1 Introduction

1.1 Rainforests

- contain x- one such approach is emerging due to technological and theoretical advancements, this is acoustic monitoring

1.2 Bioacoustic Monitoring

STRENGTHS (huge amount of data which can be used to answer ecological questions, cryptic species, less biases from human presence, can be stored and further work done INCLUDING greatly increased repeatability of any study- an advantage over e.g. visual surveying), ASSESS EFFECTIVENESS OF CONSERVATION STRATEGIES (subtly stress this point, speak to Tom about good papers to bring in showing the importance and neglect of this element in conservation) further advantage is development of audiomoths (cheaper)

however, WEAKNESSES incredibly time-consuming and often infeasible (and another source of bias) to have expert listen to all data collected. - possibly difficult to answer some ecological questions e.g. understanding of diel activity of vocal patterns.

there is a strong incentive to incorporate latest advances in...

1.3 Machine Learning

- brief sentences summing up what machine learning is, what it can do - slow uptake point (lack of communication between ML researchers and ecologists) - list some examples of (non-deep) ML used in ecology, mention that even in relatively recent works it was very hands-on and manual as well (see papers in tom-linked - Wrege's elephants, primates) - quite often is off-the-shelf, with limited success (Knight et al. 2017 review), especially in...

1.4 Very Biodiverse Regions

- particular limitation is in very biodiverse regions, due to challenges with hand-selected features - REVIEW OF ML IN VERY BIODIVERSE REGIONS, **this is a key feature of my project - stress **GAP IN DATABASES AND WORK** as described by Browning et al. 2017 - therefore, there it is a big deal that emerging methods which automatically select features ('feature learning') are having a great deal of success. one such method is...

1.5 Deep Learning

- advantages of deep learning over other methods (could perhaps be very few references to recent reviews or papers that used a couple). - examples of deep learning methods used in ecology - ***find key examples of it being more successful - difficulty has been in getting labelled datasets big enough - people have been working on this in recent years, lots of work on **data augmentation** (only specific augmentation methods may be effective given the classes of this problem - see Salamon and Bello 2016 for variable effectiveness of augmentation) and **transfer learning**

good example of all of the above coming together for high impact conservation research is...

1.6 Monitoring of Spider Monkeys

- suitability of species for this type of monitoring approach: heavily reliant on acoustic communication due to being almost entirely arboreal, frugivorous (patchily-distributed food) with complex fission-fusion societies - have an array of calls of known meaning (but fundamentally we'll first be working with presence/absence only) - high impact, very important species for ecosytem functioning (spread loads of seeds) in incredibly biodiverse areas - conservation status?

all things considered...

1.7 What I will do is..

- apply deep learning techniques to create an **automated detection system for spider monkey whinnys**. following similar work done previously (e.g. link with Kahl et al. 2017, but more on this in methods e.g. setting best parameters for CNNs etc) - will then use this to answer **ecological question** - what are the daily vocal activity patterns of spider monkeys? - when in the day are spider monkeys calling, when are they not calling. - this type of research has been done before for a small number of other species - this will act as pilot test to maximise efficiency for wider project, in which whole of Osa Peninsula of Costa Rica will be acoustically-monitored, and deep learning will enable the collection of a huge amount of ecological data on the spider monkeys - (possibly even for multiple calls) - reason being for design of wildlife corridors suitable for A. geoffroyi - as habitat of suitable for this target species is known to then be of sufficient quality for a number of other threatened species - to connect the populations currently isolated on the peninsula with unnocupied suitable habitat further inland)

- sentence describing technological and theoretical advancements leading to this being an exciting time in ecology, an incredibly powerful tool for species monitoring at a critical time

CONCLUDING PART OF INTRO - clearly define aims of research project and any hypotheses tested

aim - create an automated detection and classification system for spider monkey calls, accurate enough (or with the potential to be accurate enough, if subsequently labelled examples of calls are used to supplement training) to be used as a tool in species occupancy modelling the distribution on the Osa Peninsula of Costa Rica

2 Methods

2.1 Data: collection and labelling

- fieldwork: - recordings of rainforest collected using AudioMoth recording devices - devices were fastened to trees for approximately three days - possibility of damamge influenced placement, water-damage meant we didn't collect as much as was possible but a lot was collected (work that goes towards filling data-gap in recordings from these regions) - 1 month of data-collecting - ASK JENNA FOR ESTIMATE FOR HOW MUCH SHE HAD LISTENED TO. perhaps. - labelling using Praat files, into regions of 'call' and 'non-call' - consistent labelling, done by one person (authority figure, several years experience listening to spider monkey calls) - positive clips: 3 seconds long as all calls fit within this window, average duration of call equals DO THIS CALC AT LATER DATE - negative clips: 3 seconds long, sampled randomly from regions of call-containing 60-second clips known to not contain calls - further negative clips generated from early runs of system on folders of files. same person listened to all three-second clips that had been pulled to one side with a confidence score of 60% or above, this generated some more positive clips, but a great deal of negative clips - original number of positives equalled X from data collected on Osa Peninsula plus Y from other sources

2.2 Data: preprocessing, augmentation

- standardised - denoised, denoising function of 2013 team used by Kahl et al. in 2017. subtracts mean amplitude from each frequency band

- AUGMENTED: I followed several augmentation methods of Kahl et al. 2017 - Gaussian noise - noise blending - vertical pitch roll

plus I used a random crop augmentation to ensure the network was trained on calls that had been interupted part-way through (a very important stage as the full system splits minute-long files into three-second clips for testing, making it very likely that calls will be at least slightly interupted)

To investigate the effects of the various preprocessing techniques, I trained and tested the system on with and without all changes in all combinations.

- augmentations are crucial in increasing the sample sizes of rare sounds for training CNNs (which are known to require a large amount of data, e.g. thousands of examples at a minimum, to reach state-of-the-art results) - they also increase generalisability of the system, if the training data undergoes augmentations that mimic a wide variety of conditions possible in the test data (such as overlapping of signal and a prominent noise, like a howler monkey or a chicken), that may well have been absent from the training data

2.3 Detector: design and training

- general architecture taken from Salamon and Bello 2016, a standard design that achieved excellent results in a similar problem - ADD FIGURE OF ARCHITECTURE - implemented in Python package Keras, using Theano as a backend - trained using three-second clips, labelled as either being negative ('0') or positive ('1') - although initially keeping the classes (positive and negative) balanced, as is often done in machine learning, I experimented with a biased training ratio which was more reflective of the true ratio of positive to negative clips in the original data - mention hyperparams to set up next section

2.4 Detector: optimisation and testing

- performance evaluated using stratified 10-fold cross-validation - generates ten sets of train/test data from original dataset, maintaining a similar ratio of positives and negatives. the model is then retrained for each set giving a comprehensive picture of model performance (a chosen metric can be reported as the average of ten indepenent training instances, with a measure of spread such as standard deviation) - although it's recommended that general overall network architecture is taken from the literature, within that, optimal hyperparameters for neural networks vary on a problem-by-problem basis (WHY?). therefore, I used a grid search approach to test all combinations of a number of different options (activation layer, optomiser algorithm, number of hidden layers, including batch-normalisation or not, learning rate,

number of epochs to train network for, dropout percentage) - grid search implemented in Python package hyperas

2.5 Overall system design and functionality

- iterates over folder of 1-minute long files (as produced by AudioMoth recording devices, BUT WILL HOPEFULLY BE ABLE TO WORK WITH (almost) ANY LENGTH FILE) - for each file, splits it into twenty 3-second clips - each clip is then fed into the CNN, which outputs the resulting activation value of the last (output) layer of the network (a value between 0 and 1), and if this value is above a threshold activation value (e.g. 0.7) - specified by the user at runtime - the clip is considered to contain the signal - the system was tested (precision, recall etc.) with a threshold of X, but the option is there to alter the threshold to alter the sensitivity of the system (tradeoff between false positives and false negatives)

OUTPUTS OF SYSTEM: - a folder containing all detected-positive three second clips (informatively labelled with the original file name of the 60-second clip, the time location within the clip they came from i.e. the start and end of three-second interval, a number representing which of the total number of detected clips from their file they were, and the activation value multiplied by 100 acting as a proxy of confidence) e.g. - a summary CSV file of all detected clips, with headings file name, approx. position in recording (secs), the time and date of recording of the original file (calculatable as the recording device automatically labels the files using a hexadecimal code representing the date-time of recording), and the confidence of the CNN's classification (again, calculated using activation value of output layer for each clip).

RUNTIME OF SYSTEM (or perhaps this should be in Results): approx. one second per file, so a folder of four thousand 60-second recordings takes just over an hour. TEST THIS AGAIN WHEN I DO OVERNIGHT RUNS.

3 Results

figure X figure Y

4 Discussion

4.1 Why Do I Think It Didn't Work

4.2 What Would I Do To Improve It With More Time

- Other Algorithms
    - genetic algorithms: more efficient way to search for hyperparameters -