Automatic recognition of frog calls

Eliza G. Foran[1], Evan D. Suggs[2], Tenecious A. Underwood[3] , Winona Snapp-Childs[1], Sheri A. Sanders[1]
[1]Indiana University, [2]University of Tennessee at Chattanooga, [3]Livingstone College

INTRODUCTION

Frogs can be considered the “canary in the coal mine” when it comes to environmental problems. This is because frogs are especially vulnerable to disease, pollutants, and habitat loss. Recent shifts in amphibian biodiversity, where around 81% of species are decreasing rapidly and 18% are increasing [1] are of broad concern. Thus, the focus of this project is to provide a tool for quantifying frog biodiversity in the age of the amphibian crisis. To do this we, we expanded a previously made Jetstream [2,3] workflow with deep machine learning with both convolutional neural networks (CNN) and recurrent neural networks (RNN) to enable accurate, automatic identification of frog calls.

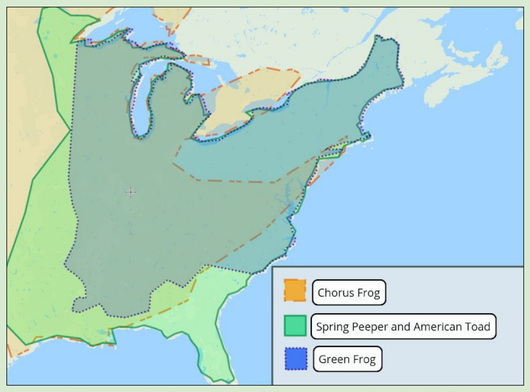


Figure 1: Range of frog species in US (data from United States Geological Survey)

METHODS

Data curation and processing

Over 5000 raw audio recordings of four species of frogs (Chorus Frog, Spring Peeper, American Toad, and Green Frog) were collected from Cornell's Macaulay Library archives of wildlife sounds [4].

We reduced audio recordings to be single channel (left and right hearing combined), down-sampled the (sampling) rate to 16 kHz, and segmented the files to be nine seconds resulting in 2828 available samples.

Chorus Frog

Spring Peeper

American Toad

Green Frog

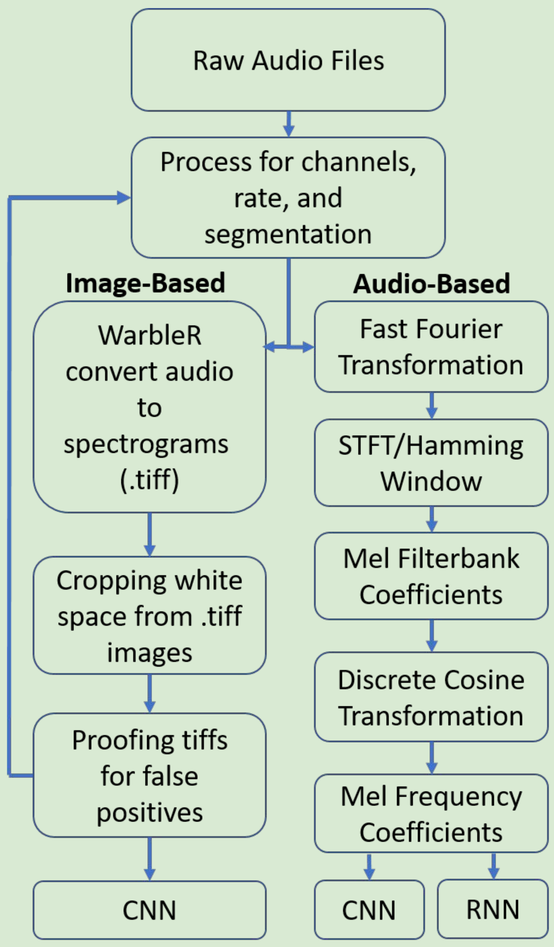
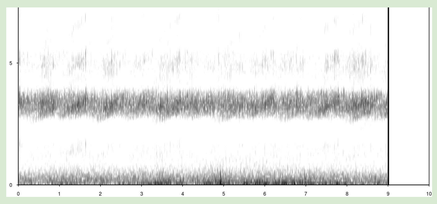
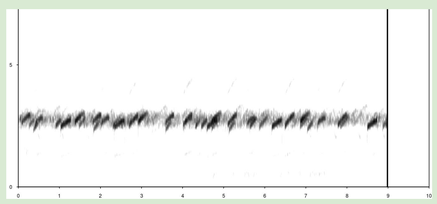


Figure 2: Workflow applied for audio and image-based neural networks within Jetstream.

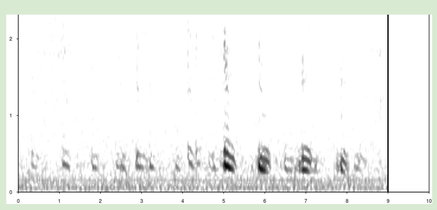


Chorus Frog



Spring Peeper

American Toad



Green Frog

Figure 3: Spectrograms of the four frog species.

To achieve a simple, successful model we first trained our three neural networks with two species (Chorus Frog, Spring Peepers). Then we added two additional species (American Toad, Green Frog), retraining the three neural networks with four species.

Table 1. Proportion of data selected for model training versus prediction

	CNN-image	CNN-audio	RNN-audio
2 species	Training: 1244 (86.15%) Testing: 200 (13.85%)	Training: 1015 (83.54%) Testing: 200 (16.46%)	Training: 1015 (83.54%) Testing: 200 (16.46%)
4 species	Training: 2428 (85.86%) Testing: 400 (14.14%)	Training: 1911 (82.69%) Testing: 400 (17.31%)	Training: 1911 (82.69%) Testing: 400 (17.31%)

RESULTS

Convolution neural network (CNN) on Images[5]

Input: Cropped frequency over time spectrograms from WarbleR, black and white
Layers: ReLU-activated Conv2D, MaxPool2D, ReLU-activated Conv2D, Dropout, 2 Dense, Batch Normalization, soft-max-activated Dense
Space Used: 4.69 Gigabytes for spectrograms

Convolutional Neural Network (CNN) on Audio[6]

Input: Raw audio files as .wav, 16 kHz, nine second segments
Layers: 3 ReLU-activated Conv2D, Dropout, Flatten, 3 Dense
Space Used: 1.86 Gigabytes for audio files

Recurrent Neural Network (RNN) on Audio[6]

Input: Raw audio files as .wav, 16 kHz, nine second segments
Layers: 2 LSTM, 4 time-distributed Dense with ReLU activation, Flatten, softmax-activated Dense
Space Used: 1.86 Gigabytes for audio files

Table 2. Training times for the three neural networks

	Training Time with 2 species, 4 species (min:sec)		Predicting Time with 2 species, 4 species (min:sec)	
CNN on spectrograms	9:45	21:39	0:33	1:14
RNN on audio	10:01	24:36	1:31	2:53
CNN on audio	13:16	28:28	1:28	2:38

Statistical analyses

- To compare the three models performance for this dataset, we tested the accuracy of the predictions over ten runs of training (Table 3).

Table 3: Prediction accuracy of the three models (avg + std dev %) after 10 runs

	CNN-image (avg ± std dev. %)	CNN-audio (avg ± std dev. %)	RNN-audio (avg ± std dev. %)
2 species	97.72 ± 0.80	99.50 ± 0.00	99.45 +/- 0.16
4 species	97.35 ± 0.38	88.88 +/- 0.89	89.83 ± 0.69

- Non-parametric ANOVA test was performed using Kruskal-Wallis rank sum test on four species models: chi-square = 21.774, df = 2, $p < 0.001$.
  - Tukey HSD post-hoc comparisons showed that:
    - CNN-image > CNN-audio
    - CNN-image > RNN-audio

DISCUSSION

- An audio-based RNN is less accurate with our training and testing sets, but the model will improve with larger number of datasets, such as frog survey inputs. This model does have more sustainable storage options compared to the other models.
  - Future work must include more model training to increase accuracy to > 95%.
- The CNN-image model is currently the most accurate and predicts faster than the other models.
  - We have therefore added the CNN-image model to the previously developed workflow.
- Last year, students built a Jetstream-based workflow that uses Raspberry Pi microcomputers to automatically record frog calls in remote areas and push them to a Drupal webpage for visualization [3]. Our research adds machine learning applications for complete automated identification without having to interpret spectrograms or proof calls by ear. Ideally, this workflow can substitute for national frog call surveys or allow field station researchers to focus on data analysis rather than data collection.

References
[1] R Alexander Pyron. 2018. Global amphibian declines have winners and losers.Proceedings of the National Academy of Sciences 115, 15 (2018), 3739–3741.
[2] Craig A Stewart, David Y Hancock, Matthew Vaughn, Jeremy Fischer, Tim Cockerill, Lee Liming, Nirav Merchant, Therese Miller, John Michael Lowe, Daniel C Stanzione, et al. 2016. Jetstream: performance, early experiences, and early results. In Proceedings of the XSEDE16 Conference on Diversity, Big Data, and Science at Scale. ACM, 22.
[3] Eliza G. Foran, Jazzly Anderson, Tom Slayton, Emmanuel Guido, Thomas G. Doak, Sheri A. Sanders. Developing a workflow for bioacoustic recording devices and frog call analysis within Jetstream; 2019 May 14; Bloomington, Indiana. In Proceedings of the CEWiT Poster Competition.
[4] Wil Hershberger, H C Gerhardt, Brad Walker / Macaulay Library at the Cornell Lab of Ornithology
[5] Dakila Ledesma. 2019. cnn-tutorial.md. https://github.com/bgq527/intro-ml/blob/master/cnn-tutorial.md [Online; accessed <7/2/2019>]
[6] Seth Adams. 2019. Deep Learning for Audio Classification. Youtube. [Online, accessed 7/17/2019].

Acknowledgements
This material is based upon work supported by the National Science Foundation under Grant No. 1445604 (Jetstream) and 1759906(NCGAS) Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Special thanks to Seth Adams and Dakila Ledesma's neural network resources, Jefferson Davis, Carrie Ganote and Laura Huber's for help setting up and debugging our neural networks, the Jetstream system administration team (Steve Bird, Mike Lowe, and George Turner), and Bhavya Papudeshi for guidance with Jetstream.