

Species-specific audio detection: a comparison of three template-based detection algorithms using random forests

Carlos J. Corrada Bravo^{1,2}, Rafael Álvarez Berríos² and T. Mitchell Aide^{2,3}

¹ Department of Computer Science, University of Puerto Rico—Río Piedras, San Juan, Puerto Rico

² Sieve Analytics, Inc., San Juan, Puerto Rico

³ Department of Biology, University of Puerto Rico—Río Piedras, San Juan, Puerto Rico

ABSTRACT

We developed a web-based cloud-hosted system that allow users to archive, listen, visualize, and annotate recordings. The system also provides tools to convert these annotations into datasets that can be used to train a computer to detect the presence or absence of a species. The algorithm used by the system was selected after comparing the accuracy and efficiency of three variants of a template-based detection. The algorithm computes a similarity vector by comparing a template of a species call with time increments across the spectrogram. Statistical features are extracted from this vector and used as input for a Random Forest classifier that predicts presence or absence of the species in the recording. The fastest algorithm variant had the highest average accuracy and specificity; therefore, it was implemented in the ARBIMON web-based system.

Subjects Bioinformatics, Computational Biology, Data Mining and Machine Learning

Keywords Acoustic monitoring, Machine learning, Animal vocalizations, Recording visualization, Recording annotation, Generic species algorithm, Web-based cloud-hosted system, Random Forest classifier, Species prediction, Species-specific audio detection

INTRODUCTION

Monitoring fauna is an important task for ecologists, natural resource managers, and conservationists. Historically, most data were collected manually by scientists that went to the field and annotated their observations (*Terborgh et al., 1990*). This generally limited the spatial and temporal extend of the data. Furthermore, given that the data were based on an individual's observations, the information was difficult to verify, reducing its utility for understanding long-term ecological processes (*Acevedo & Villanueva-Rivera, 2006*).

To understand the impacts of climate change and deforestation on the fauna, the scientific community needs long-term, wide-spread and frequent data (*Walther et al., 2002*). Passive acoustic monitoring (PAM) can contribute to this need because it facilitates the collection of large amounts of data from many sites simultaneously, and with virtually no impact to the fauna and environment (*Brandes, 2008; Lammers et al., 2008; Tricas & Boyle, 2009; Celis-Murillo, Deppe & Ward, 2012*). In general, PAM systems include a microphone or a hydrophone connected to a self powered system and enough memory to store various weeks or months of recordings, but there are also permanent systems that

Submitted 21 December 2016

Accepted 5 April 2017

Published 24 April 2017

Corresponding author

Carlos J. Corrada Bravo,
carlos.corrada2@upr.edu

Academic editor

Elena Marchiori

Additional Information and
Declarations can be found on
page 21

DOI 10.7717/peerj-cs.113

© Copyright

2017 Corrada Bravo et al.

Distributed under

Creative Commons CC-BY 4.0

OPEN ACCESS

use solar panels and an Internet connection to upload recordings in real time to a cloud based analytical platform ([Aide et al., 2013](#)).

Passive recorders can easily create a very large data set (e.g., 100,000s of recordings) that is overwhelming to manage and analyze. Although researchers often collect recordings twenty-four hours a day for weeks or months ([Acevedo & Villanueva-Rivera, 2006](#); [Brandes, 2008](#); [Lammers et al., 2008](#); [Sueur et al., 2008](#); [Marques et al., 2013](#); [Blumstein et al., 2011](#)), in practice, most studies have only analyzed a small percentage of the total number of recordings.

Web-based applications have been developed to facilitate data management of these increasingly large datasets ([Aide et al., 2013](#); [Villanueva-Rivera & Pijanowski, 2012](#)), but the biggest challenge is to develop efficient and accurate algorithms for detecting the presence or absence of a species in many recordings. Algorithms for species identification have been developed using spectrogram matched filtering ([Clark, Marler & Beeman, 1987](#); [Chabot, 1988](#)), statistical feature extraction ([Taylor, 1995](#); [Grigg et al., 1996](#)), k-Nearest neighbor algorithm ([Han, Muniandy & Dayou, 2011](#); [Gunasekaran & Revathy, 2010](#)), Support Vector Machine ([Fagerlund, 2007](#); [Acevedo et al., 2009](#)), tree-based classifiers ([Adams, Law & Gibson, 2010](#); [Henderson, Hildebrand & Smith, 2011](#)) and template based detection ([Anderson, Dave & Margoliash, 1996](#); [Mellinger & Clark, 2000](#)), but most of these algorithms are built for a specific species and there was no infrastructure provided for the user to create models for other species.

In this study, we developed a method that detects the presence or absence of a species' specific call type in recordings with a response time that allows researchers to create, run, tune and re-run models in real time as well as detect hundreds of thousands of recordings in a reasonable time. The main objective of the study was to compare the performance (e.g., efficiency and accuracy) of three variants of a template-based detection algorithm and incorporate the best into the ARBIMON II bioacoustics platform. The first variant is the Structural Similarity Index described in [Wang et al. \(2004\)](#), a widely use method to find how similar two images are (in our case the template with the tested recording). The second method filters the recordings with the dynamic thresholding method described in [Van der Walt et al. \(2014\)](#) and then use the Frobenius norm to find similarities with the template. The final method uses the Structural Similarity Index, but it is only applied to regions with high match probability determined by the OpenCV's matchTemplate procedure ([Bradski, 2000](#)).

MATERIALS AND METHODS

Passive acoustic data acquisition

We gathered recordings from five locations: four in Puerto Rico and one in Peru. Some of the recordings were acquired using the Automated Remote Biodiversity Monitoring Network (ARBIMON) data acquisition system described in [Aide et al. \(2013\)](#), while others were acquired using the newest version of ARBIMON permanent recording station, which uses an Android cell phone and transmits the recorded data through a cellular network. All recordings have a sampling rate of 44.1 kHz, a sampling depth of 16-bit and an approximate duration of 60 s (± 5 s).



Figure 1 Recording locations in Puerto Rico. Map data: Google, Image—Landsat/Copernicus and Data—SIO, NOAA, US Navy, NGA and GEBCO.

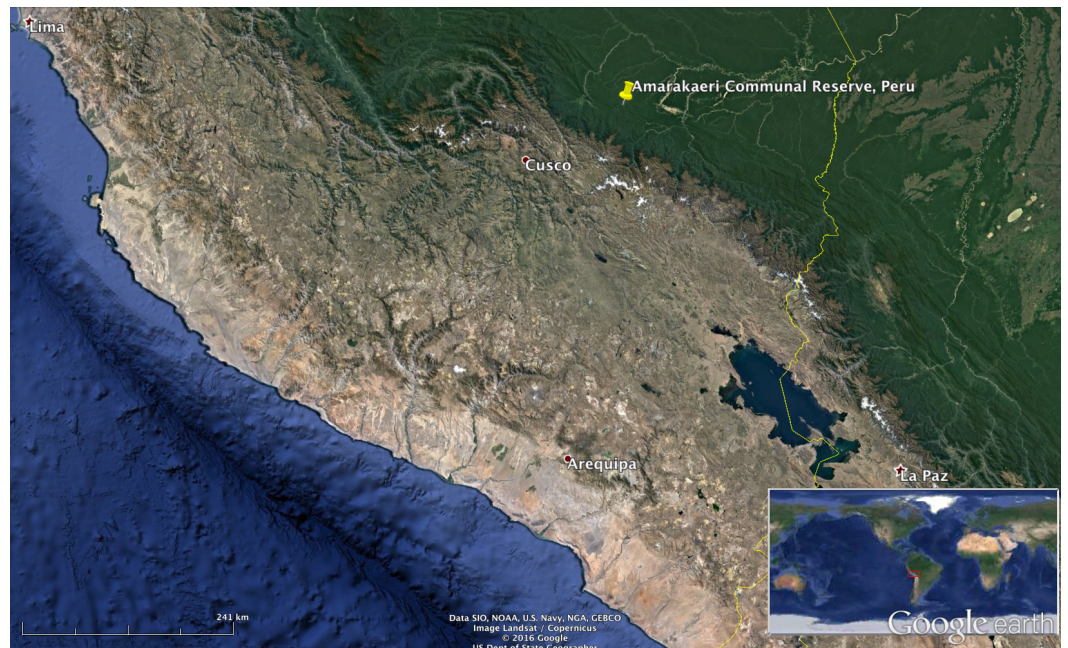


Figure 2 Recording location in Peru. Map data: Google, US Dept. of State Geographer, Image—Landsat/Copernicus and Data—SIO, NOAA, US Navy, NGA and GEBCO.

The locations in Puerto Rico were the Sabana Seca permanent station in Toa Baja, the Casa la Selva station in Carite Mountains (Patillas), El Yunque National Forest in Rio Grande and Mona Island (see Fig. 1). The location in Peru was the Amarakaeri Communal Reserve in the Madre de Dios Region (see Fig. 2). In all the locations, the recorders were programmed to record one minute of audio every 10 min. The complete dataset has more than 100,000 1-minute recordings. We randomly chose 362 recordings from Puerto Rico and 547 recordings from Peru for comparing the three algorithm variants.

We used the ARBIMON II web application interface to annotate the presence or absence of 21 species in all the recordings. Regions in the recording where a species emits a sound were also marked using the web interface. Each region of interest (ROI) is a rectangle

Table 1 Species, class, location and count of recordings with validated data.

Species	Group	Total	Presence	Absence	Location
<i>Eleutherodactylus cooki</i>	Amphibian	38	19	19	Carite
<i>Eleutherodactylus brittoni</i>	Amphibian	38	17	21	Sabana Seca
<i>Eleutherodactylus cochranae</i>	Amphibian	54	30	24	Sabana Seca
<i>Eleutherodactylus coqui</i>	Amphibian	53	41	12	Sabana Seca
<i>Eleutherodactylus juanariveroi</i>	Amphibian	35	14	21	Sabana Seca
<i>Unknown Insect</i>	Insect	48	22	26	Sabana Seca
<i>Epinephelus guttatus</i>	Fish	152	76	76	Mona Island
<i>Megascops nudipes</i>	Bird	100	50	50	El Yunque
<i>Microcerculus marginatus</i>	Bird	80	40	40	Peru
<i>Basileuterus chrysogaster</i>	Bird	60	30	30	Peru
<i>Myrmoborus leucophrys</i>	Bird	160	80	80	Peru
<i>Basileuterus bivittatus</i>	Bird	100	50	50	Peru
<i>Liosceles thoracicus</i>	Bird	76	38	38	Peru
<i>Chlorothraupis carmioli</i>	Bird	112	56	56	Peru
<i>Megascops guatemalae</i>	Bird	28	8	20	Peru
<i>Saltator grossus</i>	Bird	68	34	34	Peru
<i>Myrmeciza hemimelaena</i>	Bird	180	90	90	Peru
<i>Thamnophilus schistaceus</i>	Bird	60	30	30	Peru
<i>Hypocnemis subflava</i>	Bird	140	70	70	Peru
<i>Pernostola lophotes</i>	Bird	100	50	50	Peru
<i>Formicarius analis</i>	Bird	80	40	40	Peru

delimited by starting time, ending time, lowest frequency and highest frequency along with a species and sound type. The species included in the analysis are listed in Table 1, along with the number of total recordings and the number of recordings where the species is present or absent.

Algorithm

The algorithm recognition process is divided into three phases: (1) Template Computation, (2) Model Training and (3) Detection (see Fig. 3). In Template computation, all ROIs submitted by the user in the training set are aggregated into a template. In Model Training the template is used to compute recognition functions from validated audio recordings and features from the resulting vector V are computed. These features are used to train a random forest model. In the Detection phase the template is used to compute the features, but this time the features are fed to the trained random forest model to compute a prediction of presence or absence.

In the following sections the Template Computation process will be explained, then the process of using the Template to extract features from a recording is presented and finally, the procedures to use the features to train the model and to detect recordings are discussed.

Template computation

The template refers to the combination of all ROIs in the training data. To create a template, we first start with the examples of the specific call of interest (i.e., ROIs) that were annotated

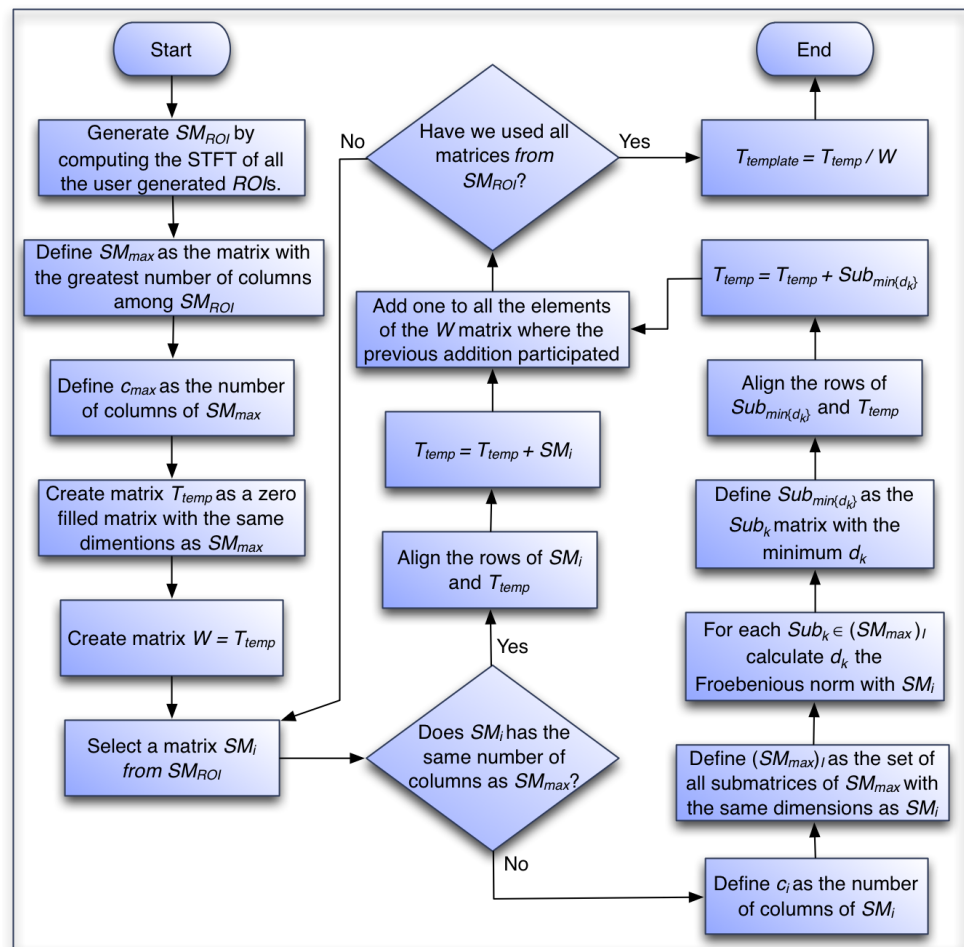


Figure 4 Flowchart of the algorithm to generate the template of each species.

Thus, if we were to perform an element-wise matrix sum between two ROI matrices with potentially different number of rows, we should only sum rows that represent the same frequency.

To take into account the difference in the number of columns of the ROI matrices, we use the Frobenius norm to optimize the alignment of the smaller ROI matrices and perform element-wise sums between rows that represent the same frequency. We present that algorithm in the following section and a flow chart of the process in Fig. 4.

Template computation algorithm:

1. Generate the set of SM_{ROI} matrices by computing the short time Fourier Transform of all the user generated ROIs.
2. Create matrix SM_{max} , a duplicate of the first created matrix among the matrices with the largest number of columns.
3. Set c_{max} as the number of columns in SM_{max} .
4. Create matrix T_{temp} , with the same dimensions as SM_{max} and all entries equal to 0. This matrix will contain the element-wise addition of all the extracted SM_{ROI} matrices.

5. Create matrix W with the same dimensions of SM_{max} and all entries equal to 0. This matrix will hold the count on the number of SM_{ROI} matrices that participate in the calculation of each element of T_{temp} .
6. For each one of the SM_i ROI matrices in SM_{ROI} :
 - (a) If SM_i has the same number of columns as T_{temp} :
 - i. Align the rows of SM_i and T_{temp} so they represent equivalent frequencies and perform an element-wise addition of the matrices and put the result in T_{temp} .
 - ii. Add one to all the elements of the W matrix where the previous addition participated.
 - (b) If the number of columns differs between SM_i and T_{temp} , then find the optimal alignment with SM_{max} as follows:
 - i. Set c_i as the number of columns in SM_i .
 - ii. Define $(SM_{max})_I$ as the set of all submatrices of SM_{max} with the same dimensions as SM_i . Note that the cardinality of $(SM_{max})_I$ is $c_{max} - c_i$.
 - iii. For each $Sub_k \in (SM_{max})_I$:
 - A. Compute $d_k = NORM(Sub_k - SM_i)$ where $NORM$ is the Frobenius norm defined as:

$$NORM(A) = \sqrt{\sum_{(i,j)} |a_{i,j}^2|}$$
 where $a_{i,j}$ are the elements of matrix A .
 - iv. Define $Sub_{min\{d_k\}}$ as the Sub_k matrix with the minimum d_k . This is the optimal alignment of SM_i with SM_{max} .
 - v. Align the rows of $Sub_{min\{d_k\}}$ and T_{temp} so they represent equivalent frequencies, perform an element-wise addition of the matrices and put the result in T_{temp} .
 - vi. Add one to all the elements of the W matrix where the previous addition participated.
7. Define the matrix $T_{template}$ as the element-wise division between the T_{temp} matrix and the W matrix.

The resulting $T_{template}$ matrix summarizes the information available in the ROI matrices submitted by the user and it will be used to extract information from the audio recordings that are to be analyzed. In this article each species $T_{template}$ was created using five ROIs.

In Fig. 5A a training set for the *Eleutherodactylus coqui* is presented and in Fig. 5B the resulting template can be seen. This tool is very useful because the user can see immediately the effect of adding or subtracting a specific sample to the training set.

Model training

The goal of this phase is to train a random forest model. The input to train the random forest are a series of statistical features extracted from vectors V_i that are created by computing a recognition function (similarity measure) between the computed $T_{template}$ and submatrices of the spectrogram matrices of a series of recordings.

In the following section, we present the details of the algorithm that processes a recording to create the recognition function vector, and in Fig. 6 we present a flowchart of the process.

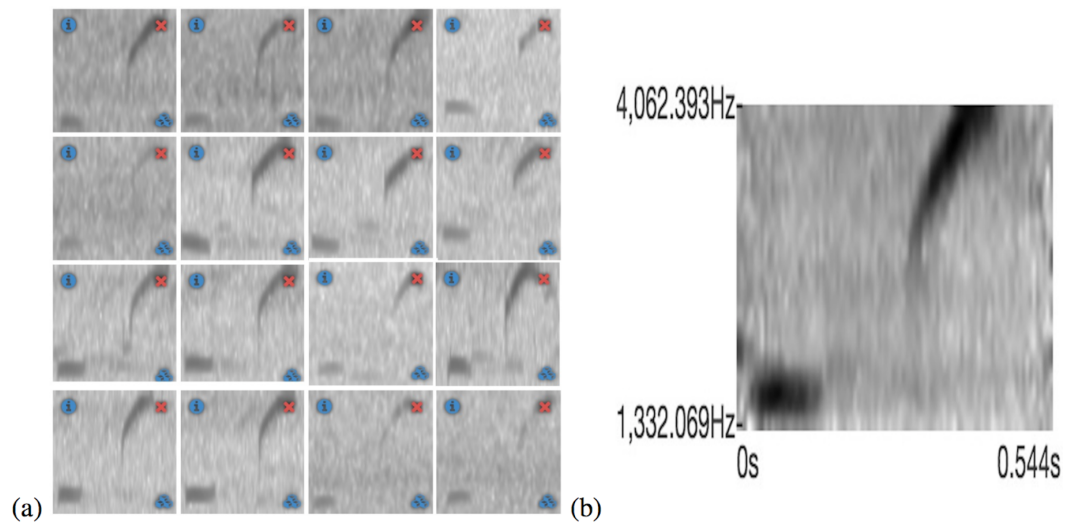


Figure 5 (A) A training set with 16 examples of the call of *E. coqui*. (B) The resulting template from the training set.

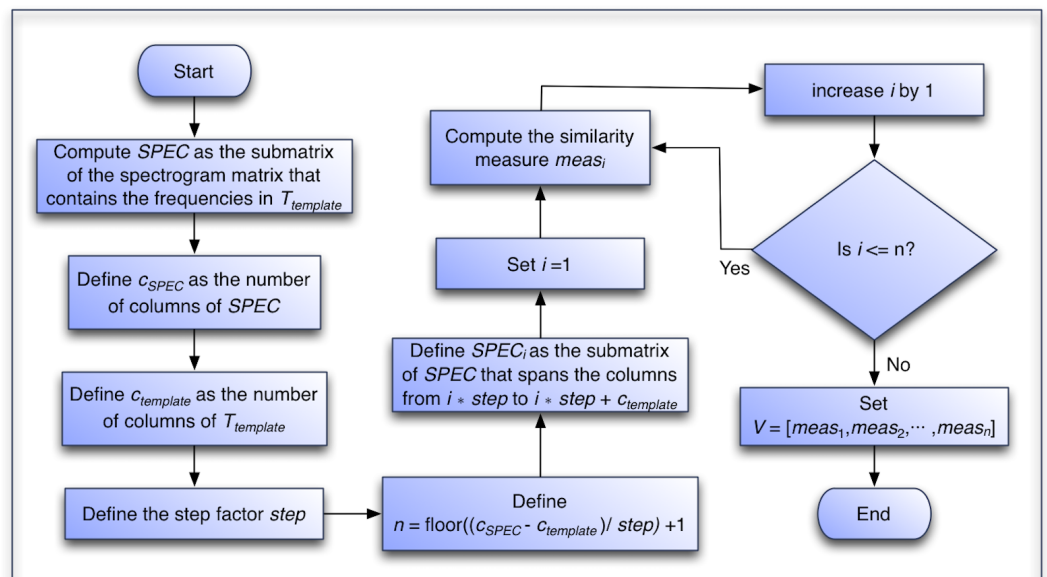


Figure 6 Flowchart of the algorithm to generate the similarity vector of each recording.

Algorithm to Create the Similarity Vector:

1. Compute matrix $SPEC$, the submatrix of the spectrogram matrix that contains the frequencies in $T_{template}$. Note that we are dealing with recordings that have the same sample rate as the recordings used to compute the $T_{template}$.
2. Define c_{SPEC} , the number of columns of $SPEC$.
3. Define $c_{template}$, the number of columns of $T_{template}$. Note that $c_{SPEC} \gg c_{template}$ since the $SPEC$ matrix have the same number of columns as the whole spectrogram and that the

¹Note that for recordings with a sample rate of 44,100 when we calculate the STFT with a window of size 512 and a 50% overlap, one step is equivalent to 5.8 ms; therefore, 16 steps is less than 100 ms. Although this procedure may miss the strongest match, the length of the calls are much longer than the step interval; therefore, there is a high probability of detecting the species-specific call.

$T_{template}$ matrix fits $c = c_{SPEC} - c_{template} + 1$ times inside the $SPEC$ matrix. There are c submatrices of $SPEC$ with the same dimensions as $T_{template}$.

4. Define $step$, the step factor by which $T_{template}$ will progressed over the $SPEC$ matrix.
5. Define $n = \left\lfloor \frac{c_{SPEC} - c_{template}}{step} \right\rfloor + 1$. Note that if $step = 1$ then $n = c$. In this work, however, this parameter was selected as $step = 16$ as a trade-off for speed.¹
6. Define $SPEC_i$ as the submatrix of $SPEC$ that spans the columns from $i \times step$ to $i \times step + c_{template}$.
7. Set $i = 1$.
8. While $i \leq n$
 - (a) Compute the similarity measure $meas_i$ for $SPEC_i$ (the definition of $meas_i$ for each of the three variants is provided in the following section).
 - (b) Increase i by 1. Note that this is equivalent to progressing $step$ columns in the $SPEC$ matrix.
9. Define the vector V as the vector containing the n similarity measures resulting from the previous steps. That is, $V = [meas_1, meas_2, meas_3, \dots, meas_n]$.

Recognition function

We used three variations of a pattern match procedure to define the similarity measure vector V . First, the Structural Similarity Index described in Wang et al. (2004) and implemented in Van der Walt et al. (2014) as `compare_ssim` with the default window size of seven unless the generated pattern is smaller. It will be referred in the rest of the manuscript as the SSIM variant. For the SSIM variant we define $meas_i$ as:

$$meas_i = SSI(T_{template}, SPEC_i),$$

where $SPEC_i$ is the submatrix of $SPEC$ that spans the columns from $i \times step$ to $i \times step + c_{template}$ and the same number of rows as $T_{template}$ and $V = [meas_1, meas_2, meas_3, \dots, meas_n]$ with

$$n = \left\lfloor \frac{c_{SPEC} - c_{template}}{step} \right\rfloor + 1.$$

Second, the dynamic thresholding method (`threshold_adaptive`) described in Van der Walt et al. (2014) with a block size of 127 and an arithmetic mean filter is used over both $T_{template}$ and $SPEC_i$ before multiplying them and applying the Frobenius norm and normalized by the norm of a matrix with same dimensions as $T_{template}$ and all elements equal to one. Therefore, $meas_i$ for the NORM variant is defined as:

$$meas_i = FN(DTM(T_{template}) .* DTM(SPEC_i)) / FN(U),$$

where again $SPEC_i$ is the submatrix of $SPEC$ that spans the columns from $i \times step$ to $i \times step + c_{template}$, FN is the Frobenius norm, DTM is the dynamic thresholding method, U is a matrix with same dimensions as $T_{template}$ with all elements equal to one and $.*$ performs an element-wise multiplication of the matrices. Again, $V = [meas_1, meas_2, meas_3, \dots, meas_n]$ with

$$n = \left\lfloor \frac{c_{SPEC} - c_{template}}{step} \right\rfloor + 1.$$

Finally, for the *CORR* variation we first apply the OpenCV's *matchTemplate* procedure (Bradski, 2000) with the Normalized Correlation Coefficient option to $SPEC_i$, the submatrix of $SPEC$ that spans the columns from $i \times step$ to $i \times step + c_{template}$. However, for this variant, $SPEC_i$ includes two additional rows above and below, thus it is slightly larger than the $T_{template}$. With these we can define:

$$meas_{j,i} = CORR(T_{template}, SPEC_{j,i})$$

where $SPEC_{j,i}$ is the submatrix of $SPEC_i$ that starts at row j (note that there are five such $SPEC_{j,i}$ matrices).

Now, we select five points at random from all the points above the 98.5 percentile of $meas_{j,i}$ and apply the Structural Similarity Index to the 5 strongly-matching regions. The size of these regions is eight thirds ($8/3$) of the length of $T_{template}$, $4/3$ before and $4/3$ after the strongly-matched point that was selected. Then, define *FilterSPEC* as the matrix that contains these 5 strongly-matching regions and $FilterSPEC_i$ as the submatrix of *FilterSPEC* that spans the columns from i to $i + c_{template}$ then, the similarity measure for this variant is defined as:

$$meas_i = SSI(T_{template}, FilterSPEC_i)$$

and the resulting vector $V = [meas_1, meas_2, meas_3, \dots, meas_n]$ but this time with

$$n = 5 \times (\lfloor 8/3 \times c_{template} \rfloor + 1).$$

It is important to note that no matter which variant is used to calculate the similarity measures, the result will always be a vector of measurements V . The idea is that the statistical properties of these computed recognition functions have enough information to distinguish between a recording that has the target species present and a recording that does not have the target species present. However, notice that since c_{SPEC} , the length of $SPEC$, is much larger than $c_{template}$ the length of the vector V for the *CORR* variant is much smaller than the other two.

Random forest model creation

After calculating V for many recordings we can train a random forest model. First, we need a set of validated recordings with the specific species vocalization present in some recordings and absent in others. Then for each recording we compute a vector V_i as described in the previous section and extract the statistical features presented in Table 2. These statistical features represent the dataset used to train the random forest model, which will be used to detect recordings for presence or absence of a species call event. These 12 features along with the species presence information are used as input to a random forest classifier with a 1,000 trees.

Recording detection

Now that we have a trained model to detect a recording, we have to compute the statistical features from the similarity vector V of the selected recording. This is performed in the same way as it was described in the previous section. These features are then used as the input dataset to the previously trained random forest classifier and a label indicating presence or absence of the species in the recording is given as output.

Table 2 The statistical features extracted from vector V .

Features	
1.	mean
2.	median
3.	minimum
4.	maximum
5.	standard deviation
6.	maximum–minimum
7.	skewness
8.	kurtosis
9.	hyper-skewness
10.	hyper-kurtosis
11.	Histogram
12.	Cumulative frequency histogram

The experiment

In order to decide which of the three variants should be incorporated into the ARBIMON web-based system, we performed the algorithm explained in the previous section with each of the similarity measures. We computed 10-fold validations on each of the variants to obtain measurements of the performance of the algorithm. In each validation, 90% of the data is used as training and 10% of the data is used as validation data. Each algorithm variant used the same 10-fold validation partition for each species. The measures calculated were the area under the receiver operating characteristic (ROC) curve (AUC), accuracy or correct detection rate (Ac), negative predictive value (Npv), precision or positive predictive value (Pr), sensitivity, recall or true positive rate (Se) and specificity or true negative rate (Sp). To calculate the AUC , the ROC curve is created by plotting the false positive rate (which can be calculated as $1 - \text{specificity}$) against the true positive rate (sensitivity), then, the AUC is created by calculating the area under that curve. Notice that the further the AUC is from 0.5 the better. The rest of the measures are defined as follows:

$$Ac = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}, \quad Npv = \frac{t_n}{t_n + f_n}, \quad Pr = \frac{t_p}{t_p + f_p},$$

$$Se = \frac{t_p}{t_p + f_n} \quad \text{and} \quad Sp = \frac{t_n}{t_n + f_p}$$

with t_p the number of true positives (number of times both the expert and the algorithm agree that the species is present), t_n the number of true negatives (number of times both the expert and the algorithm agree that the species is not present), f_p the number of false positives (number of times the algorithm states that the species is present while the expert states is absent) and f_n the number of false negatives (number of times the algorithm states that the species is not present while the expert states it is present). Note that accuracy is a weighted average of the sensitivity and the specificity.

Although we present and discuss all measures, we gave accuracy and the AUC more importance because they include information on the true positive and true negative rates.

Specifically, *AUC* is important when the number of positives is different than the number of negatives as is the case with some of the species.

The experiment was performed in a computer with an Intel i7 4790K 4 cores processor at 4.00 GHz with 32 GB of RAM and running Ubuntu Linux. The execution time needed to detect each recording was registered and the mean and standard deviation of the execution times were calculated for each variant of the algorithm. We also computed the quantity of pixels on all the $T_{template}$ matrices and correlated with the execution time of each of the variants.

A global one-way analysis of variance (ANOVA) was performed on the five calculated measures across all of the 10-fold validations to identify if there was a significant difference between the variants of the algorithm. Then, a post-hoc Tukey HSD comparison test was performed to identify which one of the variants was significantly different at the 95% confidence level. Additionally, an ANOVA was performed locally between the 10-fold validation of each species and on the mean execution time for each species across the algorithm variants to identify if there was any significant execution time difference at the 95% confidence level. Similarly, a post-hoc Tukey HSD comparison test was performed on the execution times.

RESULTS

The six measurements (area under the ROC curve—*AUC*, accuracy, negative predictive value, precision, sensitivity and specificity) computed to compared the model across the three variants varied greatly among the 21 species. The lowest scores were among bird species while most of the highest scores came from amphibian species. Table 3 presents a summary of the results of the measurements comparing the three variants of the algorithm (for a detail presentation, see Appendix 1). The NORM variant did not have the highest value for any of the measures summarized in Table 3, while the CORR variant had a greater number of species with 80% or greater for all the measures and an overall median accuracy of 81%. We considered these two facts fundamental for a general-purpose species detection system.

The local species ANOVA suggested that there are significant accuracy differences at the 95% significance level for six of the 21 species studied as well as four in terms of precision and three in terms of specificity (see <https://doi.org/10.6084/m9.figshare.4488149.v1>). The algorithm variant CORR had a higher mean and median *AUC* at 78% and 81% respectively, but the SSIM variant seems to be more stable with a standard deviation of 20%. In terms of accuracy, both the SSIM and CORR have higher mean accuracy than the NORM variant. Nevertheless, variant CORR had the highest median accuracy of 81%, which is slightly higher than the median accuracy of the SSIM variant at 76%. In addition, variant CORR had more species with an accuracy of 80% or greater.

In terms of median precision, the three variants had similar values, although in terms of mean precision variants SSIM and CORR have greater values than the NORM variant. Moreover, the median and mean precision of the SSIM variant were only 1% higher than the median and mean precision of the CORR variant. In terms of sensitivity, variants SSIM

Table 3 Summary of the measures of the three variants of the algorithm. Best values are in bold.

Summary of measures	SSIM	NORM	CORR
Number of species with an Area under the curve of 80% or greater	8	7	12
Number of species with statistically significant Area under the curve	0	0	0
Mean Area under the curve	0.76	0.71	0.78
Median Area under the curve	0.75	0.72	0.81
Standard Deviation of Area under the curve	0.20	0.21	0.21
Number of species with an Accuracy of 80% or greater	8	7	12
Number of species with statistically significant Accuracy	3	0	3
Mean Accuracy	0.77	0.73	0.77
Median Accuracy	0.76	0.75	0.81
Standard Deviation of Accuracy	0.12	0.14	0.14
Number of species with a Negative predictive value of 80% or greater	7	5	10
Number of species with statistically significant Negative predictive value	0	0	0
Mean Negative predictive value	0.73	0.71	0.74
Median Negative predictive value	0.71	0.75	0.79
Standard Deviation of Negative predictive value	0.08	0.12	0.13
Number of species with a Precision of 80% or greater	5	5	9
Number of species with statistically significant Precision	2	0	2
Mean Precision	0.73	0.68	0.72
Median Precision	0.75	0.73	0.74
Standard Deviation of Precision	0.12	0.13	0.16
Number of species with a Sensitivity of 80% or greater	8	6	11
Number of species with statistically significant Sensitivity	0	0	0
Mean Sensitivity	0.77	0.70	0.74
Median Sensitivity	0.79	0.73	0.80
Standard Deviation of Sensitivity	0.12	0.16	0.17
Number of species with a Specificity of 80% or greater	4	6	7
Number of species with statistically significant Specificity	3	0	0
Mean Specificity	0.69	0.68	0.72
Median Specificity	0.67	0.70	0.75
Standard Deviation of Specificity	0.13	0.15	0.16
Ratio of False positive to True positive	0.37	0.47	0.39
Ratio of False negative to True positive	0.45	0.47	0.39
Ratio of False positive to True negative	0.3	0.43	0.35
Ratio of False negative to True negative	0.37	0.43	0.35

and CORR had greater values than the NORM variant. It is only in terms of specificity that the CORR variant has greater values than all other variants. [Figures 7](#) and [8](#) present a summary of these results with whisker graphs.

In terms of execution times, an ANOVA analysis on the mean execution time suggests a difference between the variants ($F = 9.9341e + 30$, $df = 3$, $p < 2.2e-16$). The CORR variant had the lowest mean execution time at 0.255 s followed closely by the NORM variant with 0.271 s, while the SSIM variant had the slowest mean execution time of 2.269 s ([Fig. 9](#)). The Tukey HSD test suggests that there was no statistical significant difference between the

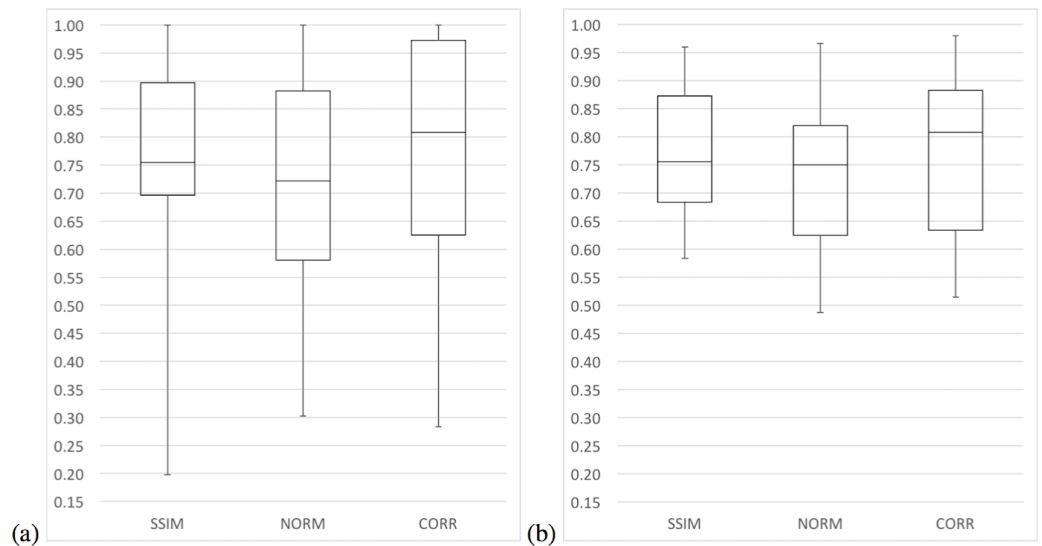


Figure 7 Whisker boxes of the 10-fold validations for the three variants of the presented algorithm for: (A) Area under the ROC curve and (B) Accuracy.

Table 4 Summary of the execution times of the three variants of the algorithm. Best values are in bold. PPMCC is the Pearson product-moment correlation coefficient.

Summary of execution times	SSIM	NORM	CORR
Mean Execution Time	2.27	0.27	0.26
Standard Deviation of Execution Time	3.04	0.06	0.07
PPMCC between Execution Time and size of template	0.96	0.33	0.11

mean execution times of the NORM and CORR variants ($p = 0.999$). However, there was a statistical significant difference at the 95% confidence level between the mean execution times of all other pairs of variants, specifically variants SSIM and CORR ($p < .2e-16$).

Moreover, the mean execution time of the SSIM variant increased as the number of pixels in the $T_{template}$ matrix increases (Fig. 9B). There was no statistically significant relationship between the $T_{template}$ pixel size and the execution time for the other two variants (Table 4).

In summary, variants SSIM and CORR outperform the NORM variant in most of the statistical measures computed having statistically significant high accuracy for three species each. In terms of execution time, the CORR variant was faster than the SSIM variant (Table 3), and the mean execution time of CORR variant did not increase with increasing $T_{template}$ size (Table 4).

DISCUSSION

The algorithm used by the ARBIMON system was selected by comparing three variants of a template-based method for the detection of presence or absence of a species vocalization in recordings. The most important features for selecting the algorithm were that it works well for many types of species calls and that it can process hundreds of thousands of recordings in a reasonable amount of time. The CORR algorithm was selected because of its speed

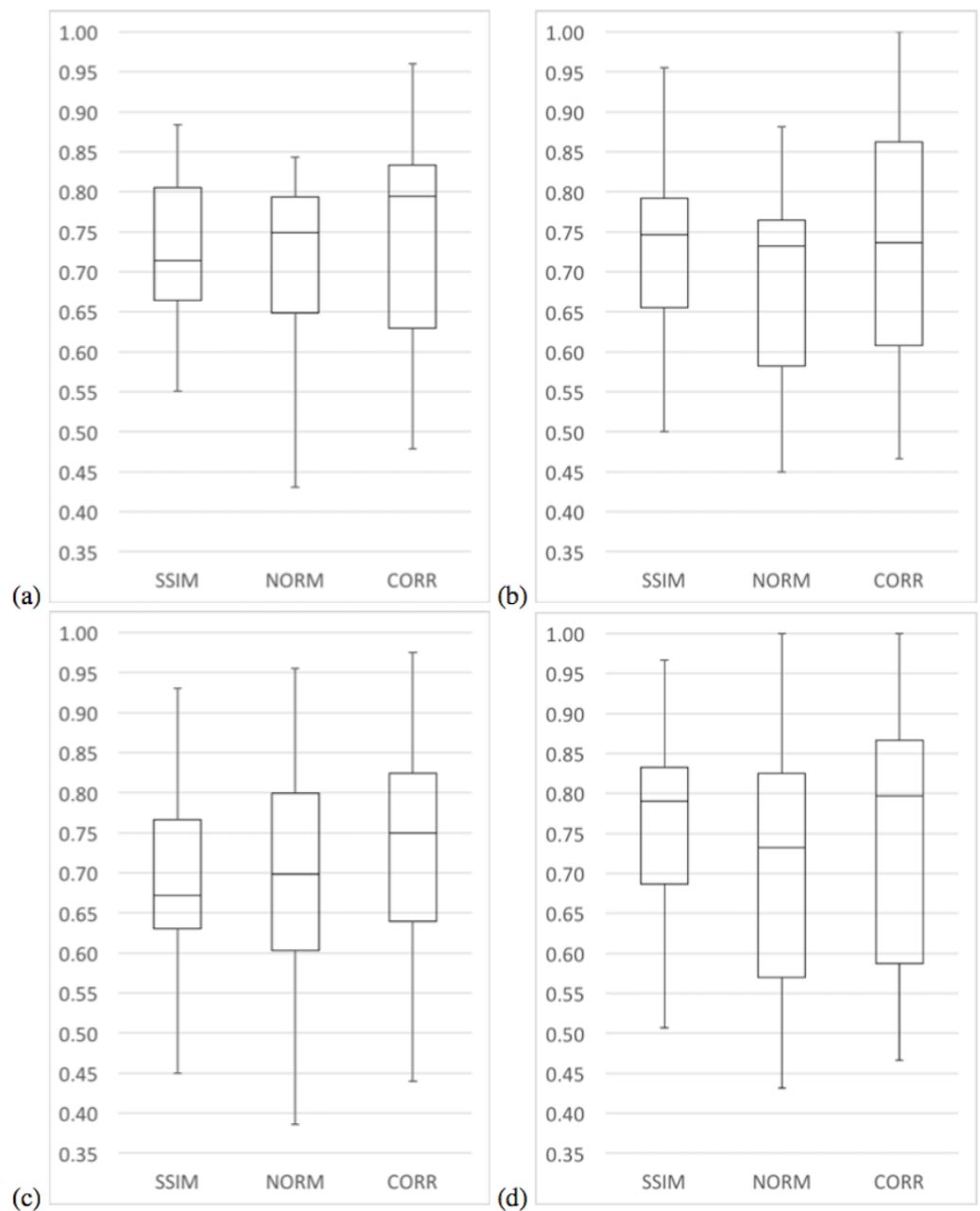


Figure 8 Whisker boxes of the 10-fold validations for the three variants of the presented algorithm for: (A) Negative predictive value, (B) Precision, (C) Sensitivity and (D) Specificity.

and its comparable performance in terms of detection efficiency with the SSIM variant. It achieved AUC and accuracy of 0.80 or better in 12 of the 21 species and sensitivity of 0.80 or more in 11 of the 21 species and the average execution time of 0.26 s per minute of recording means that it can process around 14,000 minutes of recordings per hour.

The difference in execution time between the SSIM variant and the other two was due to a memory management issue in the SSIM algorithm. An analysis revealed that all the

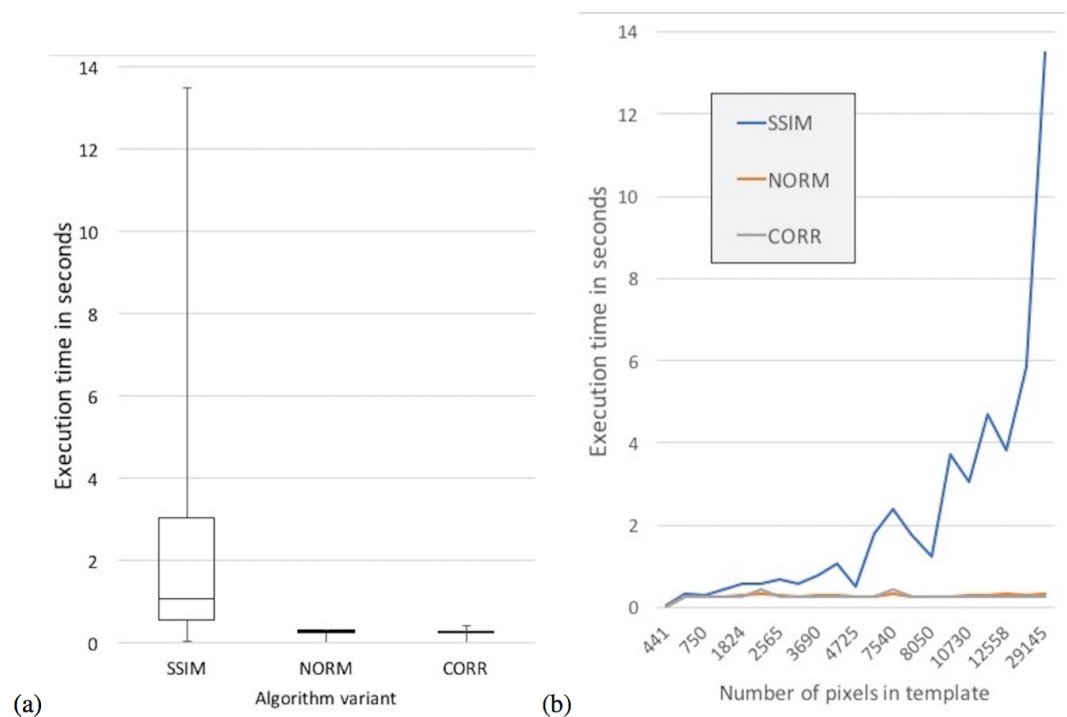


Figure 9 (A) Whisker boxes of the execution times of the three algorithms. (B) Execution times as a function of the size of the template in number of pixels.

algorithms have time complexity of

$$O((c_{SPEC} - c_{template}) \times c_{template} \times r_{template})$$

where c_{SPEC} and $c_{template}$ are the number of columns in $SPEC$ and $T_{template}$ respectively and $r_{template}$ is the number of rows in $T_{template}$. The only explanation we can give is that the SSIM function uses an uniformly distributed filter (`uniform_filter`) that has a limit on the size of the memory buffer (4,000 64-bit doubles divided by the number of elements in the dimension been process). Therefore, as the size of $T_{template}$ increases the number of calls to allocate the buffer, free and allocate again can become a burden since it has a smaller locality of reference even when the machine has enough memory and cache to handle the process. Further investigation is required to confirm this.

An interesting comparison is the method described in the work by [Fodor \(2013\)](#) and adapted and tested by [Lasseck \(2013\)](#). This method was design for the Neural Information Processing Scaled for Bioacoustics (NIPS4B) competition, and although the results are very good they do not report on time of execution. As we have mentioned, it is very important to us to have a method that provides good response times and the execution time of Lasseck's method seems to be greater than ours given the extensive pre-processing that the method performs.

Table 5 Summary of the usage of the ARBIMON2 system and its model creation feature.

Number of users in the system	453
Number of recordings in the system	1,749,551
Number of models created by users	659
Total number of detected recordings	3,780,552
Number of distinct detected recordings	723,054
Average times a recording is detected	5.22
Standard deviation of the number of times a recording is detected	7.78
Maximum number of times a recordings has been detected	58

CONCLUSIONS AND FUTURE WORK

Now that passive autonomous acoustic recorders are readily available the amount of data is growing exponentially. For example, one permanent station recording one minute of every 10 minutes every day of the year generates 52,560 one minute recordings. If this is multiplied by the need to monitor thousands of locations across the planet, one can understand the magnitude of the task at hand.

We have shown how the algorithm used in the ARBIMON II web-based cloud-hosted system was selected. We compared the performance in terms of the ability to detect and the efficiency in terms of time execution of three variants of a template-based detection algorithm. The result was a method that uses the power of a widely use method to determine the similarity between two images (Structural Similarity Index ([Wang et al., 2004](#))), but to accelerate the detection process, the analysis was only done in regions where there was a strong-match determined by the OpenCV's matchTemplate procedure ([Bradski, 2000](#)). The results show that this method performed better both in terms of ability to detect as well as in terms of execution time.

A fast and accurate general-purpose algorithm for detecting presence or absence of a species complements the other tools of the ARBIMON system, such as options for creating playlists based on many different parameters including user-created tags (see [Table 5](#)). For example, the system currently has 1,749,551 1-minute recordings uploaded by 453 users and 659 species specific models have been created and run over 3,780,552 min of recordings of which 723,054 are distinct recordings.

While this research was a prove of concept, we provide the tools and encourage users to increase the size of the training data set as this should improve the performance of the algorithm. In addition, we will pursue other approaches, such as multi-label learning ([Xie et al., 2016](#); [Zhang et al., 2016](#); [Briggs et al., 2012](#)).

As a society, it is fundamental that we study the effects of climate change and deforestation on the fauna and we have to do it with the best possible tools. We are collecting a lot of data, but until recently there was not an intuitive and user-friendly system that allowed scientists to manage and analyze large number of recordings. Here we presented a web-based cloud-hosted system that provides a simple way to manage large quantities of recordings with a general-purpose method to detect their presence in recordings.

ACKNOWLEDGEMENTS

The authors want to thank Marconi Campos-Cerqueira for his helpful comments on the manuscript.

APPENDIX 1

Table 6 provide a detail presentation of the performance of each variant of the algorithm: The area under the ROC curve, mean accuracy, mean precision, mean sensitivity and mean specificity values for each species, of the 10-fold validations for the three variants of the presented algorithm (SSIM, NORM and CORR). The mean, median and standard deviation values across all species are presented at the bottom of the table.

APPENDIX 2

The templates created by the training sets of each species are presented. We classified them by the algorithm that presented a better accuracy for that species. Figure 10 presents the templates of the species where the SSIM variant presented better accuracy, Fig. 11 presents those where the NORM variant presented better accuracy, and Fig. 12 presents the species where the CORR variant presented better accuracy.

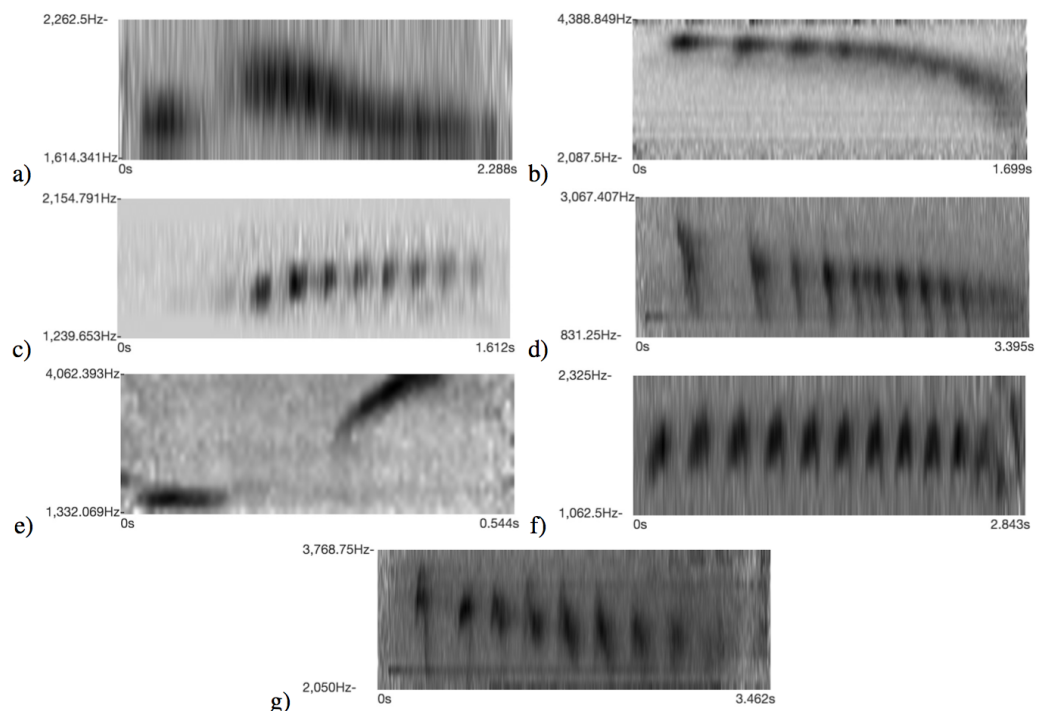


Figure 10 Sample of species that the SSIM variant presented better accuracy. (A) *F. analis*, (B) *M. hemimelaena*, (C) *E. cooki*, (D) *P. lophotes*, (E) *E. coqui*, (F) *T. schistaceus* and (G) *H. subflava*. Species (A–C) are statistically significant.

Table 6 Area Under the ROC Curve (AUC), Accuracy (Ac), negative predictive value (Npv), precision (Pr), sensitivity (Se) and specificity (Sp) of the 21 species and three variants of the algorithm. Best values are shaded and the cases where the ANOVA suggested a significant difference between the algorithm variants at the 95% confidence level are in bold.

Species	SSIM						NORM						CORR					
	AUC	Ac	Npv	Pr	Se	Sp	AUC	Ac	Npv	Pr	Se	Sp	AUC	Ac	Npv	Pr	Se	Sp
<i>E. brittoni</i>	1.00	0.92	0.81	0.77	0.72	0.95	0.42	0.89	0.83	0.80	0.77	0.92	1.00	0.98	0.84	0.80	0.77	1.00
<i>E. cochranæ</i>	1.00	0.87	0.84	0.94	0.88	0.85	0.88	0.72	0.70	0.81	0.77	0.68	1.00	0.98	0.96	1.00	0.97	1.00
<i>M. guatemalæ</i>	0.50	0.93	0.81	0.50	0.45	0.97	1.00	0.97	0.82	0.50	0.45	1.00	0.50	0.90	0.80	0.47	0.45	0.87
<i>E. cooki</i>	1.00	0.96	0.85	0.77	0.77	0.97	0.72	0.82	0.78	0.73	0.67	0.87	0.88	0.89	0.82	0.72	0.73	0.92
<i>Unknown Insect</i>	1.00	0.90	0.79	0.84	0.75	0.82	1.00	0.92	0.84	0.83	0.82	0.83	1.00	0.90	0.79	0.84	0.75	0.82
<i>E. coqui</i>	0.88	0.90	0.75	0.96	0.93	0.70	0.92	0.86	0.75	0.88	0.96	0.47	1.00	0.88	0.85	0.89	0.98	0.47
<i>M. leucophrys</i>	0.98	0.87	0.88	0.87	0.89	0.87	0.77	0.76	0.79	0.74	0.81	0.72	0.98	0.88	0.87	0.89	0.87	0.90
<i>E. juanariveroi</i>	0.20	0.78	0.69	0.60	0.48	0.79	0.50	0.88	0.70	0.55	0.48	0.83	0.63	0.81	0.69	0.47	0.45	0.80
<i>M. nudipes</i>	0.90	0.74	0.76	0.75	0.77	0.74	0.84	0.81	0.84	0.80	0.85	0.79	0.90	0.85	0.83	0.88	0.82	0.86
<i>B. bivittatus</i>	0.77	0.59	0.65	0.65	0.64	0.65	0.90	0.74	0.78	0.73	0.80	0.73	0.95	0.85	0.84	0.88	0.83	0.87
<i>C. carmioli</i>	0.78	0.77	0.75	0.83	0.73	0.83	0.78	0.73	0.75	0.73	0.76	0.72	0.83	0.81	0.80	0.86	0.80	0.84
<i>L. thoracicus</i>	0.70	0.73	0.71	0.76	0.67	0.79	0.90	0.76	0.80	0.73	0.80	0.77	0.97	0.81	0.83	0.82	0.84	0.80
<i>F. analis</i>	0.82	0.81	0.81	0.79	0.82	0.79	0.68	0.63	0.65	0.63	0.69	0.57	0.57	0.58	0.59	0.58	0.62	0.55
<i>E. guttatus</i>	0.74	0.69	0.70	0.69	0.70	0.69	0.72	0.75	0.76	0.77	0.77	0.75	0.78	0.77	0.77	0.78	0.77	0.77
<i>M. hemimelaena</i>	0.75	0.76	0.71	0.77	0.67	0.82	0.61	0.59	0.59	0.58	0.60	0.57	0.61	0.63	0.62	0.63	0.65	0.59
<i>B. chrysogaster</i>	0.56	0.68	0.66	0.67	0.62	0.74	0.69	0.75	0.70	0.72	0.65	0.83	0.80	0.73	0.69	0.64	0.66	0.78
<i>S. grossus</i>	0.70	0.66	0.66	0.68	0.66	0.67	0.78	0.74	0.72	0.75	0.70	0.76	0.81	0.71	0.73	0.74	0.78	0.62
<i>P. lophotes</i>	0.73	0.71	0.68	0.73	0.63	0.78	0.58	0.58	0.60	0.59	0.62	0.57	0.65	0.61	0.63	0.62	0.64	0.61
<i>H. subflava</i>	0.74	0.64	0.64	0.64	0.66	0.61	0.51	0.51	0.51	0.52	0.53	0.49	0.51	0.51	0.52	0.51	0.56	0.48
<i>M. marginatus</i>	0.58	0.59	0.55	0.60	0.59	0.51	0.32	0.49	0.43	0.47	0.39	0.47	0.69	0.61	0.62	0.61	0.66	0.56
<i>T. schistaceus</i>	0.62	0.58	0.58	0.61	0.51	0.67	0.30	0.50	0.46	0.45	0.49	0.43	0.28	0.52	0.48	0.49	0.44	0.52
Mean values	0.76	0.77	0.73	0.73	0.69	0.77	0.71	0.73	0.71	0.68	0.68	0.70	0.78	0.77	0.74	0.72	0.72	0.74
Median values	0.75	0.76	0.71	0.75	0.67	0.79	0.72	0.75	0.75	0.73	0.70	0.73	0.81	0.81	0.79	0.74	0.75	0.80
Standard Dev.	0.20	0.12	0.09	0.12	0.13	0.12	0.21	0.14	0.12	0.13	0.15	0.16	0.21	0.14	0.13	0.16	0.16	0.17

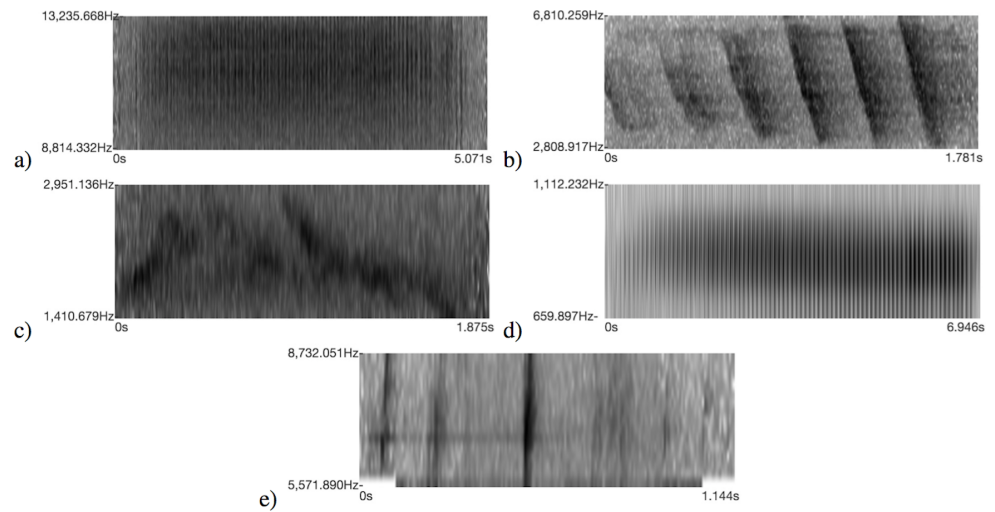


Figure 11 Sample of species that the NORM variant presented better accuracy. (A) *Unknown Insect*, (B) *B. chrysogaster*, (C) *S. grossus*, (D) *M. guatemalae* and (E) *E. juanariveroi*. Neither is statistically significant.

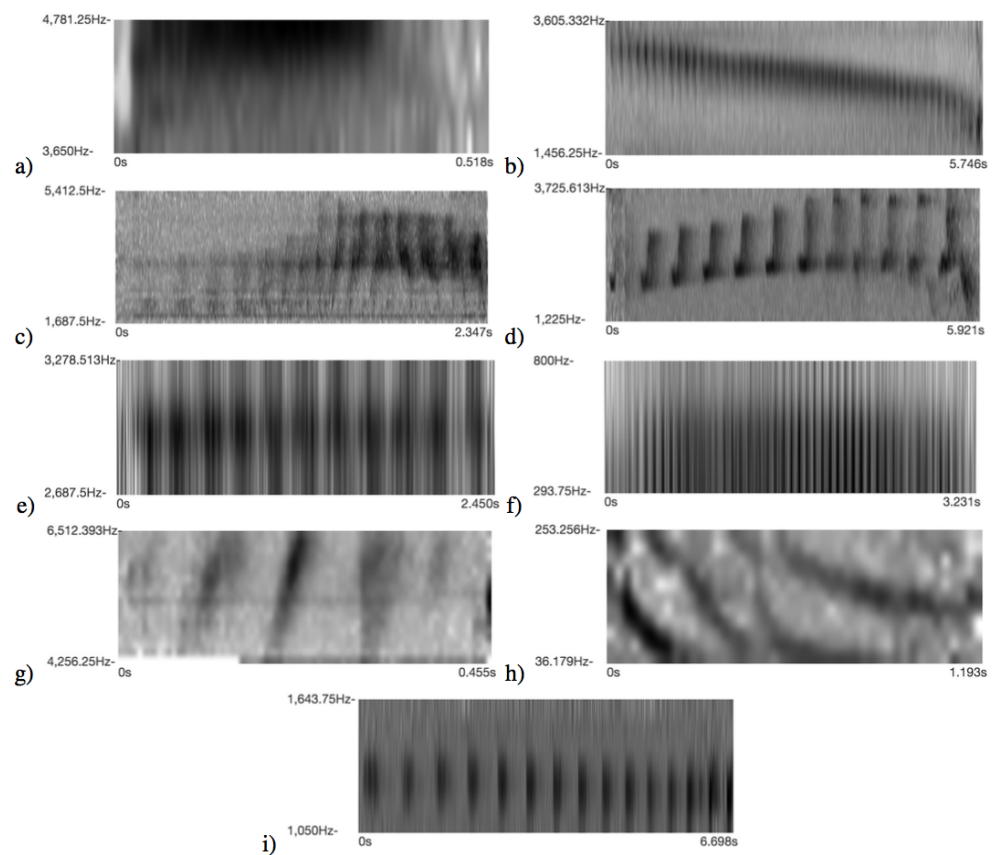


Figure 12 Sample of species that the CORR variant presented better accuracy. (A) *E. cochraeae*, (B) *M. leucophrys*, (C) *B. bivittatus*, (D) *C. carmioli*, (E) *M. marginatus*, (F) *M. nudipes*, (G) *E. brittoni*, (H) *E. guttatus* and (I) *L. thoracicus*. Species (A–C) are statistically significant.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

The authors received no funding for this work.

Competing Interests

Carlos J. Corrada Bravo and T. Mitchell Aide are the founders of Sieve Analytics, Inc. Rafael Álvarez Berríos was an employee of Sieve Analytics, Inc.

Author Contributions

- Carlos J. Corrada Bravo conceived and designed the experiments, analyzed the data, contributed reagents/materials/analysis tools, wrote the paper, prepared figures and/or tables, reviewed drafts of the paper.
- Rafael Álvarez Berríos conceived and designed the experiments, performed the experiments, analyzed the data, contributed reagents/materials/analysis tools, wrote the paper, prepared figures and/or tables, performed the computation work.
- T. Mitchell Aide conceived and designed the experiments, analyzed the data, wrote the paper, reviewed drafts of the paper.

Data Availability

The following information was supplied regarding data availability:

Corrada Bravo, Carlos J (2016): Code, recordings, validations and Regions of interest. figshare.

<https://doi.org/10.6084/m9.figshare.4488149.v1>.

REFERENCES

- Acevedo MA, Corrada-Bravo CJ, Corrada-Bravo H, Villanueva-Rivera LJ, Aide TM. 2009.** Automated classification of bird and amphibian calls using machine learning: a comparison of methods. *Ecological Informatics* 4:206–214 DOI 10.1016/j.ecoinf.2009.06.005.
- Acevedo M, Villanueva-Rivera L. 2006.** Using automated digital recording systems as effective tools for the monitoring of birds and amphibians. *Wildlife Society Bulletin* 34:211–214 DOI 10.2193/0091-7648(2006)34[211:UADRSA]2.0.CO;2.
- Adams MD, Law BS, Gibson MS. 2010.** Reliable automation of bat call identification for Eastern New South Wales, Australia, using classification trees and anascheme software. *Acta Chiropterologica* 12(1):231–245.
- Aide TM, Corrada-Bravo C, Campos-Cerqueira M, Milan C, Vega G, Alvarez R. 2013.** Real-time bioacoustics monitoring and automated species identification. *PeerJ* 1:e103 DOI 10.7717/peerj.103.
- Anderson SE, Dave AS, Margoliash D. 1996.** Template-based automatic recognition of birdsong syllables from continuous recordings. *The Journal of the Acoustical Society of America* 100:1209–1219 DOI 10.1121/1.415968.

- Blumstein DT, Mennill DJ, Clemens P, Girod L, Yao K, Patricelli G, Deppe JL, Krakauer AH, Clark C, Cortopassi KA, Hanser SF, McCowan B, Ali AM, Kirschel ANG. 2011. Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus. *Journal of Applied Ecology* 48(3):758–767 DOI 10.1111/j.1365-2664.2011.01993.x.
- Bradski G. 2000. The OpenCV Library. *Doctor Dobbs Journal* 25(11):120–126.
- Brandes ST. 2008. Automated sound recording and analysis techniques for bird surveys and conservation. *Bird Conservation International* 18:163–173 DOI 10.1017/S0959270908000415.
- Briggs F, Lakshminarayanan B, Neal L, Fern XZ, Raich R, Hadley SJK, Hadley AS, Betts MG. 2012. Acoustic classification of multiple simultaneous bird species: a multi-instance multi-label approach. *The Journal of the Acoustical Society of America* 131(6):4640–4650 DOI 10.1121/1.4707424.
- Celis-Murillo A, Deppe JL, Ward MP. 2012. Effectiveness and utility of acoustic recordings for surveying tropical birds. *Journal of Field Ornithology* 83(2):166–179 DOI 10.1111/j.1557-9263.2012.00366.x.
- Chabot D. 1988. A quantitative technique to compare and classify humpback whale (*Megaptera novaeangliae*) sounds. *Ethology* 77(2):89–102 DOI 10.1111/j.1439-0310.1988.tb00195.x.
- Clark CW, Marler P, Beeman K. 1987. Quantitative analysis of animal vocal phonology: an application to swamp sparrow song. *Ethology* 76(2):101–115 DOI 10.1111/j.1439-0310.1987.tb00676.x.
- Fagerlund S. 2007. Bird species recognition using support vector machines. *EURASIP Journal on Applied Signal Processing* 2007(1):1–8.
- Fodor G. 2013. The ninth annual MLSP competition: first place. In: *Machine learning for signal processing (MLSP), 2013 IEEE international workshop on IEEE*, 1–2.
- Grigg G, Taylor A, Mc Callum H, Watson G. 1996. Monitoring frog communities: an application of machine learning. In: *Proceedings of eighth innovative applications of artificial intelligence conference, Portland Oregon*, 1564–1569.
- Gunasekaran S, Revathy K. 2010. Content-based classification and retrieval of wild animal sounds using feature selection algorithm. In: *Second international conference on machine learning and computing..*
- Han NC, Muniandy SV, Dayou J. 2011. Acoustic classification of Australian anurans based on hybrid spectral-entropy approach. *Applied Acoustics* 72(9):639–645.
- Henderson E, Hildebrand JA, Smith MH. 2011. Classification of behavior using vocalizations of Pacific white-sided dolphins (*Lagenorhynchus obliquidens*). *The Journal of the Acoustical Society of America* 130(1):557–567.
- Lammers MO, Brainard RE, Au WWL, Mooney TA, Wong KB. 2008. An ecological acoustic recorder (EAR) for long-term monitoring of biological and anthropogenic sounds on coral reefs and other marine habitats. *The Journal of the Acoustical Society of America* 123:1720–1728 DOI 10.1121/1.2836780.

- Lasseck M. 2013.** Bird song classification in field recordings: winning solution for NIPS4B 2013 competition. In: *Proc. of int. symp. Neural Information Scaled for Bioacoustics, sabiod. org/nips4b, joint to NIPS, Nevada*, 176–181.
- Marques TA, Thomas L, Martin SW, Mellinger DK, Ward JA, Moretti DJ, Harris D, Tyack PL. 2013.** Estimating animal population density using passive acoustics. *Biological Reviews* **88**(2):287–309 DOI [10.1111/brv.12001](https://doi.org/10.1111/brv.12001).
- Mellinger DK, Clark CW. 2000.** Recognizing transient low-frequency whale sounds by spectrogram correlation. *The Journal of the Acoustical Society of America* **107**:3518–3529 DOI [10.1121/1.429434](https://doi.org/10.1121/1.429434).
- Sueur J, Pavoine S, Hamerlynck O, Duvail S. 2008.** Rapid acoustic survey for biodiversity appraisal. *PLOS ONE* **3**:e4065 DOI [10.1371/journal.pone.0004065](https://doi.org/10.1371/journal.pone.0004065).
- Taylor A. 1995.** Bird flight call discrimination using machine learning. *The Journal of the Acoustical Society of America* **97**(5):3370–3370.
- Terborgh J, Robinson SK, Parker III TA, Munn CA, Pierpont N. 1990.** Structure and organization of an Amazonian forest bird community. *Ecological Monographs* **60**:213–238 DOI [10.2307/1943045](https://doi.org/10.2307/1943045).
- Tricas TC, Boyle K. 2009.** Validated reef fish sound scans of passive acoustic monitors on Hawaiian coral reefs [Abstract 2589]. *The Journal of the Acoustical Society of America* **125**(4) DOI [10.1121/1.4783839](https://doi.org/10.1121/1.4783839).
- Van der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, Gouillart E, Yu T, The Scikit-Image Contributors. 2014.** scikit-image: image processing in Python. *PeerJ* **2**:e453 DOI [10.7717/peerj.453](https://doi.org/10.7717/peerj.453).
- Villanueva-Rivera LJ, Pijanowski BC. 2012.** Pumilio: a web-based management system for ecological recordings. *Emerging Technologies* **93**:71–81 DOI [10.1890/0012-9623-93.1.71](https://doi.org/10.1890/0012-9623-93.1.71).
- Walther G-R, Post E, Convey P, Menzel A, Parmesan C, Beebee TJC, Fromentin J-M, Hoegh-Guldberg O, Bairlein F. 2002.** Ecological responses to recent climate change. *Nature* **416**(6879):389–395 DOI [10.1038/416389a](https://doi.org/10.1038/416389a).
- Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. 2004.** Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* **13**(4):600–612 DOI [10.1109/TIP.2003.819861](https://doi.org/10.1109/TIP.2003.819861).
- Xie J, Michael T, Zhang J, Roe P. 2016.** Detecting frog calling activity based on acoustic event detection and multi-label learning. *Procedia Computer Science* **80**:627–638 DOI [10.1016/j.procs.2016.05.352](https://doi.org/10.1016/j.procs.2016.05.352).
- Zhang L, Towsey M, Xie J, Zhang J, Roe P. 2016.** Using multi-label classification for acoustic pattern detection and assisting bird species surveys. *Applied Acoustics* **110**:91–98 DOI [10.1016/j.apacoust.2016.03.027](https://doi.org/10.1016/j.apacoust.2016.03.027).