and images to include audio.

ACKNOWLEDGMENTS

This work was partially funded by the Ecosystems Informatics IGERT program via NSF Grant No. DGE 0333257, NSF-CDI Grant No. 0941748 to M.G.B., NSF Grant No. 1055113 to X.Z.F., and the College of Engineering, Oregon State University. We conducted this research at H. J. Andrews Experimental Forest, which is funded by the US Forest Service, Pacific Northwest Research Station. We would also like to thank Jay Sexsmith for his help in collecting data, Iris Koski for labeling the data, Katie Wolf for her work on noise reduction, and Dave Mellinger for his help editing.

[1]A. Balmford, R. Green, and M. Jenkins, "Measuring the changing state of nature," Trends Ecol. Evol. **18**, 326–330 (2003).
[2]C. Parmesan and G. Yohe, "A globally coherent fingerprint of climate change impacts across natural systems," Nature **421**, 37–42 (2003).
[3]Ç. Şekercioğlu, G. Daily, and P. Ehrlich, "Ecosystem consequences of bird declines," Proc. Natl. Acad. Sci. U.S.A. **101**, 18042 (2004).
[4]M. G. Betts, D. Mitchell, A. W. Diamond, and J. Bety, "Uneven rates of landscape change as a source of bias in roadside wildlife surveys," J. Wildl. Manage. **71**, 2266–2273 (2007).
[5]D. MacKenzie, J. Nichols, G. Lachman, S. Droege, J. Andrew Royle, and C. Langtimm, "Estimating site occupancy rates when detection probabilities are less than one," Ecology **83**, 2248–2255 (2002).
[6]C. Robbins, S. Droege, and J. Sauer, "Monitoring bird populations with breeding bird survey and atlas data," Ann. Zool. Fenn. **26**, 297–304 (1989).
[7]D. Luther, "Signaller: Receiver coordination and the timing of communication in Amazonian birds," Biol. Lett. **4**, 651 (2008).
[8]D. Luther and R. Wiley, "Production and perception of communicatory signals in a noisy environment," Biol. Lett. **5**, 183 (2009).
[9]T. Scott Brandes, "Automated sound recording and analysis techniques for bird surveys and conservation," Bird Conserv. Int. **18**, 163–173 (2008).
[10]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," J. Acoust. Soc. Am. **103**, 2185–2196 (1998).
[11]A. McIlraith and H. Card, "Birdsong recognition using backpropagation and multivariate statistics," IEEE Trans. Signal Process. **45**, 2740–2748 (1997).
[12]S. Fagerlund, "Bird species recognition using support vector machines," J. Adv. Signal Process. **2007** (2007).
[13]E. Vilches, I. Escobar, E. Vallejo, and C. Taylor, "Data mining applied to acoustic bird species recognition," Pattern Recogn. **3**, 400–403 (2006).
[14]E. Chesmore and E. Ohya, "Automated identification of field-recorded songs of four British grasshoppers using bioacoustic signal recognition," Bull. Entomol. Res. **94**, 319–330 (2004).
[15]V. Trifa, A. Kirschel, C. Taylor, and E. Vallejo, "Automated species recognition of antbirds in a Mexican rainforest using hidden Markov models," J. Acoust. Soc. Am. **123**, 2424–2431 (2008).
[16]S. Anderson, A. Dave, and D. Margoliash, "Template-based automatic recognition of birdsong syllables from continuous recordings," J. Acoust. Soc. Am. **100**, 1209–1219 (1996).
[17]W. Chu and D. Blumstein, "Noise robust bird song detection using syllable pattern-based hidden Markov models," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (2011).
[18]R. Bardeli, D. Wolff, F. Kurth, M. Koch, K. Tauchert, and K. Frommolt, "Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring," Pattern Recogn. Lett. **31**, 1524–1534 (2009).
[19]P. Jancovic and M. Köküer, "Automatic detection and recognition of tonal bird sounds in noisy environments," J. Adv. Sign. Process. **2011**, 1–10 (2011).
[20]Z. Zhou and M. Zhang, "Multi-instance multi-label learning with application to scene classification," Adv. Neural Inf. Process. Syst. **19**, 1609 (2007).
[21]C.-H. Lee, Y.-K. Lee, and R.-Z. Huang, "Automatic recognition of bird songs using cepstral coefficients," J. Inf. Technol. Appl. **1**, 17–23 (2006).
[22]P. Somervuo, A. Härmä, and S. Fagerlund, "Parametric representations of bird sounds for automatic species recognition," IEEE Trans. Audio, Speed, Lang. Process. **14**, 2252–2263 (2006).
[23]A. Selin, J. Turunen, and J. Tanttu, "Wavelets in recognition of bird sounds," J. Adv. Signal Process. **2007**, 1–9 (2007).
[24]Z. Chen and R. C. Maher, "Semi-automatic classification of bird vocalizations using spectral peak tracks," J. Acoust. Soc. Am. **120**, 2974–2984 (2006).
[25]See supplementary material at http://dx.doi.org/10.1121/1.4707424 for sample audio recordings and spectrograms.
[26]S. Fagerlund, "Automatic recognition of bird species by their sounds," Ph.D. thesis, Helsinki University of Technology, Helsinki, 2004.
[27]C. Juang and T. Chen, "Birdsong recognition using prediction-based recurrent neural fuzzy networks," Neurocomputing **71**, 121–130 (2007).
[28]D. Mellinger and J. W. Bradbury, "Acoustic measurement of marine mammal sounds in noisy environments," in *Proceedings of the International Conference on Underwater Acoustical Measurements: Technologies and Results* (2007), pp. 273–280.
[29]T. Brandes, "Feature vector selection and use with hidden Markov models to identify frequency-modulated bioacoustic signals amidst noise," IEEE Trans. Audio, Speech, Lang. Process. **16**, 1173–1180 (2008).
[30]L. Breiman, "Random forests," Mach. Learn. **45**, 5–32 (2001).
[31]L. Neal, F. Briggs, R. Raich, and X. Fern, "Time-frequency segmentation of bird song in noisy acoustic environments," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (2011).
[32]S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," in *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing* (1980), Vol. 28, pp. 357–366.
[33]J. Volkmann, S. S. Stevens, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," J. Acoust. Soc. Am. **8**, 208–208 (1937).
[34]C. Kwan, G. Mei, X. Zhao, Z. Ren, R. Xu, V. Stanford, C. Rochet, J. Aube, and K. Ho, "Bird classification algorithms: Theory and experimental results," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (2004), Vol. 5, pp. 289–292.
[35]J. Cai, D. Ee, B. Pham, P. Roe, and J. Zhang, "Sensor network for the monitoring of ecosystem: Bird species recognition," in *3rd International Conference on Intelligent Sensors, Sensor Networks and Information* (2008), pp. 293–298.
[36]A. Härmä, "Automatic identification of bird species based on sinusoidal modeling of syllables," in *Proceedings of the IEEE International Confer-*

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms

ence on Acoustics, Speech, and Signal Processing (2003), Vol. 5, pp. 545–548.
[37]A. Härmä and P. Somervuo, "Classification of the harmonic structure in bird vocalization," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (2004), Vol. 5, pp. 701–704.
[38]T. Dietterich, R. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," Artif. Intell. **89**, 31–71 (1997).
[39]Z. Zhou, M. Zhang, S. Huang, and Y. Li, "MIML: a framework for learning with ambiguous objects," arXiv:0808.3231.
[40]S. Yang, H. Zha, and B. Hu, "Dirichlet-Bernoulli alignment: a generative model for multi-class multi-label multi-instance corpora," Adv. Neural Inf. Process. Syst. **9**, 2143–2150 (2010).
[41]Z. Zhou, "Multi-instance learning: A survey," Technical Report, AI Lab, Department of Computer Science and Technology, Nanjing University, 2004.
[42]J. Foulds and E. Frank, "A review of multi-instance learning assumptions," Knowledge Eng. Rev. **25**, 1–25 (2010).
[43]Z. Zha, X. Hua, T. Mei, J. Wang, G. Qi, and Z. Wang, "Joint multi-label multi-instance learning for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition* (2008), pp. 1–8.
[44]Y. Li, S. Ji, S. Kumar, J. Ye, and Z. Zhou, "Drosophila gene expression pattern annotation through multi-instance multi-label learning," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence* (2009).
[45]C. Shen, J. Jiao, B. Wang, and Y. Yang, "Multi-instance multi-label learning for automatic tag recommendation," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics* (2009).
[46]M. Mandel and D. Ellis, "Multiple-instance learning for music information retrieval," in *Proceedings of the International Symposium on Music Information Retrieval* (2008).
[47]N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition* (2005), Vol. 1, pp. 886–893.
[48]C. Gini, "Variability and mutability, contribution to the study of statistical distributions and relations," J. Am. Stat. Assoc. **66**, 534–544 (1971).
[49]M. Zhang and Z. Wang, "MIMLRBF: RBF neural networks for multi-instance multi-label learning," Neurocomputing **72**, 3951–3956 (2009).
[50]M. Zhang, "A k-nearest neighbor based multi-instance multi-label learning algorithm," in *22nd IEEE International Conference on Tools with Artificial Intelligence* (2010), pp. 207–212.
[51]S. F. Pearson, "Hermit warbler (*Setophaga occidentalis*), the birds of North America online (1997)," retrieved from Birds of North America Online: http://bna.birds.cornell.edu/bna/species/303 (last viewed January 26, 2012).
[52]A. Dimou, G. Tsoumakas, V. Mezaris, I. Kompatsiaris, and I. Vlahavas, "An empirical study of multi-label learning methods for video annotation," in *7th International Workshop on Content-Based Multimedia Indexing* (2009), pp. 19–24.
[53]D. Lewis, "Evaluating text categorization," in *Proceedings of the Speech and Natural Language Workshop* (1991), pp. 312–318.
[54]A. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," Pattern Recog. **30**, 1145–1159 (1997).
[55]M. Zhang and Z. Zhou, "M3MIML: A maximum margin method for multi-instance multi-label learning," in *8th IEEE International Conference on Data Mining* (2008), pp. 688–697.
[56]S. Images, "Transductive Multi-Instance Multi-Label learning algorithm with application to automatic image annotation," Expert Syst. Appl. **37**, 661–670 (2009).
[57]H. Slabbekoorn and T. Smith, "Bird song, ecology and speciation," Philos. Trans. R. Soc. London, Ser. B **357**, 493 (2002).
[58]S. Vijayanarasimhan and K. Grauman, "What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations," in *IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 2262–2269.

Downloaded 23 Feb 2013 to 216.165.95.64. Redistribution subject to ASA license or copyright; see http://asadl.org/terms