Classification of Tropical Disease-carrying Mosquitoes Using Deep Learning and SHAP

Vinicius L. N. Fonseca[1], Fagner Cunha[1], Larissa Andrade[1], Juan G. Colonna[1], David De Yong[2]

[1]Instituto de Computação – Universidade Federal do Amazonas (UFAM)
Caixa Postal 69077-470 – Manaus – AM – Brazil

[2]Facultad de Ingeniaría – Universidad Nacional de Río Cuarto (UNRC)
Postal Code X5800 – Río Cuarto – Córdoba – Argentina

{vinicius.fonseca, fagner.cunha, las, juancolonna}@icomp.ufam.edu.br

ddeyong@ing.unrc.edu.ar

Abstract. *In this paper, we present a novel technique for identifying mosquitoes that carry tropical diseases using Deep Learning and SHAP for model interpretability. We propose an end-to-end deep (E2E) Convolutional Neural Network (CNN) architecture that leverages mosquito wingbeat sounds to extract relevant features. To achieve high-performance audio processing, we integrate Kapre, an audio processing library optimized for GPU execution. Our approach also incorporates SHAP to provide a transparent explanation of the model's predictions, enabling us to identify and characterize the time-frequency patterns that the model emphasizes. Ultimately, our research aims to support disease control initiatives by providing an automated means of identifying disease-carrying mosquito species, which has the potential to improve public health in tropical regions.*

1. Introduction

Mosquitoes are one of the world's most severe health hazards, capable of transmitting serious diseases such as: Malaria, Zika virus, Chikungunya, Yellow Fever, Dengue, West Nile Virus, Lymphatic Filariasis, and many forms of encephalitis. It is estimated that about 700 million people are infected annually, resulting in more than one million deaths per year [Caraballo and King, 2014].

The most effective method for preventing vector-borne diseases is through the use of insecticides. However, extensive monitoring and surveillance are necessary to ensure their effectiveness. Some health security departments have implemented mosquito surveillance and control programs based on trap monitoring [Lühken et al., 2014]. For these programs to be successful, it is crucial to develop an accurate identification tool. Traditional methods of mosquito classification rely on labor-intensive and time-consuming manual identification, which is susceptible to human error [Kampen et al., 2015]. Therefore, there has been a recent surge of interest in using deep learning techniques and machine learning models to automate this process. Such tools are critical for the effective implementation of control and prevention strategies.

The most commonly used traps for capturing mosquitoes record the wingbeats of the insects in an audio signal format. Thus, the mosquitoes are classified based on the

frequency of their wingbeats, a method that has been extensively researched in the field of entomology. This is because each mosquito species has a unique acoustic signature that can be used to identify them in the field. Traditional methods for mosquito classification based on wingbeat frequency have relied on signal processing techniques such as Fourier Transform [Rohlf and Archie, 1984] and Mel Frequency Cepstral Coefficients (MFCC) [Logan et al., 2000].

This paper presents a novel approach to classifying disease-carrying mosquitoes using audio signals recorded by traps. Our proposed method is an end-to-end (E2E) Deep Learning (DL) model based on Convolutional Neural Networks (CNNs). To prepare the audio signals for input to the CNN model, we convert them into spectrogram images, which provide a time-frequency representation. Our DL model is based on the Biophony architecture, which has shown promising results in related bioacoustics classification tasks [Fleishman et al., 2020]. By using this architecture, we aim to extract relevant features from the audio signals that can accurately identify and classify disease-carrying mosquitoes.

This approach has the potential to significantly improve mosquito identification and classification, leading to more effective disease control and prevention measures. Our results demonstrate the effectiveness of the proposed approach accurately classifying different species of mosquitoes. In addition to our proposed E2E model, we also employ SHapley Additive exPlanations (SHAP) to interpret the model decisions. By gaining insights into the important features learned by the CNN model, we can improve mosquito identification and classification.

2. Related Works

Deep learning methods typically require a significant amount of labeled training data. However, Ko et al. [2018] proposed a solution to this challenge by combining multiple pre-trained convolutional neural networks (CNNs). This approach involves concatenating features produced by the CNNs, followed by a dimensionality reduction using linear discriminant analysis (LDA). The classification is then performed using an Support Vector Machine (SVM). This method has been successfully used to classify sounds of anuran, bird, and insect species, and outperformed other types of CNN architectures in terms of overall accuracy.

Ntalampiras [2019] proposed a solution for insect species classification based on the sounds of their wingbeats. Their method utilizes a Hidden Markov Models (HMM) – a specific type of Directed Acyclic Graph (DAG) – for classification. This approach reduces the amount of required training data, which is beneficial for small bioacoustic datasets. A major advantage of this method is that it does not require model retraining when new insect sounds are available, and its DAG structure allows for easy interpretability. Although HMMs have shown good results in various applications, their accuracy is limited as they rely on handcrafted features that need to be mapped to discrete alphabetic symbols. This process can be time-consuming and may not capture all relevant information present in the data.

Nolasco et al. [2019] proposed a machine learning solution to classify beehive states using audio data obtained from the NU-Hive project, which aims to monitor beehives' conditions by analyzing the sounds bees make. The survival of bees is crucial

as they are the most important pollinators of food crops globally. The authors compare the performance of SVM and CNNs for identifying the states of different beehives using features based on Hilbert-Huang Transform (HHT) and Mel-Frequency Cepstral Coefficients (MFCC). Their findings indicate that SVM outperformed CNNs in generalizing to new data. However, the method has limitations in handling signals of arbitrary size, and pre-processing steps increase the demand for computing resources.

In recent years, mosquito wing-beat frequency capture techniques have evolved significantly, moving away from the use of common microphones that forced mosquitoes to stay close and behave in an unnatural way when compared to their behavior in the wild. This caused machine learning models to be trained on unrealistic environment, leading to biases in the results when transfer to a real scenario. To address this issue, an optical capture device was developed by Potamitis and Rigakis [2015], thereby improving the quality of the samples for training machine learning models.

Despite the advances in this research field, most of related works rely on methods that use handcrafted feature extraction before a CNN, *e.g.* spectrogram generation. Also, CNN are used as "black boxes"without providing much insight into how the model arrived at a particular result. Therefore, our goal is to propose an end-to-end model and attempt to explain how it arrived at a given outcome.

3. Material and Methods

3.1. Dataset

The Wingbeats dataset consists of recordings in the *.wav* format, which were captured using optical devices. The dataset includes recordings from insectaries, each containing one species of mosquitoes of both sexes. As the mosquitoes fly, their movement causes oscillations in the captured light signal of the sensor, which is transformed into a pseudo-acoustic signal. Each audio snippet in the dataset has a length of 5000 samples and a sampling rate of 8KHz [Potamitis and Rigakis, 2015]. The total number of samples is shown on Table 1.

Tabela 1. Data Distribution in the Wingbeats Dataset

Species	**number of samples**
Ae. aegypti	85553
Ae. albopictus	20231
An. gambiae	49471
An. arabiensis	19297
Cu. pipiens	30415
Cu. quinquefasciatus	74599
Total	279566

In the data partitioning phase, we carefully split the dataset into training, testing, and validation subsets to ensure a fair evaluation of the proposed CNN model. To increase the diversity of data present in each set and prevent the same mosquito from appearing in different sets, we selected different specimens records for each set. Moreover, we also ensured that the dataset was balanced across different classes to avoid bias towards any

particular species. A detailed information on the partitioning of the Wingbeats dataset is provided in Table 2.

Tabela 2. Distribution of data into train, test and validation sets.

Subset	Ae. aegypti	Ae. albopictus	An. gambiae	An. arabiensis	C. pipiens	C. quinquefasciatus	Total
Train	14627	14175	14383	13684	14272	14539	85680
Test	5165	5001	5074	4824	5026	5137	30227
Validation	869	827	838	789	807	870	5000
Total Species	20661	20003	20295	19297	20105	20546	120907

3.2. Metrics

To evaluate the performance of our model, we utilized a confusion matrix, which is a commonly used tool in machine learning to describe the results of the predictions made by the model. A confusion matrix includes four components: True Positive (TP); True Negative (TN); False Positive (FP); and False Negative (FN). With the confusion matrix, it is possible to obtain commonly used metrics such as Precision (P), Recall (R), F1-score (F1), and Accuracy (Acc). Precision measures the proportion of correct positive class predictions in relation to the total number of samples classified as positive (Equation 1). Recall measures the proportion of true positive samples that were correctly classified by the model in relation to the total number of positive samples (all samples that should have been classified as positive) (Equation 2). F1-score is a metric used to balance precision and recall (Equation 3). Accuracy measures the proportion of correct predictions made by the model in relation to the total number of samples (Equation 4).

$$\mathrm{P} = \frac{TP}{TP + FP} \quad (1) \qquad \mathrm{R} = \frac{TP}{TP + FN} \quad (2)$$

$$\mathrm{F1} = \frac{2PR}{P + R} \quad (3) \qquad \mathrm{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

3.3. Kapre layers

A Kapre layer[1] is a simple and efficient way to process audio data with Keras. It serves as the first layer of the model, standing between the input and convolution network, performing time-to-frequency conversions, normalization, and data augmentation, all with a focus on real-time execution on the GPU [Choi et al., 2017]. Figure 1 illustrates who it was integrated into our E2E model.

[1] https://github.com/keunwoochoi/kapre

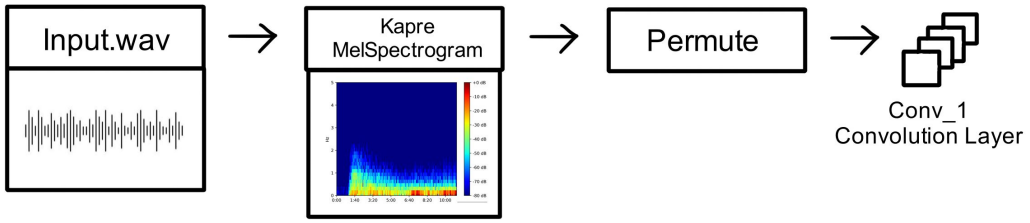


Figura 1. Implementation of the Kapre Layer

In our model we used the `get_melspectrogram_layer()` method that transforms the audio input to a Mel spectrogram, which uses two 1-dimensional convolutions initialized with real and imaginary part of discrete Fourier transform kernels respectively, following the definition of discrete Fourier transform:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot [\cos(2\pi kn/N) - i \cdot \sin(2\pi kn/N)] \tag{5}$$

where $k \in [0, N-1]$, N is the signal length, and x_n are signal samples. The complete E2E model is implemented with `conv2d()` layers of Keras backend, which means the Fourier transform kernels can be trained with backpropagation. The `get_melspectrogram_layer()` is an extended conversion of `layerMelSpectrogram()` based on `Spectrogram()` with a multiplication by a mel-scale matrix from linear frequencies, which can be trained [Choi et al., 2017].

3.4. Shapley Additive Explanations

SHAP (Shapley Additive Explanations) is a model interpretation method proposed by Lundberg and Lee [2017] that provides a theoretical approximation to explain the reasoning behind a model's prediction. This method is implemented by a library that can be used to interpret different types of models, such as linear regression, logistic regression, decision trees, Neural Networks, among others. The goal of SHAP is to provide an interpretable explanation for any machine learning model predictions, by presenting the individual contribution of each feature in predicting a specific sample. To achieve the explanation, it calculates Shapley values, which are based on cooperative game theory, for each feature of the input, such as pixels of an image or attributes of a dataset. In summary, SHAP has unique mathematical properties that ensure the importance of the explanations and it is also a local method, meaning that it works for individual samples. The SHAP equation is:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j, \tag{6}$$

where g is the model explanation, $z' \in \{0,\ 1\}^M$ the coalition vector, with M being the maximum size of the coalition vector, and $\phi_j \in \mathbb{R}$ representing the Shapley value assigned to feature j. The coalition vector is a mathematical representation that captures the contribution of each feature to the prediction of a specific instance. It is calculated using the Shapley value theory, which provides a fair way of distributing the "payment"(i.e.,

the prediction) among the features. An entry of 1 in the coalition vector indicates that the feature is “present,”while 0 indicates that it is “absent” [Molnar, 2020].

The SHAP library offers various explainers for machine learning models. We use the `Gradient_Explainer()` method, which uses expected gradients. Expected gradients combine the ideas of Integrated Gradients [Sundararajan et al., 2017], SHAP, and SmoothGrad [Smilkov et al., 2017] into a single equation. This approach allows for the use of the entire dataset as the reference distribution (instead of just one reference value) and enables local smoothing. If we approximate the model using a linear function between each reference data sample and the current input to be explained, assuming that the inputs are independent, then expected gradients calculate approximate SHAP values.

4. Proposed Model

In our E2E model, we utilized as backbone the convolutional architecture of the Microsoft Biophony model without incorporating any pre-trained weights[2]. To prevent overfitting, each convolution layer was followed by a Max Pool operation and a 25% Dropout. Additionally, we incorporated the Kapre layer, as discussed in Section 3.3. We modified the last classification layer to accommodate our multi-class problem, which includes six distinct classes.

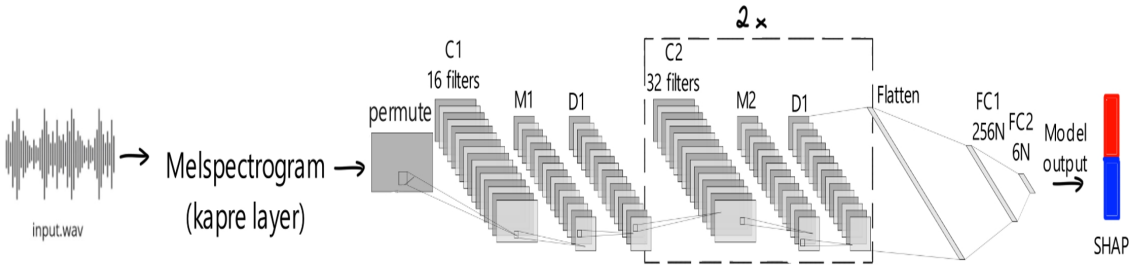


Figura 2. CNN model architecture. The convolution layers are represented by the abbreviation ”C”, the Max Pooling layers by ”M”, and the fully connected layers by ”FC”(Fully Connected).

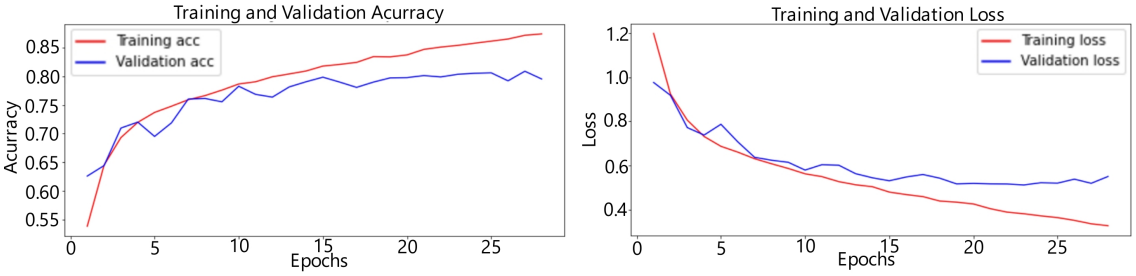
The main advantage of using an end-to-end model over a traditional task-oriented method is that the former requires less expert knowledge, leading to faster and more efficient training. As show by Chen et al. [2014] and Fanioudakis et al. [2018], task-oriented methods involve manual feature engineering, which can be time-consuming, subjective and can also introduce modeling bias. On the other hand, an end-to-end model learns the features from the raw data itself, eliminating the need for manual feature engineering. This approach results in a more robust model that is less prone to over-fitting and generalizes better to new data.

5. Experiments and Results

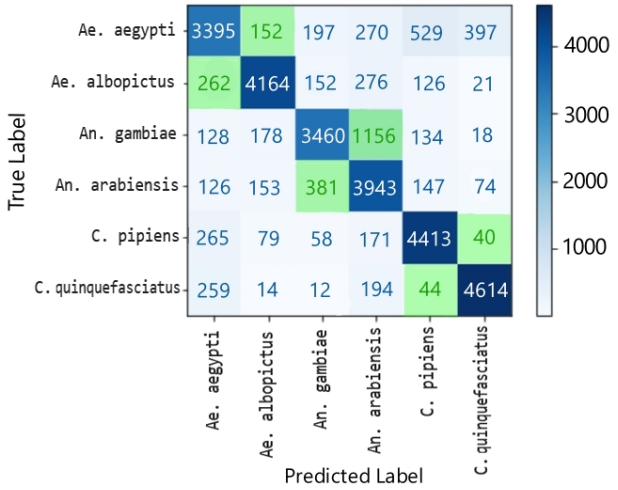
We trained the model using `EarlyStopping()` callback with a patience of five epochs and a Δ of 0.00001, which stop training when the validation loss has stopped improving. Thus, resulting in 27 epochs of training. We observed in Figure 3a that the curves stabilize during training, indicating that the model is learning the data correctly and not just fitting

[2] `https://github.com/microsoft/acoustic-bird-detection`

to the training set (overfitting). Figure 3b shows the confusion matrix generated by the final CNN model and the classification report on the test set.



(a) Training and Validation Loss and accuracy Graphs.



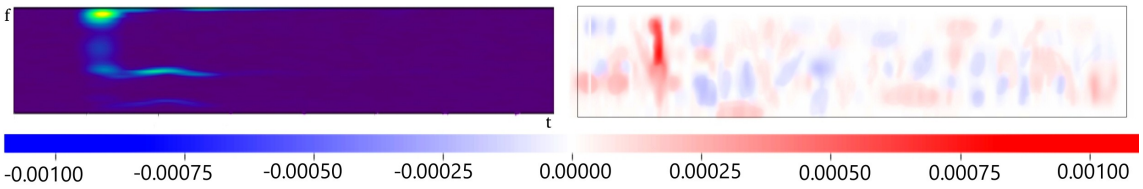
	precision	recall	f1-score
Ae. aegypti	0.77	0.66	0.71
Ae. albopictus	0.84	0.83	0.84
An. gambiae	0.81	0.68	0.74
An. arabiensis	0.66	0.82	0.73
C. pipiens	0.82	0.88	0.85
C. quinquefasciatus	0.89	0.90	0.90
accuracy			0.79
macro avg	0.80	0.79	0.79
weighted avg	0.80	0.79	0.79

(b) Confusion Matrix and metrics values.

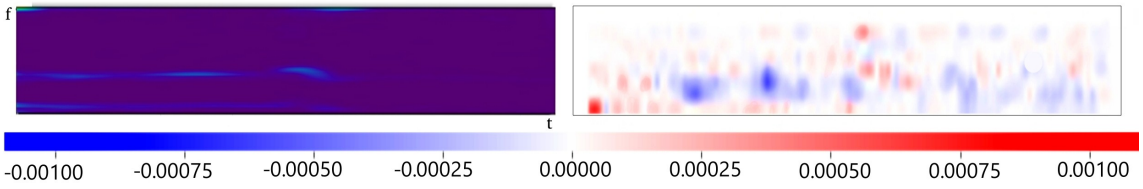
Figura 3. Model training and test performance.

As depicted in Figure 3b, the model faced difficulty in distinguishing between *An. gambiae* and *An. arabiensis*, with incorrect predictions made 1156 times. A similar confusion occurred between *Ae. aegypti* and *Ae. albopitcus*. However, this is not a significant issue when it comes to identifying the spread of a particular disease transmitted by mosquitoes. These pairs of mosquitoes transmit the same disease, which is why they are colored green in Figure 3b. Our concerns lie in the predictions that fall outside this category, such as the confusion between *Ae. aegypti* and *C. pipiens*, due to the fact that they are vectors of different diseases.

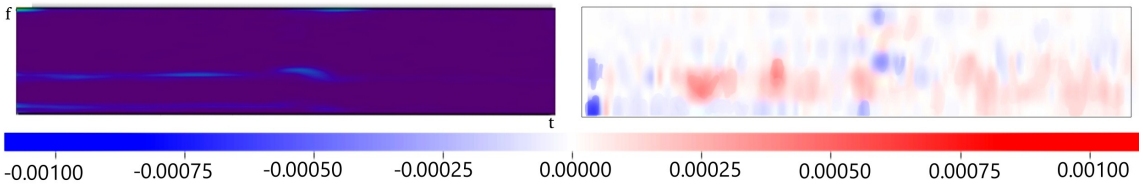
The distinctive methodology of this work lies in its attempt to offer an explanation for the CNN's decision-making process and why the model is struggling with certain classes. To achieve this, the Gradient Explainer method of SHAP is employed, with the outputs displayed in the following figures 4a, 4b and 4c. The focus is on the inputs and true labels for the class "**Ae. aegypti**"to interpret the model's decision-making process, instead of mere speculation. The primary objective is to unravel the model's workings and avoid it being labeled as an opaque black box. In summary, this methodology aims to shed light on the model's decision-making process.



(a) A correctly classified spectrogram of Ae. aegypti and its SHAP values.



(b) An incorrectly classified spectrogram of Ae. aegypti and its SHAP values.



(c) An incorrectly classified spectrogram of Ae. aegypti missclassified as Cu. pipiens. This SHAP values belongs to the Cu. pipiens.

Figura 4. Three Ae. aegypti spectrograms generated by the Kapre layer and its feature importance obtained with SHAP.

Upon analyzing Figures 4a and 4b, a significant difference can be observed. The main distinction is that Figure 4a exhibits a substantial concentration of frequency energy in a short period of time. This pattern played an important role in the model classifying the input as belonging to the class *Ae. aegypti*. On the other hand, Figure 4b lacks this time-frequency pattern, which caused the model to misclassify the input as a *Cu. pipiens*. We can also observe that Figure 4c is essentially the inverse of Figure 4b, where all the important pixels are inverted. The area present in Figure 4c is of utmost importance to classify correctly. This is probably due to the method in which the audio was captured. Although it presented activities of the wingbeat, they were extremely short, which confused the CNN.

6. Conclusion and Future Research

Our E2E CNN model has demonstrated high accuracy in classifying mosquitoes based on their wingbeat sounds, which is a promising achievement. The ability to accurately identify different mosquito species has significant implications for public health, as it can aid in mosquito control efforts and prevent the spread of mosquito-borne diseases. By using this model, researchers and public health professionals can quickly and accurately identify mosquito species, allowing them to develop effective prevention and control measures. In addition, this technology could be used to detect new or emerging mosquito species, allowing for early intervention and control. Overall, the potential impact of this model on public health is substantial, making it a valuable tool in the fight against mosquito-borne diseases.

In this paper, we extend the Biophony model by incorporating a Kapre layer to enhance feature extraction compared to traditional handcrafted features. We also introduce the use of SHAP and its Gradient Explainer method for interpreting our machine learning model. Our findings demonstrate that SHAP enables clear visualization of the time-frequency patterns learned by the CNN model for certain mosquito species. To the best of our knowledge, our work is the first to apply SHAP to explain model decisions in bioacoustics research, despite its simple and intuitive interface in image analysis.

In future work, we plan to improve the efficiency of our model by implementing the Keras method `flow_from_directory()`, which will allow us to avoid RAM limitations and handle larger datasets. We also aim to optimize the model for deployment on low-resource hardware, with the ultimate goal of developing an efficient trap device for mosquito control. Additionally, we will explore other model interpretation methods, such as Local Interpretable Model-Agnostic Explanations (LIME) proposed by [Ribeiro et al., 2016] and Gradient-weighted Class Activation Mapping (grad-CAM) introduced by [Selvaraju et al., 2017], and compare their performance with SHAP.

7. Acknowledgements

The present work is the result of the Research and Development (R&D) project 001/2020, signed with Federal University of Amazonas and FAEPI, Brazil, which has funding from Samsung, using resources from the Informatics Law for the Western Amazon (Federal Law nº 8.387/1991), and its disclosure is in accordance with article 39 of Decree No. 10.521/2020.

Referências

H. Caraballo and K. King. Emergency department management of mosquito-borne illness: malaria, dengue, and west nile virus. *Emergency medicine practice*, 16(5):1–23, 2014.

Y. Chen, A. Why, G. Batista, A. Mafra-Neto, and E. Keogh. Flying insect detection and classification with inexpensive sensors. *JoVE (Journal of Visualized Experiments)*, page e52111, 2014.

K. Choi, D. Joo, and J. Kim. Kapre: On-gpu audio preprocessing layers for a quick implementation of deep neural network models with keras. In *Machine Learning for Music Discovery Workshop at 34th International Conference on Machine Learning*. ICML, 2017.

E. Fanioudakis, M. Geismar, and I. Potamitis. Mosquito wingbeat analysis and classification using deep learning. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 2410–2414. IEEE, 2018.

A. Fleishman, C. Eberly, D. Klein, and M. McKown. Tutorial: Accurate Bioacoustic Species Detection from Small Numbers of Training Clips Using the Biophony Model. https://github.com/microsoft/acoustic-bird-detection, 2020.

H. Kampen, J. M. Medlock, A. G. Vaux, C. J. Koenraadt, A. J. Van Vliet, F. Bartumeus, A. Oltra, C. A. Sousa, S. Chouin, and D. Werner. Approaches to passive mosquito surveillance in the eu. *Parasites & vectors*, 8:1–13, 2015.

K. Ko, S. Park, and H. Ko. Convolutional feature vectors and support vector machine for animal sound classification. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 376–379. IEEE, 2018.

B. Logan et al. Mel frequency cepstral coefficients for music modeling. In *Ismir*, volume 270, page 11. Plymouth, MA, 2000.

R. Lühken, W. P. Pfitzner, J. Börstler, R. Garms, K. Huber, N. Schork, S. Steinke, E. Kiel, N. Becker, E. Tannich, et al. Field evaluation of four widely used mosquito traps in central europe. *Parasites & Vectors*, 7:1–11, 2014.

S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

C. Molnar. *Interpretable machine learning*. Lulu.com, 2020.

I. Nolasco, A. Terenzi, S. Cecchi, S. Orcioni, H. L. Bear, and E. Benetos. Audio-based identification of beehive states. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8256–8260. IEEE, 2019.

S. Ntalampiras. Automatic acoustic classification of insect species based on directed acyclic graphs. *The Journal of the Acoustical Society of America*, 145(6):EL541–EL546, 2019.

I. Potamitis and I. Rigakis. Novel noise-robust optoacoustic sensors to identify insects through wingbeats. *IEEE Sensors Journal*, 15(8):4621–4631, 2015.

M. T. Ribeiro, S. Singh, and C. Guestrin. "why should i trust you?"explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

F. J. Rohlf and J. W. Archie. A comparison of fourier methods for the description of wing shape in mosquitoes (diptera: Culicidae). *Systematic Zoology*, 33(3):302–317, 1984.

R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.