# Biomimetic Detection of Dynamic Signatures in Foliage Echoes

Ananya Bhardwaj

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

Rolf Müller, Chair

Alexander Leonessa

Michael J. Roan

December 16, 2020

Blacksburg, Virginia

# Biomimetic Detection of Dynamic Signatures in Foliage Echoes

Ananya Bhardwaj

(ABSTRACT)

Horseshoe bats (family Rhinolophidae) are among the bat species that dynamically deform their reception baffles (pinnae) and emission baffles (noseleaves) during signal reception and emissions, respectively. These dynamics are a focus of prior studies that demonstrated that these effects could introduce time-variance within emitted and received signals. Recent lab-based experiments with biomimetic hardware have shown that these dynamics can also inject time-variant signatures into echoes from simple targets. However, complex foliage echoes, which comprise a large portion of the received echoes and contain useful information for these bats, have not been studied in prior research. We used a biomimetic sonarhead which replicated these dynamics, to collect a large dataset of foliage echoes (>55,000). To generate a neuromorphic representation of echoes that was representative of the neural spikes in bat brains, we developed an auditory processing model based on Horseshoe bat physiological data. Then, machine learning classifiers were employed to classify these spike representations of echoes into distinct groups, based on the presence or absence of dynamics' effects. Our results showed that classification with up to 80% accuracy was possible, indicating the presence of these effects in foliage echoes, and their persistence through the auditory processing. These results suggest that these dynamics' effects might be present in bat brains, and therefore have the potential to inform behavioral decisions. Our results also indicated that potential benefits from these effects might be location specific, as our classifier was more effective in classifying echoes from the same physical location, compared to a dataset with significant variation in recording locations. This result suggested that advantages of these effects may be limited to the context of particular surroundings if the bat brain similarly

fails to generalize over variation in locations.

# Biomimetic Detection of Dynamic Signatures in Foliage Echoes

Ananya Bhardwaj

(GENERAL AUDIENCE ABSTRACT)

Horseshoe bats (family Rhinolophidae) are an echolocating bat species, i.e., they emit sound waves and use the corresponding echoes received from the environment to gather information for navigation. This species of bats demonstrate the behavior of deforming their emitter (noseleaf), and ears (pinna), while emitting or receiving echolocation signals. Horseshoe bats are adept at navigating in the dark through dense foliage. Their impressive navigational abilities are of interest to researchers, as their biology can inspire solutions for autonomous drone navigation in foliage and underwater. Prior research, through numerical studies and experimental reproductions, has found that these deformations can introduce time-dependent changes in the emitted and received signals. Furthermore, recent research using a biomimetic robot has found that echoes received from simple shapes, such as cube and sphere, also contain time-dependent changes. However, prior studies have not used foliage echoes in their analysis, which are more complex, since they include a large number of randomly distributed targets (leaves). Foliage echoes also constitute a large share of echoes from the bats' habitats, hence an understanding of the effects of the dynamic deformations on these foliage echoes is of interest. Since echolocation signals exist within bat brains as neural spikes, it is also important to understand if these dynamic effects can be identified within such signal representations, as that would indicate that these effects are available to the bats' brains. In this study, a biomimetic robot that mimicked the dynamic pinna and noseleaf deformation was used to collect a large dataset (>55,000) of echoes from foliage. A signal processing model that mimicked the auditory processing of these bats and generated simulated spike responses was also developed. Supervised machine learning was used to classify these simulated spike

responses into two groups based on the presence or absence of these dynamics' effects. The success of the machine learning classifiers of up to 80% accuracy suggested that the dynamic effects exist within foliage echoes and also spike-based representations. The machine learning classifier was more accurate when classifying echoes from a small confined area, as compared to echoes distributed over a larger area with varying foliage. This result suggests that any potential benefits from these effects might be location-specific if the bat brain similarly fails to generalize over the variation in echoes from different locations.

# Dedication

*I dedicate this thesis to my parents, Ramesh and Madhu Bhardwaj, for their unending faith in me and their everpresent support whenever I needed it. To my brother Anurag, whose affection and positivity knows no bounds. A little mention for Liverpool F.C., it was there to lift my spirits on the bad days and add to the happiness in the good.*

# Acknowledgments

I would like to thank Dr. Rolf Müller for giving me the opportunity to pursue this research. I appreciate immensely his trust and guidance throughout my time working on this research. His leadership, approach to problem-solving, attention to detail, and seemingly limitless wide-ranging knowledge of science, are the reference points I will always aspire to reach in my career. I would also like to thank my committee members, Dr. Alexander Leonessa and Dr. Michael Roan, for their mentorship and technical advice. I wish to acknowledge Dr. Mohammad Omar Khyam and David Alexandre, who contributed to a significant part of this research. I'd also like to acknowledge Liujun Zhang, for his help in collecting the data for this project, and for his friendship. I'd also like to thank the other grad students in the lab, Joseph Sutlive, Xiaoyan Yin, Ruihao Wang, Peiwen Qiu, Michael Goldsworthy, Luhui Yang, and Shuxin Zhang, for their help along the way.

I would also like to express my gratitude to the Office of Naval Research and the Naval Engineering Education Consortium for supporting this research

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

ADC  Analog to Digital Converter

AUC  Area Under the Curve

BF    Best Frequency

BW   Bandwidth

CF    Constant Frequency

DAC  Digital to Analog Converter

DRNL Dual Resonance Non Linear

DSC  Doppler Shift Frequency

FM    Frequency Modulated

IHC   Inner hair cell

LIAF  Leaky Integrate and Fire

MAV  Micro Air Vehicle

ML    Machine Learning

NEEC Naval Engineering Education Consortium

ONR  Office of Naval Research

PLA   Polylactic Acid

ReLU  Rectified Linear Unit

RK    Response Kernel

RMS   Root Mean Square

ROC   Receiver Operating Characteristic

SLAM  Simultaneous Localization and Mapping

SNR   Signal to Noise Ratio

SRM   Spike Response Model

UART  Universal Asynchronous Reception and Transmission

UAV   Unmanned Air Vehicle

# Chapter 1

# Introduction

## 1.1 Background

### 1.1.1 Bioinspiration

Engineering is the process of finding science-based solutions to real-world problems. Often these solutions are not novel ideas, but instead adaptations of existing concepts. When examples from the biological world serve as the inspiration for solutions to engineering problems, it is termed as Bioinspired Engineering. Bioinspiration is not a novel concept, rather, bioinspiration has led to the development of technology throughout human history. One of the foremost examples is how engineers sought inspiration from birds for the design of airplanes [2], and more recently, insects have inspired flapping winged flight robots and micro aerial vehicles (MAVs) [3, 4]. Biological organisms have developed natural materials through millions of years of evolution possesses to tackle their unique challenges, and this sophistication of materials inspires engineering material and system design [5]. Several species across the breadth of the biological world have served as inspiration to engineers, and this inspiration can provide a more direct route to invention than more traditional branches of science [6]. For example, the branching of trees has inspired the damping of architectural and structural beams [7]. The hair flow sensors of aquatic organisms such as the Mexican blind fish have inspired similar engineered solutions for thermal, piezoelectric, or optical flow

measurements [8]. One of the most exciting and promising areas of bioinspired research is in the study of chemical reactions in biology that lead to solar energy harvesting, given the most consequential challenge to our civilization is the one posed by climate change [9].

### 1.1.2 Bats as a Model of Bioinspiration

Various species constitute sources of bioinspiration for engineers and physicists for a plethora of different applications. Bats are one of these model organisms, and they form an important and interesting case study. Bats are one of the most successful and diverse mammalian orders, with there being over 1,400 identified bat species in the world [10]. Bats also exhibit significant variation in their biology due to this diversity. Bats are found all over the world and have successfully adapted to all different types of environments. It is due to this biological success that bats are a major source of inspiration for engineers in many interest areas. Engineers have mimicked bats' flight dynamics to develop flapping-wing robots for autonomous mission applications such as in the detection of chemicals [11]. Bats have also inspired material scientists' efforts towards developing materials that can replicate the compliance of bat skin on their wings, which enables their flight by allowing for significant deformation with little strain [12]. A special aspect of bat biology, and the most pertinent to this work, is their ability to navigate using echolocation. Echolocation is the process of actively creating sound emissions and then processing the returning echoes to derive knowledge about the environment. Although some other animal groups such as toothed whales [13, 14] and certain birds (oilbirds, swiftlets) [15] have evolved echolocation systems, our focus here is on bats due to some unique challenges they face and the abilities that they possess. Bats employ echolocation to detect what is in their surroundings [16], localize objects [17] and also classify targets [18]. As they operate with air as the medium for sound propagation, bats face significant technical challenges to echolocation, such as spherical spreading loss and

atmospheric attenuation. Additionally, since the targets bats need to detect are small, such as prey insects, they require small wavelengths to achieve significant target strength from the echoes, hence leading to a high pulse frequency which leads to yet further attenuation. Despite these challenges, bats can thrive in these different environments, and effectively navigate, identify and hunt prey, find roosting locations, and evade predators. Bats have been suggested to be capable of homing effectively after being released up to 160 km from their roost [19]. For small distances, less than 8 mile, research has suggested that auditory cues (echolocation) are the primary mechanism for homing in bats [20]. Another recent study has shown that long-tailed bats (*Chalinolobus tuberculatus*) were successfully able to home to their cave after being released at distances from 5 km up to 20 km [21]. Bats have also been suggested to be capable of creating spatial maps in their memory of areas that are larger than their limited operating range of individual echolocation calls [22]. They can also navigate around obstacles in their surroundings, studies in flight rooms have shown bats can avoid stationary as well as moving obstacles using echolocation [23, 24]. Bats adjust the length and frequency of their echolocation calls with their flight speed during the approach to a prey [25]. Bats are also remarkably adept at navigating through dense foliage environments in the dark, such as tropical forests [26]. In this study, bats were also shown to be effective at hunting for prey such as winged insects in cluttered forest environments [26]. Bats have also been shown to successfully identify landmarks and use that information for navigation through openings in nets in behavioral experiments [27]. Another feat of their echolocation ability is their apparent solution to the cocktail party environment problem. When foraging and echolocating in the presence of other bats, they are still able to navigate effectively without confusion due to the echolocation calls of others [28, 29]. Research has also shown the astonishing ability of bats in discriminating surface texture variations on the scale of 1 mm, which shows the sensitivity and sophistication of the biosonar system of bats [30].

## 1.2 Autonomous Navigation and its Opportunities for Bioinspiration

### 1.2.1 Autonomous Navigation and its Applications

Unmanned aerial vehicles (UAVs) or drones are a growing area of innovation and engineering development. Drones have applications where the presence of human operators is dangerous or impossible. One such use case is in the area of atmospheric research, where drones are capable of taking air quality measurements in wide-ranging topographies, as well as near pollution sources that are difficult to reach [31]. Drones are capable of taking other measurements for meteorological purposes and atmospheric modeling, and by virtue of their portability, they offer a high level of spatial and temporal resolution in the measured data [31]. Another area of application for drones is in search and rescue missions [32], and these missions can vary significantly depending on the geography of where they are conducted. The endurance of drones in terms of operating life in one charge cycle, their ability to collect good quality data, and avoid obstacles during navigation are determining factors for their effectiveness. Hence the types of sensors they employ to operate are crucial. Another upcoming and critical area of drone use is in environmental protection, in preventing poaching, unauthorized deforestation, and wildlife trafficking. Drones can also help in environmental protection through monitoring and detection of environmental disasters, such as wildfires [33] which are becoming an increasing issue with the recent spate of wildfires in California and Australia. Other disasters, such as the failure of nuclear powerplants that require monitoring of the spread of nuclear material and radiation is an application well suited for drone use. They can also be used for surveying areas affected by floods or earthquake damage. Other than applications where the advantage of drone use comes from precluding human presence in dangerous areas,

applications where they lead to reduced manual labor are also important. One example is in the increasing drone use by delivery companies to deliver mail and packages directly to the houses of consumers [34]. Here, the efficacy of the drones is affected by the location of the houses and the amount and density of foliage surrounding them. Some houses, with dense foliage surrounding them, pose a challenge to these drones which are more suited to the urban built environment. Some other potential uses of drones are at airports to clear the runways of birds [35]. Researchers are working on developing algorithms that can help herd a flock of birds towards certain areas.

## 1.2.2 Challenges in Autonomous Navigation

There are multiple challenges facing drone design, but a primary challenge lies in sensing for navigation and path planning, especially in conditions where they require navigation through complex environments such as foliage. Forests, particularly below canopy, are often a GPS denied or a GPS challenged environment [36]. Even if GPS is available, its accuracy is low given the poor precision achievable with the portable sensors which can be added to lightweight drones. In addition, GPS is not an effective modality when it comes to flight through foliage instead of flight over it. Traditionally, autonomous UAVs in GPS denied environments have relied on vision-based sensors, i.e., cameras for sensing their surroundings and using the visual information for executing their navigation strategy. However, there are limitations to this approach. Being a passive sensing method, cameras require well-lit conditions for proper operation. This poses an issue not only in nighttime navigation but also when considering navigation through dense forests where light can be limited even during the daytime. Recent studies have relied upon ocular sensors with learning techniques for navigation through relatively sparse forest, but their failure rates have been very high [37]. Navigation algorithms relying on optical sensors also require significant computational power

due to the high dimensionality of the optimization problem using path planning algorithms designed for visual sensors [38]. Another vision-based sensing modality that can be used in drones for navigation in cluttered environments is lidar. Lidar is sensitive to external weather conditions, such as the presence of dust. In a significantly dense dust cloud, lidar scans the edges of the dust cloud as a surface [39]. Sometimes lidar systems even fail to return a range measurement in dusty conditions. Whereas there have been developments of new signal processing algorithms to improve lidar sensing in dusty conditions, the fundamental physics does still pose a challenge to the system [39]. Similar issues arise in vision-based sensing systems when the environmental conditions of rain and fog occur. Another area of limitation in drone flight is the payload weight, which applies to the sensors as well. Heavy sensing systems can lead to decreased flight endurance, hence lighter systems are preferable [40]. This is an area where ultrasound sensors or acoustic approaches can be more beneficial than optical sensors.

### 1.2.3   Bat Bioinspiration in Autonomous Navigation

The limitations of traditional sensing approaches lead to an interest in bioinspired techniques, and bats due to their astonishing flying abilities are an ideal example to learn from. Bats use acoustic signals for navigation and are able to navigate using just one emitter and two acoustic receivers as their sensory system. Their approach is therefore physically lighter, since it requires fewer sensors and potentially computationally inexpensive, especially considering their small brain sizes [41]. Research has shown that using template matching based approaches, it is possible to use bat-like biosonar echoes to classify different locations [42]. In that study, researchers used averaged cochleagrams, inspired from the auditory processing of bats to create templates corresponding to different locations, and were able to match echoes from different cluttered environments to these templates and showed that biosonar

signals can be implemented for scene classification [42]. Researchers have also argued that due to the unique morphology of the bat biosonar system, they are able to encode significant spatial and spectral information using echoes, much more than is possible with standard sonar techniques. Therefore through mimicking their biosonar system it is possible to better solve the autonomous navigational challenge [43]. In that study, a biomimetic approach to create a 3D spatial mapping system was developed, akin to Simultaneous Localization And Mapping (SLAM) used now ubiquitously in robotic navigational systems [43]. This system could localize objects in 3D space using just a single emission. They combined biomimetic hardware, which included microphones placed inside of baffles designed on the physiology of bat pinna, and biomimetic software - which was the processing of acoustic signals akin to a basilar membrane using a filterbank. These filterbank outputs were used to create cochleagrams, which were then fed to a bioinspired SLAM algorithm approach or BatSLAM. Their results showed that they were successfully able to create consistent maps in their testing environments of an indoor office. They also showed their calculations converged quickly to enable navigation based on these generated metric maps to enable distance-based navigation. This approach shows the possibility of using bioinspired sonar approaches instead of standard sonar approaches in conjunction with SLAM to enable much faster navigation with fewer sonar emissions.

Beyond the development of bioinspired scene recognition and robot navigation algorithms, more recently successful efforts have been made to develop robotic systems that use biomimetic hardware and signal processing together to navigate in outdoor environments [44]. Researchers developed an autonomous bat-inspired robot, called 'Robat' which included one ultrasonic emitter and two receivers, and could navigate terrestrially in complex conditions [44]. They employed a low pulse emission rate, 1 per $0.5\,\mathrm{ms}$ which showed the reduction in the data and consequently the computations required compared to a technical sonar array-based

approach. The study showed that not only could a bat-inspired approach achieve obstacle avoidance, their implementation was also capable of object classification as well as mapping the surroundings when encountering an environment for the first time. They also used an artificial neural network to classify the surrounding objects, mainly in two categories 'plant' and 'non-plant' based on the echoes received from these objects with a high rate of accuracy. The algorithm implementation delineated a path based on the map created from the obstacles identified during its run, such that it can use the map for navigation within that environment in the future [44]. This study showed the promise of bat-inspired navigation techniques, as well as highlighting the feasibility of such an approach. These studies and their successes highlight that bats are a significant source for bioinspiration pertaining to the development of autonomous navigation systems.

## 1.3 Objective

A review of current literature in the area of bat bioinspiration, as it pertains to autonomous navigation, has shown that there is significant potential in such applications. However, the understanding of the underlying physical phenomena which enable bats to achieve these tasks is essential to developing capable solutions. The biosonar system of bats contains a large number of unique adaptations and features that can and do contribute to their abilities. The following subsections provide the background for our area of interest, which lies within understanding the effects of the dynamics of the deformation of the external auditory periphery of Horseshoe bats.

### 1.3.1 Specialized Auditory System of Horseshoe Bats

Horseshoe bats have long been a focus of research on echolocation due to their ability to navigate through densely cluttered environments, and their success in capturing prey in such environments using echolocation [26, 45]. Horseshoe bats, through evolution, have developed many specific adaptations, one of which is their use of Constant Frequency - Frequency Modulated (CF-FM) biosonar pulses [46, 47]. Their emitted pulses contain a long constant frequency (CF) or 'pure tone' section followed by a short downward frequency-modulated (FM) section 'chirp'. To supplement their unique pulse design, their hearing is tuned to those frequencies which constitute their pulses and their hearing is increasingly sensitive for the frequencies within the CF part of their signals [48].

These bats can detect the wingbeat of prey insects from small changes in the amplitude and Doppler shifts in the echoes due to their highly sensitive hearing in a narrow frequency range [49]. This increased sensitivity in this frequency range is enabled by their unique auditory periphery physiology, namely the cochlea. The horseshoe bat cochlea is highly sensitive around the frequency range where the CF portion of their calls lies. The cochlea of these bats contains a very high concentration of inner hair cells at the locations that respond to the CF frequencies of their calls. This region of increased sensitivity to the basilar membrane in the cochlea is termed as acoustic fovea [48, 50]. In addition to this increased sensitivity, Horseshoe bats also demonstrate tight control over their call frequencies to ensure that the received echoes lie within this foveal frequency range. Bats lower the frequency of the CF component of their emission pulses to adjust for the Doppler shifts that occur due to their flight speed [51]. This behavior is called Doppler Shift Compensation (DSC) and it keeps the frequency of the received echoes in the frequency range where their hearing is most sensitive, and they are therefore able to extract maximum information about their environment. The study also showed that bats use a control system where they compare the

incoming frequency of the echoes with a reference value of the system and adjust outgoing frequencies accordingly.

### Dynamics of Auditory Periphery

One conspicuous feature of the biosonar systems of horseshoe bats is the dynamic deformations of the reception and emission interfaces during echolocation. Horseshoe bats vary the orientation and shapes of their pinna [52, 53, 54] and noseleaf [55, 56] during receptions and emissions of biosonar signals respectively. Horseshoe bats have been found to show multiple different types of pinna movement during echo emission and reception, such as rapid alternating pinna movements of the two sides when these bats were presented a target closeby in a study [53]. It was also found that ear motions were strongly correlated in time with the reception of the echoes from the target, indicating potential importance to echolocation [53]. Studies focusing on the dynamics of emissions in the biosonar systems of the horseshoe bats found that different components of the noseleaf structures (the anterior leaf [55] and the lancet [56]) moved during emission, and these movements were correlated in time, with the emissions. These motions of the noseleaf were also found to be significant and on the scale of the noseleaf itself and the wavelength of the emissions [55, 56].

Research has also investigated the effects of these dynamics of the external auditory periphery of horseshoe bats. Numerical models of the noseleaf dynamics have suggested that such features of a biosonar system can introduce time-variant effects into the emitted beampatterns [56]. However, it was also found that the beam patterns corresponding to the simulated dynamic deformations did not suggest any systematic behavior to the changes with frequency. A numerical analysis of the changes in the beam patterns on the reception side found that with the deformation, the sensitivity systematically changes between the main lobe and the side lobe for different deformation stages of the pinna [54]. Lab-based

characterizations of biomimetic sonarheads have also confirmed the predictions of numerical experiments [57, 58]. A simplified biomimetic reproduction was developed to experimentally replicate the effects of the anterior leaf dynamics of the noseleaf in horseshoe bats, and it was found that the changes in shape introduced time-variant characteristics in the output beampatters [57]. This study showed that the output beams showed a variation with frequency, direction, and time [57]. Researchers have also sought to replicate experimentally the dynamics of the reception at the pinna. A basic prototype of the pinna geometry was developed, which was further augmented by small changes to add local features. They simulated mechanically the deformations in discrete steps in the prototype and collected received biosonar signals and then calculated the beampatterns for different deformation stages. They found occurrences of frequency-dependent sidelobes with deformation which matched the mathematical predictions from prior studies [58].

Building on these prior findings, more recent analyses have demonstrated that the time-variant effects can enhance sensory-coding capacity and the ability to determine the direction of a sound source [59, 60]. Researchers experimentally replicated the fast ear motions of the horseshoe bat pinna, and recorded CF signals emitted from different azimuth and elevation angles with respect to the receiver [60]. Their results demonstrated direction-dependent Doppler shifts in the received signals, and this finding suggested that the Doppler shifts due to this pinna motion can potentially aid in direction finding [60]. Most recently, experimental lab-based studies have sought to analyze the effects of these dynamics within echoes, instead of direct emissions. Using a biomimetic sonarhead which included baffle interfaces designed to mimic the pinna and noseleaf of horseshoe bats, researchers generated emissions and recorded echoes from targets with simple as well as complex geometries, such as spheres, cubes, and artificial plants [61]. The results from this study showed the presence of these time-variant signatures which enabled them to identify echoes based on different

conformation stages of the pinna deformation [61].

## 1.3.2   Foliage Echoes

A major proportion of research on the dynamics of the external auditory periphery have focused on either numerical analyses, or studied system properties of the reception or emission systems in isolation [54, 55, 56, 57, 58]. These numerical or laboratory-based studies did not incorporate the full complexity of natural environments. However, a majority of the echoes forest-dwelling bats receive are from foliage [62, 63]. Due to the random nature of foliage echoes [64], they are often considered as the "clutter" that masks useful target echoes. This assumption, while sensible when it comes to specific applications in technical sonar, or from the perspective of just object recognition in the presence of background, is not accurate as it pertains to bat biosonar. Bats can potentially gather useful information from these foliage echoes about their location in the forest, and their proximity to landmarks such as their roosts. Other examples of critical navigation tasks that have to rely on foliage clutter are finding passageways through foliage during flight. A number of bats hunt in the forests and they require the ability to find moving targets in clutter, as well as maintaining their distance from obstacles in their surroundings [26]. Hence, the current work focused on assessing the potential effects of peripheral dynamics on echoes from the natural foliage.

## 1.3.3   Neuromorphic Representation of Echoes

Studies of the dynamics' of biosonar have often used echo waveforms recorded at high sampling rates [60] or their envelopes [59, 61] in their analyses. These signal representations are inherently different from the neural codes that represent the biosonar signals in the bats' brains. The accepted understanding of auditory processing in bats of biosonar signals sug-

gests that the incoming pressure wave stimuli are converted into vibrations within the cochlea which leads to the vibration of the basilar membrane [65, 66]. The cochlea operates akin to a filterbank, where different locations along the basilar membrane respond to different frequencies of the stimulus, and the outputs of these filterbanks form spectrogram-like representations [67]. The vibrations at the basilar membrane are converted into electrical signals at the hair cells which form the site of mechanical to electrical transduction of auditory signals [65]. These parallel electrical signals for different frequency channels get converted into neural signals or spike trains at the auditory nerve, which forms the representation of these biosonar signals within bat brains [65]. Due to the sparse nature of neural codes [68], neural signals could be computationally inexpensive to process as compared to the densely sampled acoustic recordings. Hence, the current work has included biomimetic spike representations generated based on the unique auditory processing system of horseshoe bats [50, 69, 70] in its analysis of the impact of the peripheral dynamics in horseshoe bats.

### 1.3.4  Research Approach

The objective of this research is to study the effects of the dynamics of the auditory periphery in horseshoe bats in the context of foliage echoes. The second objective of the research is to implement a bioinspired neuromorphic representation of the echoes, to determine whether these effects persist in the sparse biological representation. Answering these questions would enable us to know the potential relevance of these effects in horseshoe bats, based on scenarios that closely represent their habitats, and signal representations similar to those within bats' brains than what has been studied in previous work.

To facilitate this analysis, we collected a large echo dataset from natural foliages using a recording system that mimicked the biosonar system of horseshoe bats in pulse design and dynamics. These recorded echoes were then processed with a signal-processing model

that replicates some specific features of the auditory processing in horseshoe bats including the generation of neural spikes. Finally, we applied machine-learning based classification techniques to these biomimetic spike responses to classify them based on dynamics' effects. If these effects can be accurately identified within the echoes, this indicates that natural foliage echoes generated and received by a dynamic periphery have a distinct quality.

# Chapter 2

# Biomimetic Detection of Dynamic Signatures in Foliage Echoes

## 2.1 Title

Biomimetic Detection of Dynamic Signatures in Foliage Echoes

## 2.2 Abstract

Certain bat species (family Rhinolophidae) dynamically deform their emission baffles (nose-leaves) and reception baffles (pinnae) during echolocation. Prior research using numerical models, laboratory characterizations and experiments with simple targets have suggested that this dynamics may manifest itself in time-variant echo signatures. Since the pronounced random nature of echoes from natural targets such as foliage has not been reflected in these experiments, we have collected a large number (>55,000) of foliage echoes outdoors with a sonarhead that mimics the dynamic periphery in bats. The echo data was processed with a custom auditory processing model to create spike-based echo representations. Deep-learning classifiers were able to estimate the dynamic state of the periphery, i.e., static or dynamic, based on single echoes with accuracies of up to 80%. This suggests that effects of the periph-

eral dynamics are present in the bat brains and could hence be used by the animals. The best classification performances were obtained for data that was obtained within a spatially confined area. Hence, if the bat brains suffer from the same generalization issues, they would have to have a way to adapt their neural echo processing to such local fluctuations to exploit the dynamic effects successfully.

## 2.3   Introduction

Developing autonomous systems, e.g., robots or drones, capable of navigating through complex environments presents a multitude of application opportunities in agriculture, autonomous deliveries, and surveillance [71]. Current autonomous navigation solutions primarily make use of cameras, lidar, and infrared sensors to gain information about their environment [72, 73]. Processing the image data produced by these sensors in real-time leads to high computational loads and power consumption. In addition, natural outdoor environments pose challenges for optical methods that include changes in ambient light conditions, weather, and occlusions. These issues have led engineers to consider optical solutions as sub-optimal for small drones [74]. Hence, different sensing solutions along with faster, computationally inexpensive data processing techniques that are more suited to such application environments remains of critical interest. Many echolocating bat species effectively navigate through complex environments, such as dense forest vegetation, using biosonar [75]. In doing so, they capably execute complicated tasks that engineers aspire to accomplish through autonomous robots, such as effectively navigating through small gaps between an array of strings in a complex indoor obstacle course, and outdoors bats were able to successfully forage for prey in cluttered forest environments [76]. Bats have also been shown to successfully identify landmarks and leverage them for navigation through openings in nets

in behavioral experiments [27]. These abilities exceed what is achievable with cutting-edge autonomous navigation technology, where obstacle avoidance autonomously and developing paths dynamically remains a challenge [77, 78, 79] and hence make learning from bat biosonar an attractive proposition for meeting these challenges posed by autonomous navigation in natural environments.

Bat species possess highly sophisticated biosonar systems that are presumed to contain evolutionary adaptations to the animals' respective environments. The biosonar systems of horseshoe bats (family Rhinolophidae) have several conspicuous features that could be hypothesized to be adaptations for navigating and hunting in very dense vegetation. Among these features is the ability of the bats to modulate the shape of their noseleaves [56] and pinnae [54] while emitting and receiving biosonar signals. Numerical models have suggested that such dynamic features of a biosonar system can introduce time-variant effects into the emitted and received signals [56, 80]. Lab-based characterizations of biomimetic sonarheads have confirmed these predictions [57, 58] and further analysis based on these data has demonstrated that the time-variant effects can enhance sensory-coding capacity and the ability to determine the direction of a sound source [59, 60]. Most recently, experimental lab-based studies with a biomimetic sonarhead have also found these time-variant effects in echoes from targets with simple as well as complex geometries, such as spheres, cubes, and artificial plants [61].

Whereas these prior studies have presented significant findings on the potential impact of the peripheral dynamics, they have often relied on numerical models or device characterizations performed in laboratory settings that did not capture the full complexity of natural environments. In contrast, the predominant sensory inputs in forest-dwelling bats consists of echoes from foliage [62, 63]. Because of their random nature [64], foliage echoes are frequently regarded as "clutter" that masks useful target echoes, but these clutter echoes convey

all information that bat biosonar can gather about the environment. Examples of critical navigation tasks that have to rely on foliage clutter are identification of natural landmarks and finding passageways in foliage. Nevertheless, few existing studies have focused on clutter echoes [64, 81, 82, 83]. Hence, the current work is dedicated to assess the potential effects of peripheral dynamics on echoes from the natural foliage.

Another limitation of previous studies results from the use of densely-sampled echo waveforms [60] or their envelopes [59, 61]. These signal representations are inherently different from the neural codes that represent the biosonar signals in the bats' brain. In the auditory system of bats, the signals are first separated into frequency bands at the basilar membrane [50, 69, 70] and then represented by a sparse code consisting of a few spikes for each frequency band. Due to their sparse nature, neural signals could be computationally inexpensive to process when compared to the dense sampling of acoustic recordings. Hence, the current work has included biomimetic spike representations in its analysis of the impact of the peripheral dynamics in horseshoe bats.

The specific goals of the current research have been to establish whether (i) the effects of the peripheral biosonar dynamics are manifest in foliage echoes collected in natural environments and (ii) whether these effects persist in sparse neuromorphic representations of the echoes. Answering these two questions would allow assessing the potential relevance of the peripheral biosonar dynamics in horseshoe bats based scenarios that are much closer to the nature of the habitats and brains of bats than what has been studied in previous work.

To facilitate this analysis, we have collected a large echo dataset from natural foliages using a recording system that mimicked the biosonar system of horseshoe bats in terms of pulse design and peripheral dynamics. The echoes were then processed with a signal-processing model that replicates some of the specific features of the auditory processing in horseshoe bats including the generation of neural spikes. Finally, we have applied analytical and machine-

learning based classification techniques to these biomimetic spike responses to classify them on the basis of peripheral dynamics. If the dynamics state of the biomimetic sonar can be accurately predicted from the echoes, this means that natural foliage echoes generated and received by a dynamic periphery have a distinct quality.

## 2.4  Methodology

### 2.4.1  Biomimetic Recording Hardware

A sonarhead designed to mimic the biosonar system of horseshoe bats (figure 2.1) was used for the data collection in the experiments. This system included an emitter and two receivers to generate and record ultrasonic signals respectively. The interface baffles for sound emission and reception were modelled after the geometry of the noseleaves and pinnae in greater horseshoe bats (*Rhinolophus ferrumequinum*). Noseleaves and pinnae were actuated to replicate the dynamics in the biosonar periphery of horseshoe bats.

The emission baffle was designed to closely mimic the anatomy of the noseleaf of the greater horseshoe bat and included two openings or "nostrils" (3 mm in diameter each) for sound emission. The receiver baffles were designed based on µCT scans of the pinnae of horseshoe bats. CT scans were used as reference to design 3d computer models of the baffles in mesh software (Blender, Blender Foundation, Amsterdam, Netherlands). The designs were 3d-printed with polylactic acid (PLA) to create a rigid model to serve as a template for making a mould. The moulds were made in silicone, and were then used to cast a flexible model in a silicone material for the baffles (Dragonskin 30, Smooth-on, Macungie, Pennsylvania, United States). Two electrostatic ultrasonic transducers (600 series, diameter 38.4 mm, SensComp, Livonia, Michigan) generated the ultrasonic pulses. The peak response for these transducers
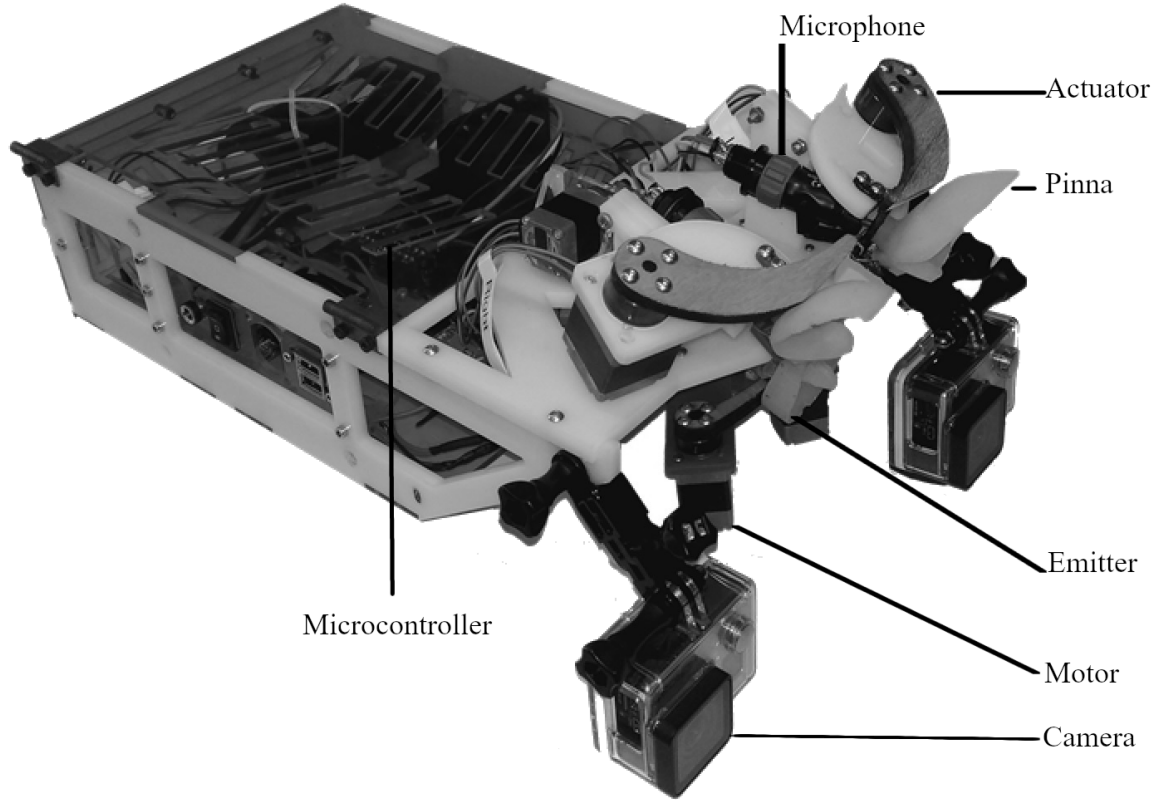
Figure 2.1: Biomimetic sonarhead that houses ultrasonic speakers for signal emission, microphones for reception, and an actuation system to mimic the peripheral dynamics of horseshoe bats.

occurred at a frequency of around 55 kHz and the -6 dB bandwidth ranged from 45 to 75 kHz approximately. One transducer each was connected to the two "nostril" openings in the noseleaf via a dedicated conical waveguide that was 7.6 cm in length. Two capacitive MEMS microphones (Monomic, Dodotronic, Rome, Italy) served as receivers. Each microphone was connected to the respective pinna via an artificial "ear canal" (diameter 4.5 mm, length 2 mm) that was inserted into the back surface of the receiver baffle near its base.

The peripheral baffle shapes were deformed by virtue of levers mounted on the shafts of stepper motors (PKP Series, Oriental Motor, Tokyo, Japan). Rotation of the stepper motor shaft causes the levers to push against the back surfaces of noseleaf and pinna baffles and

deform these shapes into a bend state. The motors were controlled via a microcontroller (Arduino MEGA, Arduino, Ivrea, Italy) with stepper-motor drivers (Easy Driver, Sparkfun, Niwot, Colorado, United States). A total of five motors were used, three to actuate the noseleaf (for the lancet and both sides of the anterior leaf parts of the noseleaf) and one for each pinna. For controlling emission and reception, a second microcontroller (Arduino DUE, Arduino, Ivrea, Italy) was used. This microcontroller included an onboard 14 bit analog-to-digital converter (ADC) which was used to sample the analog input from the microphones at $400\,^{kS}/_{s}$, and a 12 bit digital-to-analog (DAC) converter to output analog signals to the speakers. Through a serial (Universal Asynchronous Reception and Transmission – UART) connection with a computer, these sample signals were transferred for storage on the computer.

To generate echoes that closely resembled the natural input to the auditory system of horseshoe bats, the biomimetic sonarhead was set up to emit constant frequency - frequency modulated pulses (CF-FM, figure 2.2) that were designed based on emissions recorded from greater horseshoe bats [84]. Since the sonarhead's noseleaf and pinnae were about twice the size of the respective structures in the bats, the pulses were scaled down in frequency such that the CF component was at 40 kHz (instead of about 83 kHz in the bats). Similarly, the FM component was scaled to sweep down from 40 to 25 kHz (instead of from 83 to approximately 67 kHz in the bats [84]). As a result of this frequency scaling, the emission band came closer to the emitter's peak response frequency. The flanks of the pulse were smoothed by multiplying the entire pulse with a Hann window of the same length as the pulse.
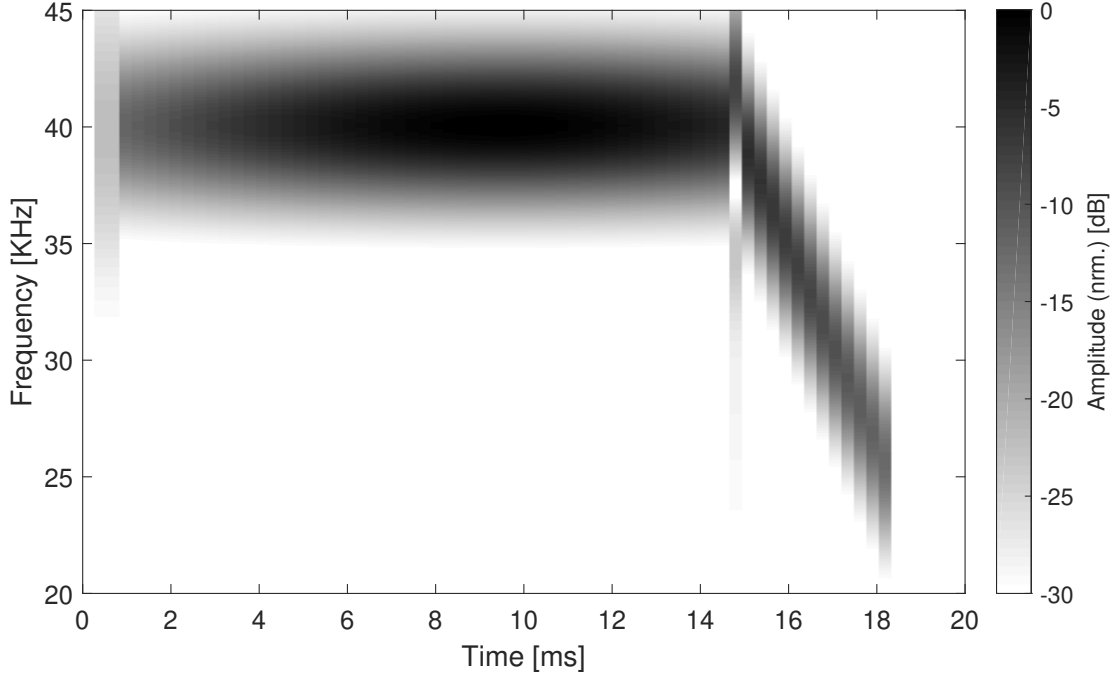
Figure 2.2: Spectrogram of a simulated constant frequency-frequency modulated (CF-FM) horseshoe bat biosonar call. It consisted of a constant-frequency (CF) component at $40\,\mathrm{kHz}$ with a duration of $12\,\mathrm{ms}$, followed by a frequency-modulated (FM) component that swept from 40 to $25\,\mathrm{kHz}$ over a duration of $3\,\mathrm{ms}$. The spectrogram was computed with a 128-point Hann window 128 and 25% overlap.

## 2.4.2  Dataset

Echoes from natural foliage were recorded along 10 tracks that were distributed over four sites on the Virginia Tech campus (Stadium Woods, Grove Lane, Inventive Lane, and the Corporate Research Center, figure 2.3). For data collection, the biomimetic sonarhead was hand-carried and oriented to face nearby foliage. A minimum distance of approximately $1\,\mathrm{m}$ was maintained between the target foliage and the sonarhead. The operator swept the sonarhead slowly vertically and horizontally within a range of approximately $\pm30°$to scan the surrounding foliage. This was repeated when the operator stepped to the next position. During the recording sessions at any of the field sites, echoes with static and dynamic states

of the sonarhead's periphery were acquired in alternating blocks of 15 echoes each creating a 50/50 split between static and dynamic state echoes. Each acquired echo was automatically labeled as "static" or "dynamic" based on the sonarhead's state during data collection to provide a basis for supervised learning of these echoes. Echoes were recorded at a rate of three per second. Based on the recorded GPS data, this corresponded to about 21 echoes per meter traversed on average.
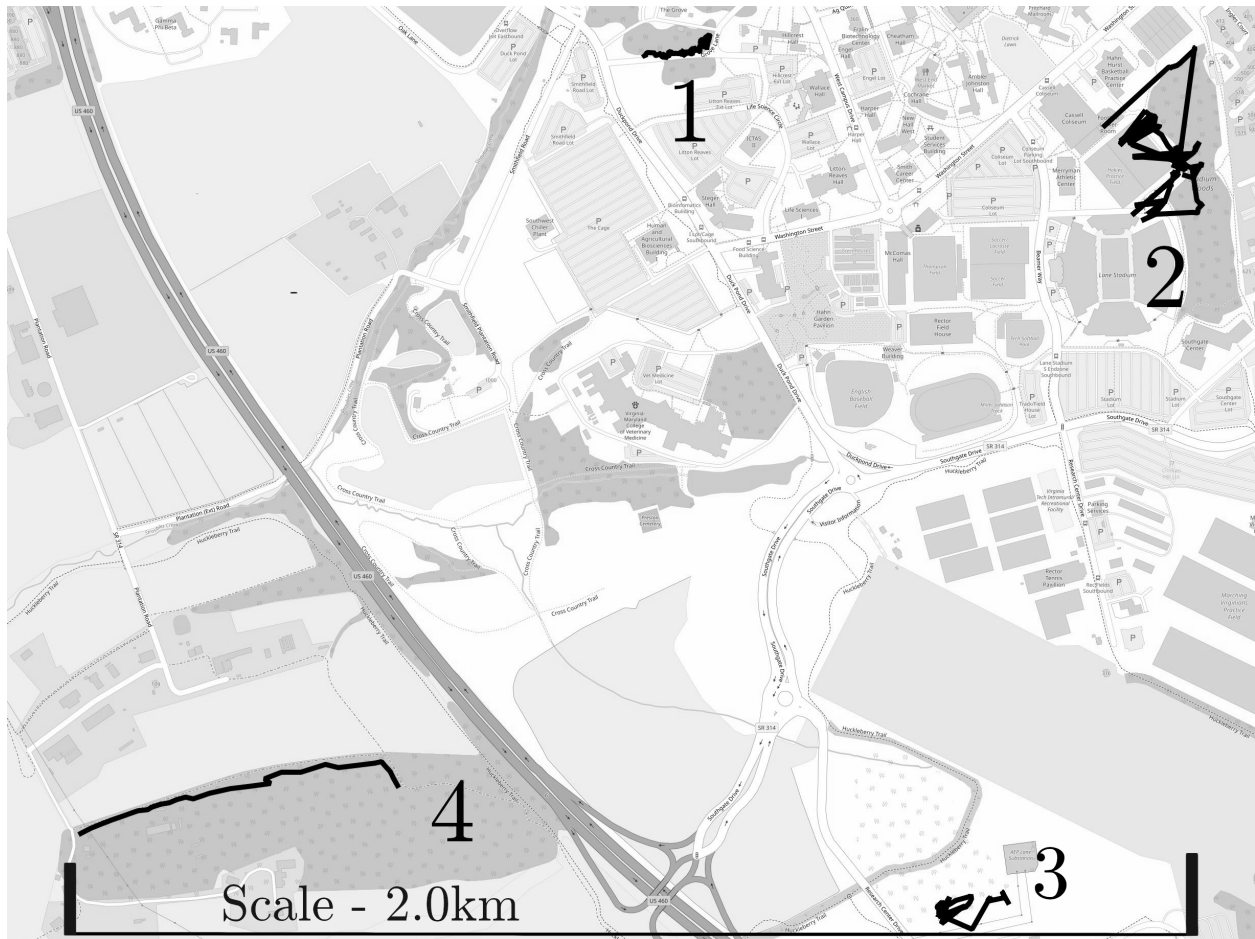


Figure 2.3: Sites of foliage echo collection. The four locations are numbered, and sample data collection tracks tracked with GPS are highlighted in black. The locations were: 1) Grove Lane, 2) Stadium Woods, 3) Corporate Research Center, 4) Inventive Lane. Map data ©OpenStreetMap contributors [1]

Correlation coefficients between the recorded echo waveforms were calculated to assess the

similarity between recorded echoes in the dataset (figure 2.8). The correlations were calculated using 200 echoes consecutively recorded, spread over a distance of 20 m. To establish a reference for the value of correlation coefficient between echo signals due to random target impulse responses, a separate numerical experiment was conducted: Artificial foliage impulse responses representing a sequence of point reflectors with random target strength and distance were created using amplitudes drawn from a Gaussian distribution and timing based on a Poisson process. These impulse responses were convolved with the emitted pulse and then Gaussian white noise was added (SNR 40 dB) to create simulated "echoes".

The complete dataset contained 55,400 foliage echoes across all four collection sites (figure 2.3) and a total of 10 "tracks" which contained echoes from one continuous data collection session within a site. Classification of echoes into "static" and "dynamic" was attempted based on the entire dataset, all echoes recorded along an individual track containing between 2,520 and 9,660 echoes, and a subset of 420 echoes that were recorded within a 20 m segment of each track. The more restricted sub-data sets, i.e., for tracks and track segments, were tested in order to reduce variations due to location or weather condition at the time of recording. Regardless of their size, all datasets used in this study had a 50/50 split of dynamic and static recording conditions.

For a more detailed analysis of the relationship between dataset size on classification accuracy, the three tracks with the largest number of echoes (Tracks 3, 5, and 7) were selected. The echoes corresponding to each track were used to construct smaller datasets of varying echo counts ranging from 420 to 7,980 in steps of 420 (figure 2.10). The datasets were split without prior shuffling or reordering, so the recording order of the echoes was maintained.

The same three largest tracks were selected to create the datasets used to investigate the relationship between physical recording location and classification accuracy. In this case, the echo data collected along the three tracks was broken up into segments containing 320

echoes that were consecutively recorded over a track distance of approximately 20 m. The echoes in these segments were used to form a training dataset. A physical location along the respective track was attached to each of these training data sets based on the echo in the center the segment. Classifier networks were trained on a 320-echo dataset centered at a given location were then tested with 10 different test datasets. The spatial centers of these test datasets were spaced from -23 m to 22 m relative to the center of the training data sets in increments of 5 m. The datasets were created such that there was no overlap between the training and testing datasets, hence test sets did not contain any echoes which were used in the training set. With the large number of echoes in each track, the training and corresponding testing datasets were created for multiple positions (16 positions), which were used for repeat classification experiments.

## 2.4.3  Signal Processing

Each recorded ultrasound signal can be divided into three portions based on its content (figure 2.4): (i) pure direct noseleaf-to-pinna transmission of the pulse, (ii) direct transmission overlapping with returning echoes, (iii) returning echoes only. All analyses carried out here were based on the third portion of the recorded signals to ensure that any effects found were mediated by the echoes and not just directly passed between noseleaf and pinna.
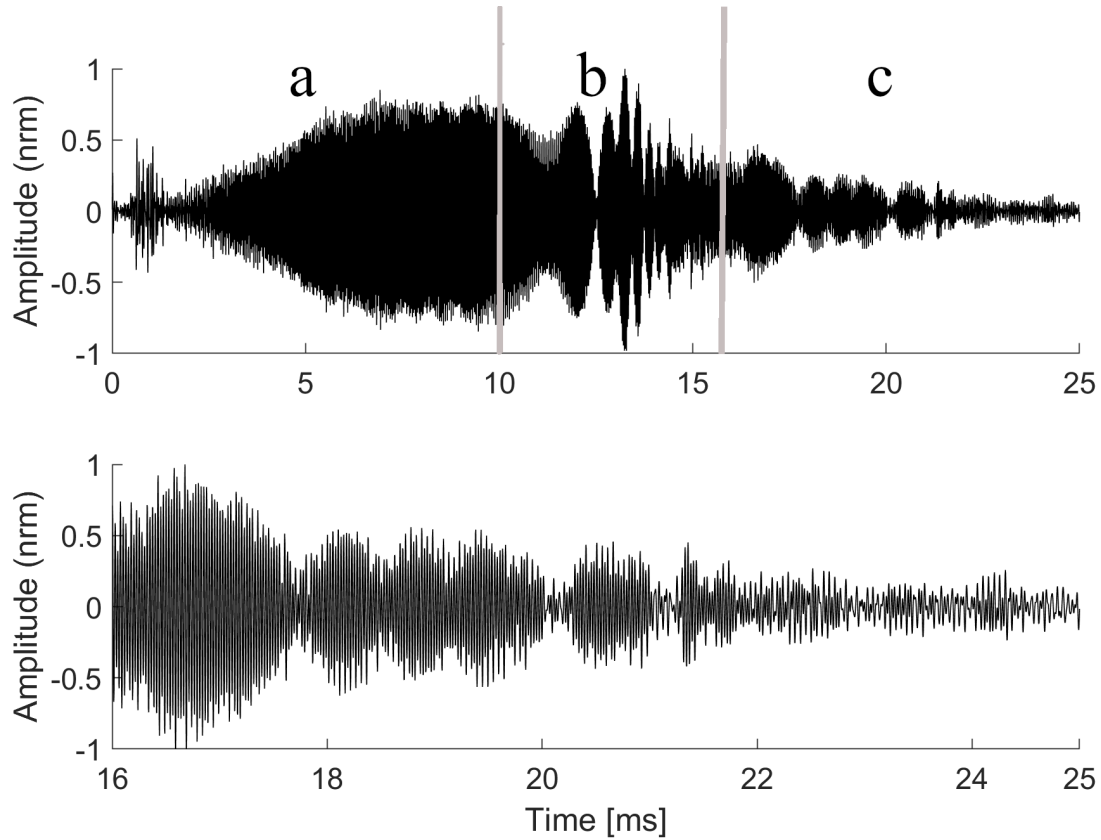
Figure 2.4: The recorded signal sections consist of a) direct recording of emitted signal, b) overlap of emission and echo from foliage c) only echo.

The echo signals were filtered to attenuate out-of-band noise and interference ($10^{\text{th}}$-order Butterworth bandpass design, passband between 20 and 45 kHz). After filtering, the signals were normalized by their respective root-mean-square (rms) values to eliminate any potential overall differences in signal energy.

The auditory processing model (figure 2.5) consisted of three components to represent (i) the basilar membrane, (ii) the inner hair cells, and (iii) the spiking neurons of the auditory nerve. The basilar membrane model was designed to mimic the mammalian cochlea taking into account the specifics found in greater horseshoe bats [69]: The quality of the auditory filters and their density as a function of center frequency were modelled on results from prior

physiological experiments [50, 69] and models [85]. The total number of frequency channels simulated was chosen to be 100 in order to provide adequate sampling of filter properties over frequency while limiting computational cost.
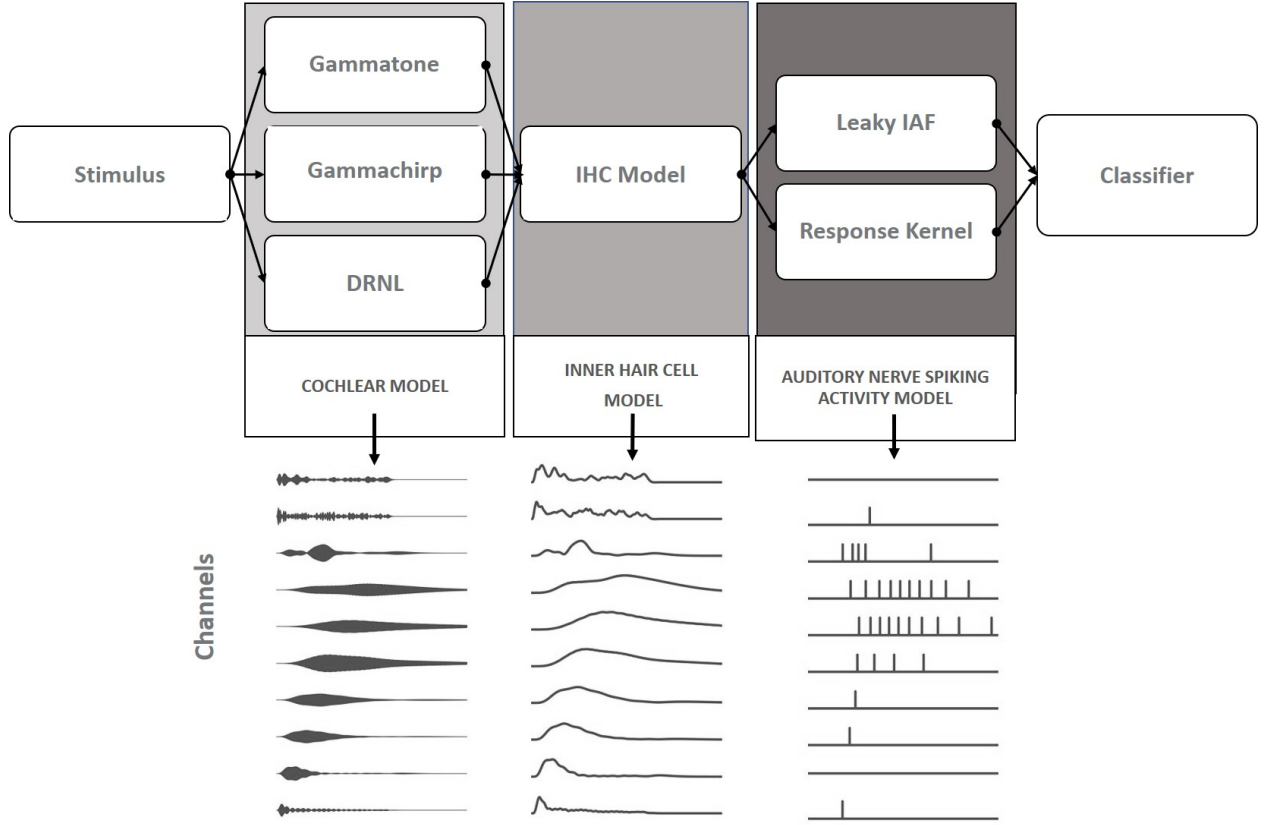


Figure 2.5: Auditory model simulates three steps basilar membrane, inner hair cell transduction, auditory nerve spike response. Sample outputs at each level are represented.

Three different basilar membrane models were tested: (i) gammatone, (ii) gammachirp, and (iii) Dual-Resonance-Non-Linear (DRNL) model. The gammatone filterbank [86] is a linear model with symmetric bandpass transfer functions. The impulse response of a gammatone filter is given by:

$$g_{GT}(t) = a t^{(n-1)} \cdot e^{-2\pi bt} \cdot \cos\left(2\pi f_c t\right), \tag{2.1}$$

where $n$, $b$, $f_c$, and $a$ represent the filter order (here set to 4 [86]), bandwidth, center

frequency, and normalization factor of the filter respectively.

The gammachirp filterbank is a modified version of the gammatone filterbank that mimics the asymmetric transfer functions of the basilar membrane [87]. The impulse response of the gammachirp is defined as:

$$g_{GC}(t) = at^{(n-1)} \cdot e^{-2\pi bt} \cdot \cos(2\pi f_c t + c\ln(t)). \tag{2.2}$$

The term $c\ln(t)$ adds a frequency modulation to the gammatone carrier. Here, a value of $-3$ was utilized for $c$.

The Dual-Resonance-Non-Linear (DRNL) basilar membrane model adds a nonlinear level-dependent behavior to the auditory filter model [88, 89]. It models the effects of input sound pressure level on the best frequency (BF) and bandwidth (BW) of the filters. The DRNL model was developed to simulate human psychophysical data [89] and consists of two parallel pathways, one linear and the other nonlinear, that are added to generate the final model output. The linear pathway consists of a scalar gain factor, a cascade of two identical gammatone filters followed by a cascade of four Butterworth lowpass filters [89]. All filters in the linear path are controlled by two parameters, the center frequency ($CF_{lin}$) and bandwidth ($BW_{lin}$), which were estimated from experimental bat neurophysiology data [50, 69, 85] in the present work. The nonlinear pathway is a cascade of three identical first-order gammatone filters with narrowly tuned bandwidths, a memoryless compressive function, another three narrowly tuned identical first-order gammatone filters, and three first-order Butterworth low-pass filters [89]. Like the linear path, the nonlinear path was also parameterized by center frequency ($CF_{nl}$) and bandwidth ($BW_{nl}$), that were applied to all the filters in the nonlinear pathway. These parameters, ($CF_{nl}$) and bandwidth ($BW_{nl}$) were defined by a relationship with the corresponding parameters in the linear path, which was determined to

fit the BM performance to the experimental psychoacoustic data in humans [89], and these same values are used here in the absence of existing bat physiological data. The compression in the nonlinear pathway can be described by:

$$y(t) = \text{sign}(x(t)) \cdot \min(a \cdot \mid x(t) \mid, b \cdot \mid x(t) \mid^v), \tag{2.3}$$

where $a$, $b$, and $v$ are parameters specific to each particular basilar membrane location. Parameter $a$ describes the "sensitivity" at the tip of the filter, and a value of $5 \times 10^{-5}$ was used, as described in [89]. The parameter $b$ represents a gain determined by fitting the model output to experimental data for different basilar membrane locations, and hence different channel frequencies utilized different values [89]. Similarly, the compression coefficient $v=0.25$ was found by a fit to experimental human data [89] The outputs after each stage of the auditory model were saved and visualized for validation of the model performance (figure 2.6). The FM channels of the filterbanks had filters with a higher bandwidth and showed greater temporal detail compared to the high quality factor CF channels (figure 2.6a).
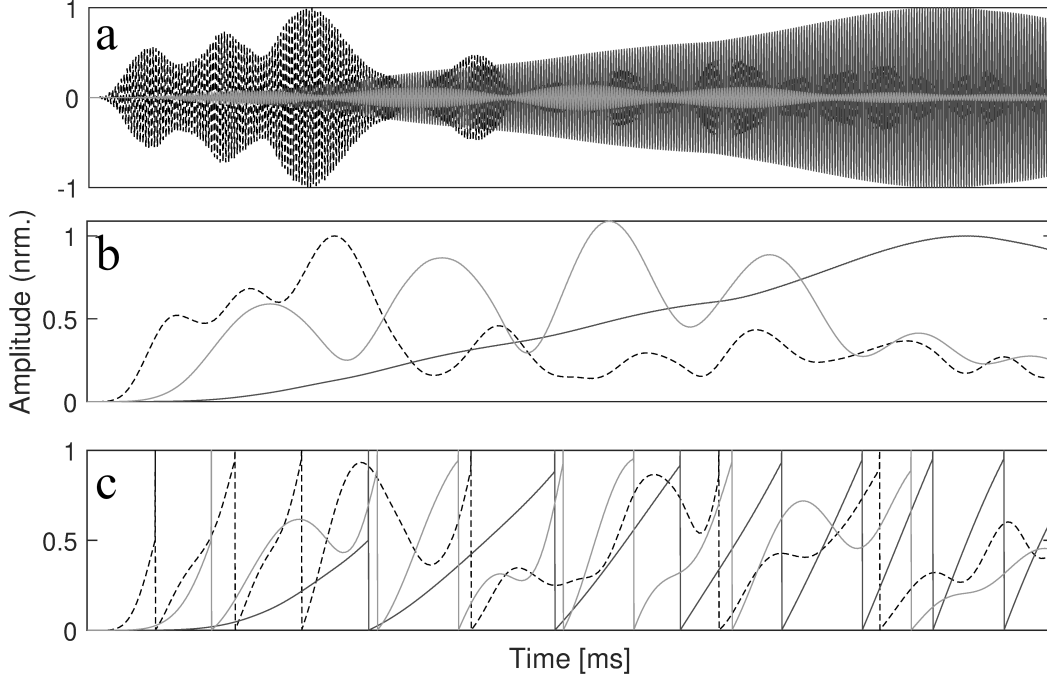
Figure 2.6: Intermediate model outputs for sample foliage echo: a) basilar membrane response, b) inner hair cell current c) membrane potential of an auditory nerve neuron. Each line represents the response to a different frequency: low (35 kHz, dotted, medium (40 kHz, solid black), and high (45 kHz, solid gray).

The signal transduction between the inner hair cells and the neurons of the auditory nerve can be modelled as a half-wave rectifier followed by a lowpass filter [90]. Here, a $5^{th}$-order Butterworth filter with a cutoff frequency of 1 kHz was used for this step. The inner hair cell model outputs preserve the low frequency content of the basilar membrane outputs, are visually similar to the signal envelopes and show the amplitude variation (figure 2.6b). The inner hair cell outputs were converted to spike responses using two spike-generation models that each represented a different level of complexity: (i) the simpler leaky integrate-and-fire model (LIAF, [91]) and (ii) the more complex response kernel model [91] that includes

refractory neural properties. The LIAF model describes the neuronal dynamics with the following differential equation for the membrane potential:

$$\tau_{\mathrm{m}} \frac{du}{dt} = -[u(t) - u_{\mathrm{rest}}] + RI(t), \tag{2.4}$$

where $u(t)$ is the membrane potential, $u_{\mathrm{rest}}$ is the resting potential, $R$ is the membrane resistance, $I(t)$ is the input current, and $\tau_{\mathrm{m}}$ the membrane time constant of the neuron. The solution of equation (2.4) with the initial condition $u(0) = u_{\mathrm{rest}}$, yields equation 2.5, which describes the time history of the membrane potential in the LIAF model:

$$u(t) = u_{\mathrm{rest}} + RI(t)\left[1 - \exp\left(-\frac{t}{\tau_{\mathrm{m}}}\right)\right]. \tag{2.5}$$

Equation 2.5 is combined with a spiking mechanism, and subject to the condition that $u(t)$ is changed to the resting potential $u_{\mathrm{rest}}$ when it reaches the fixed threshold $\vartheta$ to complete the LIAF model:

$$u(t) = u_{\mathrm{rest}} + \sum_{f}(u_r - \vartheta)\, \exp\left(-\frac{t - t^f}{\tau_m}\right) + \frac{R}{\tau_m}\int_0^\infty \exp\left(-\frac{s}{\tau_m}\right) I(t - s)\,\mathrm{d}s, \tag{2.6}$$

where the firing times $t^f$ are defined by the threshold condition. The second term on the right hand side of the equation represents the change in the membrane potential after the spike, due to the decrease in voltage from $\vartheta$ to $u_r$. The third term models the impulse response of the membrane potential due to a time-varying input current.

The generalized Spike Response Model (SRM) [91] equation is a representation of Equation 2.6 in the kernel form, and is given by:

$$u(t) = \eta(t - \hat{t}) + \int_0^\infty \kappa(t - \hat{t}, s)I(t - s)ds, \tag{2.7}$$

where $\hat{t}$ denotes the firing time of the last spike of the neuron. The Response Kernel (RK) model used here is a special case of the general SRM equation. It incorporates refractoriness in a neuron in three ways. Firstly, through the reduction in the responsiveness after a spike of the impulse response of the neuron. This is achieved by addition of a time dependence on prior spike occurrence and a recovery time constant in the impulse response kernel, the $\kappa$-kernel (Eq. 2.8), which describes the time course of the voltage response to a short current pulse $I(t)$

$$\kappa(t - \hat{t}, s) = \frac{R}{\tau_\mathrm{m}}\left[1 - \exp\left(-\frac{t - \hat{t}}{\tau_\mathrm{rec}}\right)\right]\exp\left(-\frac{s}{\tau_\mathrm{m}}\right)\Theta(s)\Theta(t - \hat{t} - s), \quad (2.8)$$

where $\Theta(s)$ denotes the Heaviside step function and $\tau_\mathrm{rec}$ is the response recovery time constant.

Secondly, refractoriness is included in the $\eta$-kernel ($\eta(t - \hat{t})$, Eq. 2.9), in the form of the the negative overshoot, $u_\mathrm{reset}$ which typically follows a spike (the spike afterpotential).

$$\eta(t - \hat{t}) = u_\mathrm{rest} - \eta_0 \exp\left(-\frac{t - \hat{t}}{\tau_\mathrm{ref}}\right)\Theta(t - \hat{t}), \quad (2.9)$$

where $\eta_0 = u_\mathrm{rest} - u_\mathrm{reset}$, $\Theta(x)$ denotes the Heaviside step function, and $\tau_\mathrm{ref}$ is the refractory time constant. Finally, the third way to include refractoriness is changing the threshold from a constant voltage as in the LIAF model to a function $\vartheta(t - \hat{t})$ that depends on the time $\hat{t}$ since the previous spike. The threshold mimics the neuron's refractoriness by increasing immediately after a spike and decaying back to its resting value $\theta_0$ to restore the responsiveness of the neuron. The dynamic threshold can be expressed as follows:

$$\vartheta(t - \hat{t}) = \vartheta_0 - \eta(t - \hat{t}). \quad (2.10)$$

The spike responses generated from the integration of the input IHC current to the spike were plotted for visual comparison of different spike models (figure 2.6c).

Since the input to the spike model was normalized and had arbitrary units, the initial values of certain parameters of the spike model were determined by trial and error. These parameters were: the spiking threshold $\vartheta$ and and the membrane time constant $\tau_{\mathrm{m}}$ for both spike models. And for the kernel model additionally the recovery $\tau_{\mathrm{rec}}$ and refractory $\tau_{\mathrm{ref}}$ time constants. The initial values were selected so the spike model resulted in a spike rate ($r$) which fell within a desired range ($1 > r > 0.5$) per ms, consistent with established physiological limits [92]. Lower and upper limits for $\vartheta$ and $\tau_{\mathrm{m}}$ were first established by changing one parameter while keeping the other constant till the lower and upper spike rate was reached, respectively. For the kernel model, the two other time constants, $\tau_{\mathrm{rec}}$ and $\tau_{\mathrm{ref}}$ were limited within a range ($0.8*\tau_{\mathrm{m}} : 1.2*\tau_{\mathrm{m}}$). Once these limits were established, parameter values were manually changed, and spike outputs were generated for different combinations. The combinations which yielded the outputs with the highest classification accuracy (see the Machine learning classification section later in this work) were selected as final parameter values.

### 2.4.4 Machine Learning based Classification

For the classification of the sonarhead state based on echo waveforms, two echo representations were used: spectrograms and spike-timing plots (figure 2.7). The spectrogram representations of the echoes were computed from the final segments of the recordings that contained only echoes and were each 9 ms (3,600 samples) in length. Each spectrogram representation (256-point Hann window spaced without overlap) contained 64 samples along the frequency dimension and 14 samples the temporal dimension. The spike-timing plot represented the

spike-model output by a matrix with dimensions ( 100×3,600) corresponding to the number of frequency channels and time samples respectively. Each entry in the matrix had a binary (0 or 1) value to represent a spike or no spike.

Studies on the precision of spike timing in the mammalian auditory system have used phase locking as a measure of temporal precision [93] and have found it to be limited to the upper frequency of 4 kHz [94, 95, 96, 97, 98]. To mimic this biological precision and also reduce the dimensions of these spike arrays, bin sizes of 0.25 ms corresponding with the 4 kHz frequency limit were selected. The high sample rate spike-train representations were reduced from 3,600 to 36 samples along the temporal dimension using a simple binning process where any number of spikes in a bin (100 samples or 0.25 ms) was represented by "1" while absence of spikes was represented by "0". The time of spike occurrences in this $100 \times 36$ vector were used to create a $100 \times 9$ vector, since a maximum of nine spikes occurred in each given frequency channel for the current echo data set.
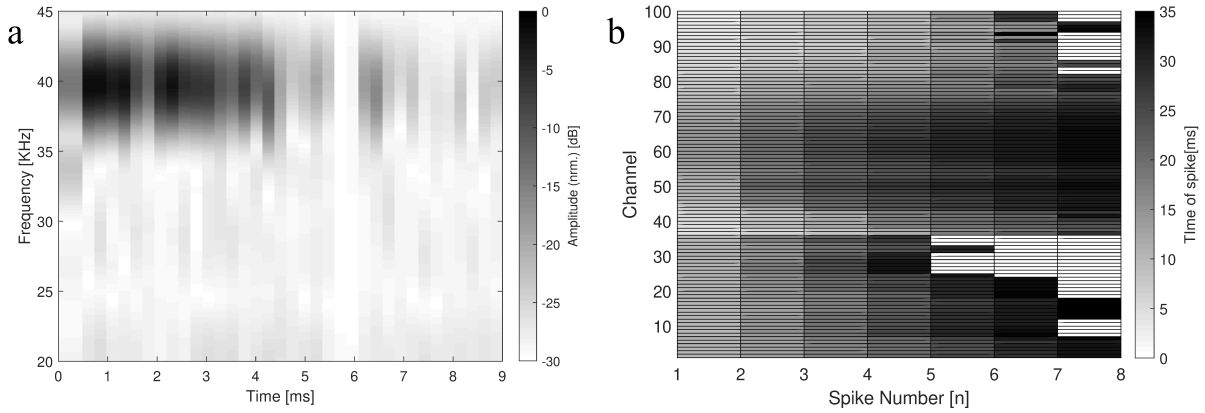


Figure 2.7: Example inputs for the Machine Learning Classifier: a) Spectrogram of Echo. b) Spike timing plot

To classify the echoes, a deep neural network that consisted of a simple linear stack of layers was created with the Keras sequential model [99] that utilizes the Tensorflow [100] core. Our network consisted of seven dense layers (including the input and output layers) and three

dropout layers. The first six dense layers incorporated rectified linear unit (ReLu) activation whereas the output layer used sigmoid activation to get predicted label values of 0 and 1 to represent "static" and "dynamic" respectively. Weight updates during network training were calculated with the Adam optimizer algorithm [101].

The classifier's input data, both spectrograms and spike-time representations, was randomly split into a training set (72.25% of the samples), a validation set (12.75%), and a hold-out testing set (15% of the samples). The classification accuracy on the validation set (i.e., "validation loss") was monitored after each training epoch and the model parameters where saved whenever it reached a new minimum value. The number of training epochs was limited to 300. However, if the validation loss increased for more than 50 successive epochs, training was stopped early to prevent over-training. Classification performance was quantified irrespective of the selected classification threshold by virtue of receiver operating characteristic (ROC) curves. The area under the curve (AUC) of the ROC curve provided a single scalar metric for the classification performance for all classification test conducted.

## 2.5  Results

The recorded foliage echoes showed little similarity as quantified by pairwise correlation coefficients (figure 2.8). The average value of the correlation coefficient (0.53) was similar to what was obtained in the numerical study that involved completely random and uncorrelated impulse responses (0.51). Furthermore, the values of the correlation coefficient did not depend on distance between the respective echoes in time or space. The mean value for correlations between all echoes of a given track (0.53, standard deviation 0.11, $N$=19,900) was not significantly different from that obtained between neighbouring echoes only (0.54, standard deviation 0.12, $N$=199, $t$-test, $p$=0.08).
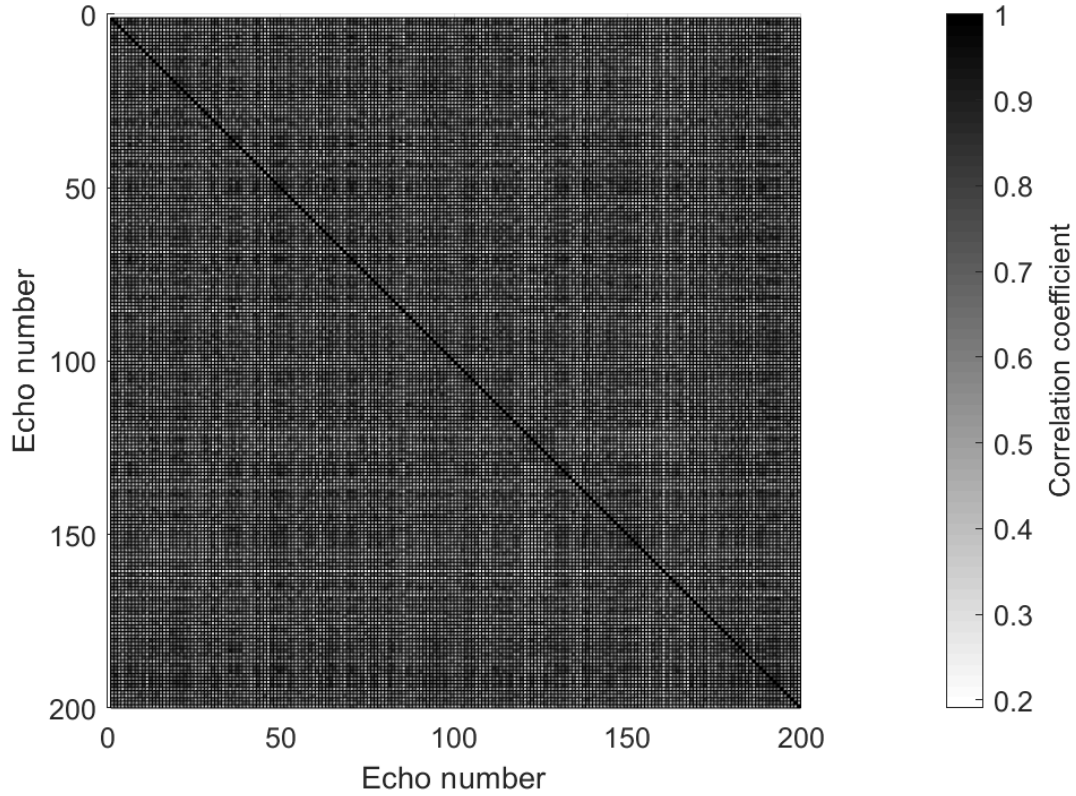
Figure 2.8: Similarity between foliage echoes quantified by pairwise correlation coefficients for track 8. Echoes with with consecutive numbers were recorded with a time gap of 300 ms. A distance of approximately 20 m was traversed while the 200 echoes were recorded.

The dynamic state of the sonarhead could be classified with both input representations and over the datasets from all tracks, however, there were differences in accuracy (figure 2.9). The classification accuracy was found to depend on dataset size, with smaller data sets yielding better results. Using the entire dataset of 55,400 echoes for classification, the AUC for the ROC curve was 0.56. For datasets limited to individual tracks (sizes ranging from the smallest containing 2920 echoes to the largest with 9660 ), the mean AUC across all datasets was 0.59 (standard deviation 0.02). For the datasets containing 420 echoes each, the average AUC was 0.79. The different echo representations had a minor impact on the accuracy for the large datasets, it had a major impact on the smaller 420-echo datasets (table 2.1). The

classification performance was found to differ significantly between the BM models for the 420-echo datasets. The mean AUC from the DRNL model (mean 0.8, standard deviation 0.15, $N$=132) compared to the GC model (mean 0.71, standard deviation 0.12, $N$=132) was higher ($t$=4.89, $p$ <0.001). The classification accuracy also varied with the location of data collection (track, figure 2.9).

Table 2.1: Classification accuracy with different echo representations.

| Echo representation | AUC (mean) |
|---|---|
| Spectrogram | 0.69 |
| Gammachirp-Spike Timing | 0.71 |
| DRNL-Spike Timing | 0.79 |
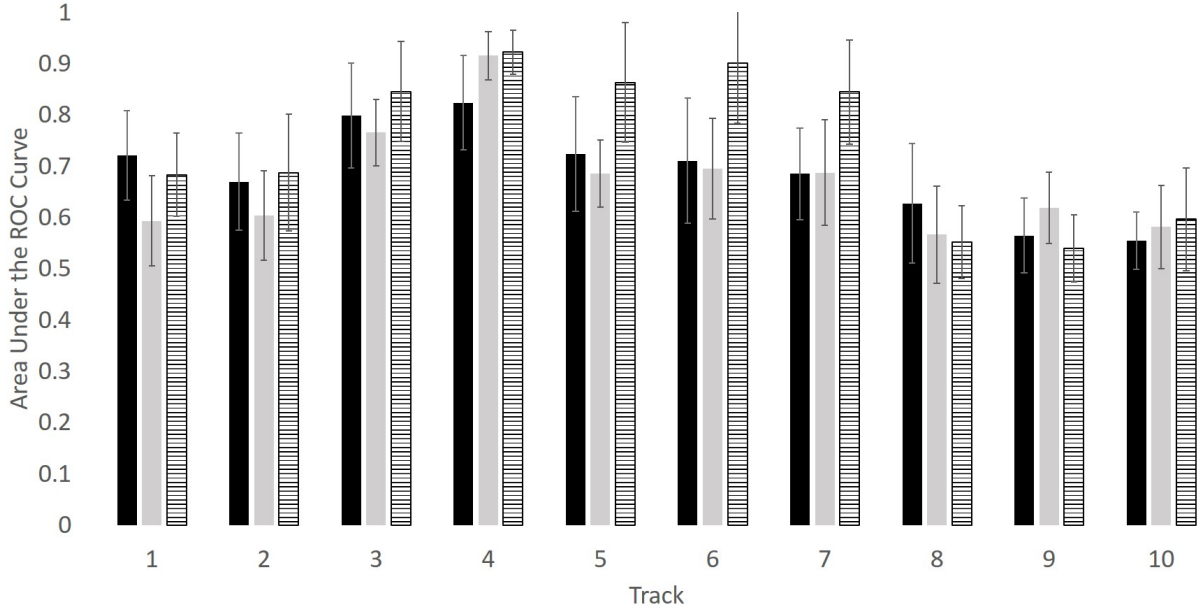


Figure 2.9: Classification accuracy for the dynamic state of the sonarhead on the basis of foliage echoes as a function of location (track) and echo representation. Classification performance was quantified by area under the curve (AUC) of the receiver operating characteristics (ROC). The classifications were performed on multiple subsets which contained 420 foliage echoes each with a track being represented by N subsets (N = 6 to 23, based on track size). All subsets contained a 50/50-split between static and dynamic sonarhead states and were collected over a distance of approximately 20 m. Mean values (bar height) and standard deviations (error bars) were calculated across all such subsets of the respective track. The bar group for each track contains spike-time representation with gammachirp BM model (black), spectrogram (gray), and spike-time representation with DRNL BM model (striped).

Classification accuracy decreased with dataset size and this behavior was observed across different signal representations and datasets from different locations (figure 2.10). Overall, classification performance decreased with dataset size from a maximum mean AUC of 0.85 for 420 echoes to a man AUC of 0.6 for a dataset size of 7,980 echoes. For the larger data sets, the training accuracy was much higher (e.g., ∼90% for 7,980 echoes) than the validation accuracy (∼60%) indicating the presence of overtraining.
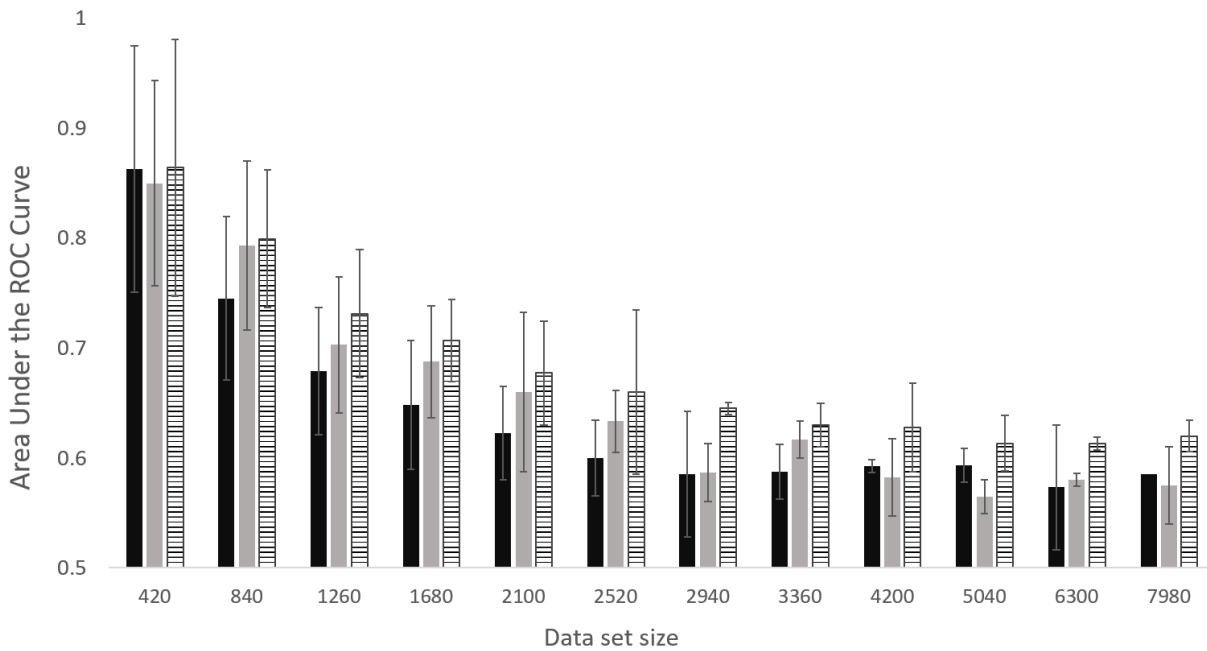


Figure 2.10: Classification performance as a function of dataset size. Classification accuracy was quantified by the area under the ROC curve (AUC) and the data for this experiment came from three different tracks (Track 3 – black, Track 7 – gray, Track 5 – stripes). Mean values (bar height) and standard deviations (error bars) were calculated from N repeat trials with the same sample size (N = 10 for 420 echo subsets, and N = 2 for 7980 echo subsets).

The classification accuracy also decreased when the physical distance between the locations of echo collection for the training and validation datasets was increased (figure 2.11). The classification accuracy was maximum ∼0.81 AUC when the mean recordings position of the testing dataset were closest to those of the training set. When the test set contained out of

sample echoes from the same location as the training set (test set comprised of echoes which were excluded from training set but from the same location), the accuracy was between between 0.76 and 0.81 AUC; for distances of 17 m, it decreased to chance level (0.5 AUC).
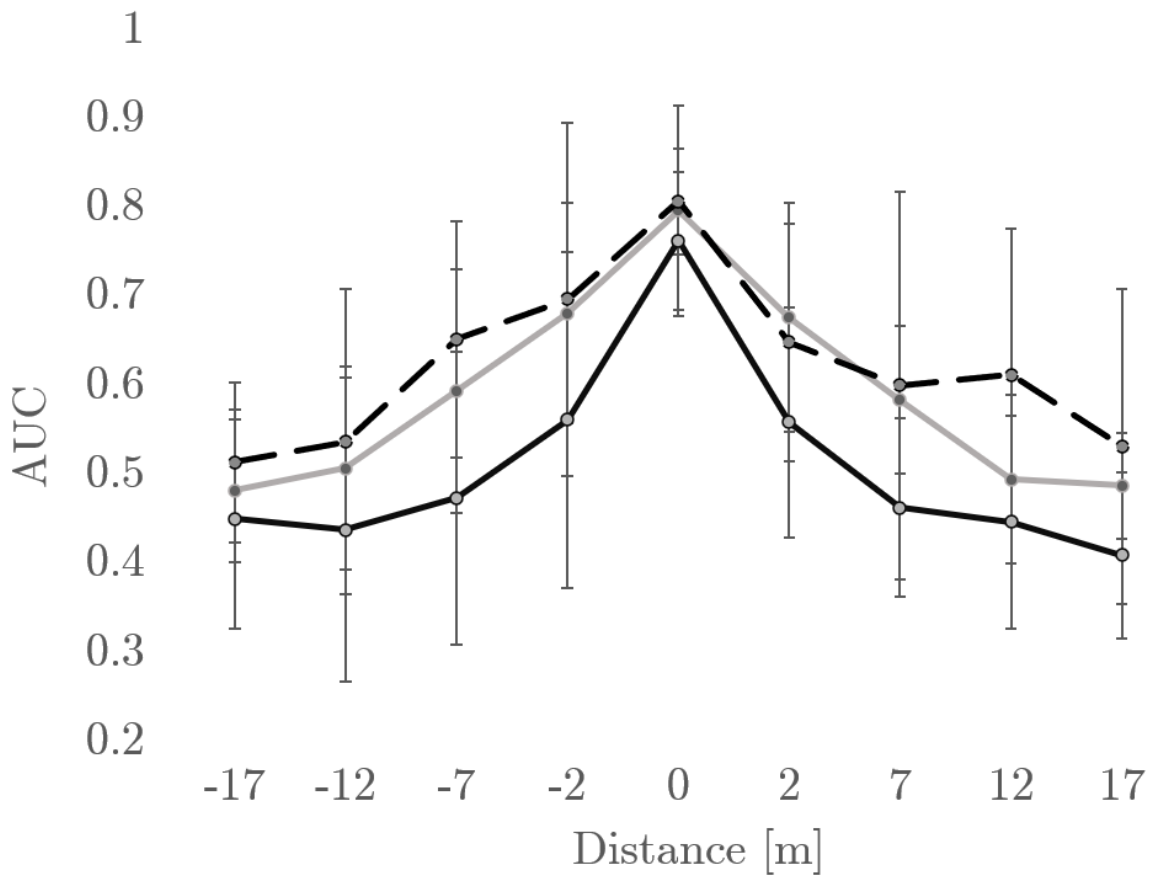


Figure 2.11: Classification accuracy (AUC) as a function of physical distance between testing and training dataset locations. Separate datasets, containing echoes recorded from two different physical locations were used for training and testing the classification network. The test sets were selected with a mean distance from the training set in steps of 5 m, from -17 m to 17 m. The mean and error bars were generated from 16 repeat trials of the experiment, with different training and test sets while maintaining the same distances between the datasets. Example results are shown for three different combinations of tracks and echo representations: track 5 and spectrogram (black dashed), track 7 and DRNL BM model (gray), track 3 and gammachirp (black solid).

## 2.6   Discussion

Our results demonstrate that foliage echoes can reflect the peripheral dynamics of a biomimetic sonar system. This finding goes beyond prior art since foliage echoes have a random nature [64] that is absent in numerical [54, 56] or physical device characterizations [57, 80] as well as in echoes from targets with simple geometry [61]. The closest prior research result so far [61] experimented with small, artificial plants that are also unlikely to capture the full range of variability in echoes that originate in natural environments. The pair-wise echo correlation analysis carried out here has demonstrated that the foliage echoes in our data set did not contain any shared deterministic patterns beyond the common sonar pulse that was also present in the random reference. Having demonstrated the detectability of effects due to the peripheral dynamics in echoes that represent the full extent of natural variability establishes that bats and biomimetic sensory systems modeled after them could exploit such effects. It does not, however, answer the question of whether such time-variant effects convey useful sensory information and how it may be exploited.

The second major finding of this work has been that the peripheral dynamics state can be classified using spike-based echo representations. This implies that these time-variant effects persist through all stages of auditory processing and especially the conversion to a sparse spike representation. Prior numerical experiments have shown spike representations preserve spectral cues to enable direction finding in bats [102], but these cues were strong deterministic patterns that can be captured by a dense rate code, as the researchers did not implement a binning process to maintain higher precision of spike timing. Whereas in our analysis, the spike representation resulted in a data reduction of 25% down from the original size and yet the effects of the peripheral dynamics were still detectable. Research on neuromorphic signal processing based on spikes [103, 104] has a demonstrated potential for applications in autonomous navigation with respect to representing sensor input and carry-

ing out motor control of a robot [105]. Furthermore, neuromorphic signal representations and processing have been used in conjunction with vision sensors to solve the challenges of autonomous navigation and obstacle avoidance in robots in cluttered environments [106]. A biomimetic sonar system combined with a sparse neuromorphic signal representation could deliver navigation in complex, natural environments in a highly parsimonious fashion. Our results are a step in this direction since the demonstrate that even non-obvious time-variant effects can be preserved in a sparse spike code.

In the classification experiments conducted here, the accuracy decreased when the spatial data-collection region was widened or the distance between the collection areas of the training and test data sets was increased. We hypothesize that this is due to changes in the vegetation characteristics with distance that our deep-learning classifier was not able to generalize for. It may still be that other classifiers could handle these spatial effects better than the ones used here. For example, a "space-invariant" classifier could be created by linking two classifiers, one for the dynamic state of the sonarhead and the other for the vegetation type. Having this adaptability could allow a bat or a biomimetic system to extract dynamics-related echo properties despite the variability in the foliage echoes over distance.

The basilar membrane models were found to significantly affect classification accuracy, with the nonlinear DRNL model performing better than the linear models. A simple explanation for this could be that the DRNL model included a larger number of parameters, although systematic optimization of the model parameters to improve classification performance would be a task for future work. Guidance for further refinement of the model and good parameter values could come from future neurophysiological experiments with horseshoe bats.

Changes in the parameters of the spike models affected the classification performance, but the maximum classification performance across both spike models after adjusting the parameters was similar. The LIAF model contained fewer parameters compared to the Kernel model and

therefore required fewer parameter changes. We observed that the spike timing plots from both models were similar in terms of spike counts and interspike intervals after adjusting for performance, which offers an explanation for the likeness in classification accuracy.

In the experiments reported here, the biomimetic sonarhead repeated the same noseleaf and pinna motions patterns throughout all data acquisitions. In contrast to that, bats are known to alternate between rigid and non rigid motions [107] and have been shown to possess a large range of variability at least in the rigid motions [108]. Hence, it may be hypothesized that this variability provides a substrate for adaptation that would allow the animals to encode more useful sensory information. Future behavioral animal studies could shed light on whether such relationships exist. Adaptation of peripheral dynamic patterns could also be studied with biomimetic robots like the one used here. Together, such biological and biomimetic experiments could eventually unravel how the bat biosonar systems are able to deal with the full complexity of the animals' environments.

## 2.7    Acknowledgements

# Chapter 3

# Conclusions

## 3.1 Research Accomplishments

- A large dataset of biosonar foliage echoes (55,000+) was recorded from different sites around the Virginia Tech campus using a biomimetic sonarhead. The sonarhead mimicked the periphery dynamics and pulse design of horseshoe bats. The recorded dataset included an equal split of echoes, 50% corresponding to a static periphery and the other 50% to moving or 'dynamic' periphery.

- To generate a neuromorphic spike based representation of the echoes, an auditory signal processing model was developed. The model included three main blocks, i) Cochlear Model, ii) Inner hair cell model, iii) Spike response model. Three cochlear models, listed in an increasing order of complexity: Gammatone, Gammachirp and the DRNL model, were adapted to replicate the physiology of horseshoe bats. A simplified model of the inner hair cells (IHCs) was used to simulate the mechanical to electric signal transduction, with half wave rectification followed by low pass filtering of signals. To model the spiking activity, two spiking neuron models, Leaky integrate and fire (LIAF) and Response kernel were used. The model converted input waveforms of the echoes into a spike based representation.

- Machine learning models were built to classify the foliage echoes between the two

conditions of the periphery state: 'static' or 'dynamic'. Two separate machine learning models were developed for the different input representations, spectrograms, and spike timing plots.

- Classification based on periphery state, as 'static' vs 'dynamic' was successfully achieved with both signal representations. The successful identification suggests that the effects of the peripheral dynamics exist within the complex and stochastic foliage echoes, to the extent that they can be identified. Furthermore, the classification based on the neuromorphic signal representation suggests that these effects persist through the auditory processing which yields a sparser signal representation (25% of original size).

## 3.2   Discussion of Findings

Our findings suggest that the effects of peripheral dynamics can be identified within foliage echoes, despite their complex and stochastic nature [64]. It is not yet known whether these effects encode any additional information within these foliage echoes, and hence our result does not point to their usefulness. However, we can speculate that the presence of these effects indicates the possibility that peripheral dynamics may have a functional role in bat biosonar systems within the specific context of navigation or foraging in foliage environments.

We demonstrated that the effects of the peripheral dynamics persist within the sparse neural spike representation of foliage echoes. In our model the output spike responses were binned, which reduced the size of the spike response arrays was reduced to 25% of its original dimension. This binning process was implemented such that the temporal precision in our spike model more accurately represented the biology of mammalian auditory systems [94, 95, 96, 97, 98]. The successful classification of these spike representations after the binning process indicates that these effects may exist within the neural spikes in the

bat brains. Therefore some information contained in echoes due to these effects may be available to the bat for behavioural decisions. Our results add to prior findings which employed numerical experiments to show that spike representations can preserve spectral cues in biosonar echoes to enable direction finding in bats [102]. However, this prior study did not employ a binning process and the directional cues were strong deterministic patterns that can be captured by a dense rate code.

These findings have potential applications in the development of autonomous robots which can use biomimetic hardware in conjunction with neuromorphic software to achieve autonomous mobility in complex cluttered environments. This combination of hardware and software can build upon existing efforts which have implemented spike based signal processing [103, 104] in autonomous navigation of robots. Research has demonstrated that neuromorphic signal processing can be utilized in autonomous navigation applications to represent sensor input and can that such sensory input can be used to control a robot [105]. Exciting recent research has been in the development of an autonomous robot which uses neuromorphic hardware in the form of a dynamic vision sensor, and combined with a spiking neural network based neuromorphic processor to effectively tackle some autonomous navigation challenges such as obstacle avoidance in cluttered environments [106]. Our results hold promise that a similar approach using biomimetic hardware and acoustic sensors may be possible in the future.

The biomimetic sonarhead in our experiment used the same simplified pinna and noseleaf motions for the entire dataset, however bats are known to be capable of different types of movements, classified as rigid and non rigid [107]. It has also been shown that the rigid motions show a large range of variability in the rigid motions [108]. Therefore the potential exists for different motions to be adapted to specific tasks which could enable greater information encoding within foliage echoes. Future behavioral studies of bats can shed further

light on this aspect, and that can then be incorporated into biomimetic reproductions to allow an even deeper analysis of their biosonar systems.

Our results indicated towards a "scene specificity", as the classifier was more effective in classifying echoes from the same location as compared to a larger data set with significant variation in the locations where the echoes were recorded. This result suggested that any potential advantages of these dynamics may be limited to the context of a particular surrounding if the bat brain similarly fails to adapt to such changes. In this case, the bat would need to readjust to different surroundings to exploit any benefits from these dynamics.

## 3.3   Suggestions for Future Work

- Whereas our results demonstrated that the presence of the effects of dynamics can be identified within foliage echoes, they do not provide any insight into their potential usefulness to a bat. Further study of these echoes, from an information theoretical perspective can potentially demonstrate whether these effects encode additional information within these signals.

- Basilar membrane models were found to have an effect on the classification accuracy of the echoes. An analysis of how different basilar membrane models affect output classification, and how model parameters can be optimized for improved classification performance was not undertaken in this study. However, that remains a point of interest for future research. Since the models in our analysis used values for certain physiological parameters based on other mammalian cochlea in the absence of horseshoe bat data, updating those parameters to more closely represent horseshoe bats' physiology could enable a better understanding of these effects.

- The parameters of the spike models used to simulate the spiking activity of the auditory were found to affect the output classification accuracy. In this study, parameters were optimized roughly using a manual optimization process. An exhaustive parameter search in conjunction with an optimization algorithm that uses classification accuracy as feedback can potentially lead to improved performance and may be of interest in the future.

- Bats are capable of executing a number of different and highly complex motions when they deform their pinna and noseleaf, whereas in the experiments here, a simplified movement was implemented. There exists a possibility that different ear motions are suited in certain contexts, and the correct pairing may enable better understanding of the effects of these motions in foliage echoes. Hence further behavioral research of bats, and their incorporation into the biomimetic experiement remains of interest.

- Development of a machine learning classifier which can exceed the performance achieved here can be another potential task for future work. Better classification accuracy may enable an improved qualitative understanding of how the effects of these dynamics appear in echoes, based on which aspects of the signals the classifier uses to identify differences. We also found that our machine learning model failed to generalize with larger datasets which contained echoes from different areas, however building a better machine learning model which is robust to the variation within the echoes may be possible.

- Our results showed that the effects of these dynamics exist in foliage echoes. Whereas the usefulness of such effects, and the mechanisms through which they potentially encode information is yet unknown. However, in the absence of that understanding, studies into potential uses of the effects due to these dynamics are possible. An idea would be a study on the impact of dynamics' effects on detection and classification

tasks.

# Bibliography

[1] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ,"
https://www.openstreetmap.org, 2017.

[2] A. Lakhtakia and R. J. Martín-Palma, *Engineered biomimicry.* Oxford, UK: Newnes,
2013.

[3] H. Dong, A. T. Bode-Oke, and C. Li, "Learning from nature: unsteady flow physics in
bioinspired flapping flight," *Flight physics-models, techniques and technologies*, 2018.

[4] H. Liu, S. Ravi, D. Kolomenskiy, and H. Tanaka, "Biomechanics and biomimetics
in insect-inspired flight systems," *Philos. Trans. R. Soc. B.*, vol. 371, no. 1704, p.
20150390, 2016.

[5] C. Sanchez, H. Arribart, and M. M. G. Guille, "Biomimetism and bioinspiration as
tools for the design of innovative materials and systems," *Nat. Mater.*, vol. 4, no. 4,
pp. 277–288, 2005.

[6] G. M. Whitesides, "Bioinspiration: something for everyone," *Interface focus*, vol. 5,
no. 4, p. 20150031, 2015.

[7] B. Theckes, E. De Langre, and X. Boutillon, "Damping by branching: a bioinspiration
from trees," *Bioinspir. Biomim.*, vol. 6, no. 4, p. 046010, 2011.

[8] J. Tao and X. B. Yu, "Hair flow sensors: from bio-inspiration to bio-mimicking—a
review," *Smart Mater. Struct.*, vol. 21, no. 11, p. 113001, 2012.

[9] A. H. Proppe, Y. C. Li, A. Aspuru-Guzik, C. P. Berlinguette, C. J. Chang, R. Cogdell,

A. G. Doyle, J. Flick, N. M. Gabor, R. van Grondelle *et al.*, "Bioinspiration in light harvesting and catalysis," *Nat. Rev. Mater.*, pp. 1–19, 2020.

[10] M. D. Database, "Mammal diversity database," Sep. 2020.

[11] G. Bunget and S. Seelecke, "Batmav: a 2-dof bio-inspired flapping flight platform," in *Active and Passive Smart Structures and Integrated Systems 2010*, vol. 7643. International Society for Optics and Photonics, 2010, p. 76433B.

[12] A. Skulborstad, S. Swartz, and N. Goulbourne, "Biaxial mechanical characterization of bat wing skin," *Bioinspir. Biomim.*, vol. 10, no. 3, p. 036004, 2015.

[13] M. Johnson, P. T. Madsen, W. M. Zimmer, N. Aguilar de Soto, and P. L. Tyack, "Beaked whales echolocate on prey," *Proc. Royal Soc. B. Biol. Sci.*, vol. 271, no. suppl_6, pp. S383–S386, 2004.

[14] J. A. Thomas, C. F. Moss, and M. Vater, *Echolocation in bats and dolphins.* University of Chicago Press, 2004.

[15] S. Brinkløv, M. B. Fenton, and J. M. Ratcliffe, "Echolocation in oilbirds and swiftlets," *Front. Physiol.*, vol. 4, p. 123, 2013.

[16] S. A. Kick, "Target-detection by the echolocating bat, eptesicus fuscus," *J. Comp. Physiol.*, vol. 145, no. 4, pp. 431–435, 1982.

[17] J. Simmons, S. Kick, B. Lawrence, C. Hale, C. Bard, and B. Escudie, "Acuity of horizontal angle discrimination by the echolocating bat, eptesicus fuscus," *J. Comp. Physiol.*, vol. 153, no. 3, pp. 321–330, 1983.

[18] H.-U. Schnitzler and W. H. O'Dell, "Performance of airborne animal sonar systems: I. microchiroptera," in *Animal sonar systems.* Springer, 1980, pp. 109–181.

[19] R. Davis, "Homing performance and homing ability in bats," *Ecol. Monogr.*, vol. 36, no. 3, pp. 201–237, 1966.

[20] T. C. Williams, J. M. Williams, and D. R. Griffin, "The homing ability of the neotropical bat phyllostomus hastatus, with evidence for visual orientation," *Anim. Behav.*, vol. 14, no. 4, pp. 468–473, 1966.

[21] J. Guilbert, M. Walker, S. Greif, and S. Parsons, "Evidence of homing following translocation of long-tailed bats (chalinolobus tuberculatus) at grand canyon cave, new zealand," *N. Z. J. Zool.*, vol. 34, no. 3, pp. 239–246, 2007.

[22] J. R. Barchi, J. M. Knowles, and J. A. Simmons, "Spatial memory and stereotypy of flight paths by big brown bats in cluttered surroundings," *J. Exp. Biol.*, vol. 216, no. 6, pp. 1053–1063, 2013.

[23] P. H.-S. Jen and T. Kamada, "Analysis of orientation signals emitted by the cf-fm bat, pteronotus p. parnellii and the fm bat, eptesicus fuscus during avoidance of moving and stationary obstacles," *J. Comp. Physiol.*, vol. 148, no. 3, pp. 389–398, 1982.

[24] A. Surlykke, S. Boel Pedersen, and L. Jakobsen, "Echolocating bats emit a highly directional sonar sound beam in the field," *Proc. Royal Soc. B. Biol. Sci.*, vol. 276, no. 1658, pp. 853–860, 2009.

[25] E. Fujioka, S. Mantani, S. Hiryu, H. Riquimaroux, and Y. Watanabe, "Echolocation and flight strategy of japanese house bats during natural foraging, revealed by a microphone array system," *J. Acoust. Soc. Am.*, vol. 129, no. 2, pp. 1081–1088, 2011.

[26] G. Neuweiler, W. Metzner, U. Heilmann, R. Rübsamen, M. Eckrich, and H. Costa, "Foraging behaviour and echolocation in the rufous horseshoe bat (rhinolophus rouxi) of sri lanka," *Behav. Ecol. Sociobiol.*, vol. 20, no. 1, pp. 53–67, 1987.

[27] C. Yu, J. Luo, M. Wohlgemuth, and C. F. Moss, "Echolocating bats inspect and discriminate landmark features to guide navigation," *J. Exp. Biol.*, vol. 222, no. 8, 2019.

[28] C. Chiu, W. Xian, and C. F. Moss, "Adaptive echolocation behavior in bats for the analysis of auditory scenes," *J. Exp. Biol.*, vol. 212, no. 9, pp. 1392–1404, 2009.

[29] M. Warnecke, C. Chiu, J. Engelberg, and C. F. Moss, "Active listening in a bat cocktail party: adaptive echolocation and flight behaviors of big brown bats, eptesicus fuscus, foraging in a cluttered acoustic environment," *Brain Behav. Evol.*, vol. 86, no. 1, pp. 6–16, 2015.

[30] J. Simmons, W. Lavender, B. Lavender, C. Doroshow, S. Kiefer, R. Livingston, A. Scallet, and D. Crowley, "Target structure and echo spectral discrimination by echolocating bats," *Science*, vol. 186, no. 4169, pp. 1130–1132, 1974.

[31] T. F. Villa, F. Gonzalez, B. Miljievic, Z. D. Ristovski, and L. Morawska, "An overview of small unmanned aerial vehicles for air quality measurements: Present applications and future prospectives," *Sensors*, vol. 16, no. 7, p. 1072, 2016.

[32] S. Waharte and N. Trigoni, "Supporting search and rescue operations with uavs," in *2010 International Conference on Emerging Security Technologies.* IEEE, 2010, pp. 142–147.

[33] A. Restas *et al.*, "Drone applications for supporting disaster management," *World J. Eng. Technol.*, vol. 3, no. 03, p. 316, 2015.

[34] M. Heutger and M. Kückelhaus, "Unmanned aerial vehicles in logistics a dhl perspective on implications and use cases for the logistics industry," *DHL Cust. Sol. Innov.*, 2014.

[35] S. Gade, A. A. Paranjape, and S.-J. Chung, "Herding a flock of birds approaching an airport using an unmanned aerial vehicle," in *AIAA Guidance, Navigation, and Control Conference*, 2015, p. 1540.

[36] P. Sigrist, P. Coppin, and M. Hermy, "Impact of forest canopy on quality and accuracy of gps measurements," *Int. J. Remote. Sens.*, vol. 20, no. 18, pp. 3595–3610, 1999.

[37] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert, "Learning monocular reactive uav control in cluttered natural environments," in *2013 IEEE international conference on robotics and automation*. IEEE, 2013, pp. 1765–1772.

[38] J. Q. Cui, S. Lai, X. Dong, and B. M. Chen, "Autonomous navigation of uav in foliage environment," *J. Intell. Robot. Syst.*, vol. 84, no. 1-4, pp. 259–276, 2016.

[39] T. G. Phillips, N. Guenther, and P. R. McAree, "When the dust settles: The four behaviors of lidar in the presence of fine airborne particulates," *J. Field Robot.*, vol. 34, no. 5, pp. 985–1009, 2017.

[40] K. Anderson and K. J. Gaston, "Lightweight unmanned aerial vehicles will revolutionize spatial ecology," *Front. Ecol. Environ.*, vol. 11, no. 3, pp. 138–146, 2013.

[41] J. F. Eisenberg and D. E. Wilson, "Relative brain size and feeding strategies in the chiroptera," *Evolution*, pp. 740–751, 1978.

[42] D. Vanderelst, J. Steckel, A. Boen, H. Peremans, and M. W. Holderied, "Place recognition using batlike sonar," *Elife*, vol. 5, p. e14188, 2016.

[43] J. Steckel and H. Peremans, "Batslam: Simultaneous localization and mapping using biomimetic sonar," *PLOS ONE*, vol. 8, no. 1, pp. 1–11, 01 2013.

[44] I. Eliakim, Z. Cohen, G. Kosa, and Y. Yovel, "A fully autonomous terrestrial bat-like acoustic robot," *PLOS Comput. Biol.*, vol. 14, no. 9, pp. 1–13, 09 2018.

[45] G. Jones, "Prey selection by the greater horseshoe bat (rhinolophus ferrumequinum): optimal foraging by echolocation?" *J. Anim. Ecol.*, pp. 587–602, 1990.

[46] H.-U. Schnitzler and E. K. Kalko, "Echolocation by insect-eating bats: We define four distinct functional groups of bats and find differences in signal structure that correlate with the typical echolocation tasks faced by each group," *Bioscience*, vol. 51, no. 7, pp. 557–569, 2001.

[47] G. Jones and J. M. Rayner, "Foraging behavior and echolocation of wild horseshoe bats rhinolophus ferrumequinum and r. hipposideros (chiroptera, rhinolophidae)," *Behav. Ecol. Sociobiol.*, vol. 25, no. 3, pp. 183–191, 1989.

[48] G. Schuller and G. Pollak, "Disproportionate frequency representation in the inferior colliculus of doppler-compensating greater horseshoe bats: evidence for an acoustic fovea," *J. Comp. Physiol.*, vol. 132, no. 1, pp. 47–54, 1979.

[49] R. Kober and H.-U. Schnitzler, "Information in sonar echoes of fluttering insects available for echolocating bats," *J. Acoust. Soc. Am.*, vol. 87, no. 2, pp. 882–896, 1990.

[50] V. Bruns and E. Schmieszek, "Cochlear innervation in the greater horseshoe bat: demonstration of an acoustic fovea," *Hear. Res.*, vol. 3, no. 1, pp. 27–43, 1980.

[51] H.-U. Schnitzler, "Control of doppler shift compensation in the greater horseshoe bat, rhinolophus ferrumequinum," *J. Comp. Physiol.*, vol. 82, no. 1, pp. 79–92, 1973.

[52] H. Schneider and F. P. Möhres, "Die ohrbewegungen der hufeisenfledermäuse (chiroptera, rhinolophidae) und der mechanismus des bildhörens," *J. Comp. Physiol.*, vol. 44, no. 1, pp. 1–40, 1960.

[53] D. Griffin, D. Dunning, D. Cahlander, and F. Webster, "Correlated orientation sounds and ear movements of horseshoe bats," *Nature*, vol. 196, no. 4860, pp. 1185–1186, 1962.

[54] L. Gao, S. Balakrishnan, W. He, Z. Yan, and R. Müller, "Ear deformations give bats a physical mechanism for fast adaptation of ultrasonic beam patterns," *Phys. Rev. Lett.*, vol. 107, no. 21, p. 214301, 2011.

[55] L. Feng, L. Gao, H. Lu, and R. Müller, "Noseleaf dynamics during pulse emission in horseshoe bats," *PLOS ONE*, vol. 7, no. 5, p. e34685, 2012.

[56] W. He, S. Pedersen, A. Gupta, J. Simmons, and R. Müller, "Lancet dynamics in greater horseshoe bats, rhinolophus ferrumequinum," *PLOS ONE*, vol. 10, no. 4, p. e0121700, 2015.

[57] Y. Fu, P. Caspers, and R. Müller, "A dynamic ultrasonic emitter inspired by horseshoe bat noseleaves," *Bioinspir. Biomim.*, vol. 11, no. 3, p. 036007, 2016.

[58] M. Pannala, S. Z. Meymand, and R. Müller, "Interplay of static and dynamic features in biomimetic smart ears," *Bioinspir. Biomim.*, vol. 8, no. 2, p. 026008, may 2013. [Online]. Available: https://doi.org/10.1088%2F1748-3182%2F8%2F2%2F026008

[59] R. Müller, A. Gupta, H. Zhu, M. Pannala, U. Gillani, Y. Fu, P. Caspers, and J. Buck, "Dynamic substrate for the physical encoding of sensory information in bat biosonar," *Phys. Rev. Lett.*, vol. 118, no. 15, p. 158102, 2017.

[60] X. Yin and R. Müller, "Fast-moving bat ears create informative doppler shifts," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 116, no. 25, pp. 12 270–12 274, 2019.

[61] J. Sutlive and R. Müller, "Dynamic echo signatures created by a biomimetic sonar head," *Bioinspir. Biomim.*, vol. 14, no. 6, p. 066014, 2019.

[62] M. B. Fenton, "The foraging behaviour and ecology of animal-eating bats," *Can. J. Zool.*, vol. 68, no. 3, pp. 411–422, 1990.

[63] M. E. Bates, J. A. Simmons, and T. V. Zorikov, "Bats use echo harmonic structure to distinguish their targets from background clutter," *Science*, vol. 333, no. 6042, pp. 627–630, 2011.

[64] R. Müller and R. Kuc, "Foliage echoes: a probe into the ecological acoustics of bat echolocation," *J. Acoust. Soc. Am.*, vol. 108, no. 2, pp. 836–845, 2000.

[65] C. D. Geisler, *From sound to synapse: physiology of the mammalian ear.* New York, NY, USA: Oxford University Press, USA, 1998.

[66] G. Neuweiler *et al.*, *The biology of bats.* New York, NY, USA: Oxford University Press on Demand, 2000.

[67] S. Haplea, E. Covey, and J. Casseday, "Frequency tuning and response latencies at three levels in the brainstem of the echolocating bat, eptesicus fuscus," *J. Comp. Physiol.*, vol. 174, no. 6, pp. 671–683, 1994.

[68] B. Willmore and D. J. Tolhurst, "Characterizing the sparseness of neural codes," *Netw. Comput. Neural Syst.*, vol. 12, no. 3, pp. 255–270, 2001.

[69] N. Suga, G. Neuweiler, and J. Möller, "Peripheral auditory tuning for fine frequency analysis by the cf-fm bat, rhinolophus ferrumequinum," *J. Comp. Physiol.*, vol. 106, no. 1, pp. 111–125, 1976.

[70] V. Bruns, "Peripheral auditory tuning for fine frequency analysis by the cf-fm bat, rhinolophus ferrumequinum," *J. Comp. Physiol.*, vol. 106, no. 1, pp. 77–86, 1976.

[71] D. Floreano and R. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.

[72] R. W. Beard and T. W. McLain, *Small unmanned aircraft: Theory and practice.* Princeton, NJ, USA: Princeton university press, 2012.

[73] L. Mejias, J. Lai, and T. Bruggemann, "Sensors for missions," in *Handbook of Unmanned Aerial Vehicles.* Springer, 2015, pp. 385–399.

[74] Y. Lu, Z. Xue, G. Xia, and L. Zhang, "A survey on vision-based uav navigation," *Geo. Spat. Inf. Sci.*, vol. 21, no. 1, pp. 21–32, 2018.

[75] H. Schnitzler, C. Moss, and A. Denzinger, "From spatial orientation to food acquisition in echolocating bats," *Trends Ecol. Evol.*, vol. 18, no. 8, pp. 386–394, 2003.

[76] G. Jones, M. Morton, P. Hughes, and R. Budden, "Echolocation, flight morphology and foraging strategies of some west african hipposiderid bats," *J. Zool*, vol. 230, no. 3, pp. 385–400, 1993.

[77] Y. Yamada, K. Ito, A. Oka, S. Tateiwa, T. Ohta, R. Kobayashi, S. Hiryu, and Y. Watanabe, "Obstacle-avoidance navigation by an autonomous vehicle inspired by a bat biosonar strategy," in *Conference on Biomimetic and Biohybrid Systems.* Springer, 2015, pp. 135–144.

[78] I. Eliakim, Z. Cohen, G. Kosa, and Y. Yovel, "A fully autonomous terrestrial bat-like acoustic robot," *PLOS Comput. Biol.*, vol. 14, no. 9, p. e1006406, 2018.

[79] Y. Lu, Z. Xue, G.-S. Xia, and L. Zhang, "A survey on vision-based uav navigation," *Geo. Spat. Inf. Sci.*, vol. 21, no. 1, pp. 21–32, 2018.

[80] S. Z. Meymand, M. Pannala, and R. Müller, "Characterization of the time-variant behavior of a biomimetic beamforming baffle," *J. Acoust. Soc. Am.*, vol. 133, no. 2, pp. 1141–1150, 2013.

[81] C. Ming, A. K. Gupta, R. Lu, H. Zhu, and R. Müller, "A computational model for biosonar echoes from foliage," *PLOS ONE*, vol. 12, no. 8, p. e0182824, 2017.

[82] C. Ming, H. Zhu, and R. Müller, "A simplified model of biosonar echoes from foliage and the properties of natural foliages," *PLOS ONE*, vol. 12, no. 12, p. e0189824, 2017.

[83] B. Fontaine, H. Peremans, and J. Steckel, "3d sparse imaging in biosonar scene analysis," 2009.

[84] H. Schnitzler, "The ultrasonic locating sounds of the horseshoe bats ( chiroptera-rhinolophidae ) in different orientation situations," *J. Comp. Physiol.*, vol. 57, no. 4, pp. 376–408, 1968.

[85] R. Müller and H. Schnitzler, "Acoustic flow perception in cf-bats: extraction of parameters," *J. Acoust. Soc. Am.*, vol. 108, no. 3, pp. 1298–1307, 2000.

[86] R. D. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice, "An efficient auditory filterbank based on the gammatone function," in *a meeting of the IOC Speech Group on Auditory Modelling at RSRE*, vol. 2, no. 7, 1987.

[87] T. Irino and R. Patterson, "A time-domain, level-dependent auditory filter: The gammachirp," *J. Acoust. Soc. Am.*, vol. 101, no. 1, pp. 412–419, 1997.

[88] R. Meddis, L. P. O'Mard, and E. A. Lopez-Poveda, "A computational algorithm for computing nonlinear auditory frequency selectivity," *J. Acoust. Soc. Am.*, vol. 109, no. 6, pp. 2852–2861, 2001.

[89] E. A. Lopez-Poveda and R. Meddis, "A human nonlinear cochlear filterbank," *J. Acoust. Soc. Am.*, vol. 110, no. 6, pp. 3107–3118, 2001.

[90] D. Jaeger and R. Jung, *Encyclopedia of computational neuroscience.* New York, NY, USA: Springer Publishing Company, Incorporated, 2015.

[91] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal dynamics: From single neurons to networks and models of cognition.* Cambridge, UK: Cambridge University Press, 2014.

[92] B. Wang, W. Ke, J. Guang, G. Chen, L. Yin, S. Deng, Q. He, Y. Liu, T. He, R. Zheng *et al.*, "Firing frequency maxima of fast-spiking neurons in human, monkey, and mouse neocortex," *Front. Cell. Neurosci.*, vol. 10, p. 239, 2016.

[93] P. Heil and A. J. Peterson, "Basic response properties of auditory nerve fibers: a review," *Cell Tissue Res.*, vol. 361, no. 1, pp. 129–158, 2015.

[94] D. H. Johnson, "The relationship between spike rate and synchrony in responses of auditory-nerve fibers to single tones," *J. Acoust. Soc. Am.*, vol. 68, no. 4, pp. 1115–1122, 1980.

[95] A. N. Temchin and M. A. Ruggero, "Phase-locked responses to tones of chinchilla auditory nerve fibers: implications for apical cochlear mechanics," *J. Assoc. Res. Otolaryngol.*, vol. 11, no. 2, pp. 297–318, 2010.

[96] J. E. Rose, J. F. Brugge, D. J. Anderson, and J. E. Hind, "Phase-locked response to low-frequency tones in single auditory nerve fibers of the squirrel monkey." *J. Neurophysiol.*, vol. 30, no. 4, pp. 769–793, 1967.

[97] A. Dreyer and B. Delgutte, "Phase locking of auditory-nerve fibers to the envelopes of high-frequency sounds: implications for sound localization," *J. Neurophysiol.*, vol. 96, no. 5, pp. 2327–2341, 2006.

[98] C. J. Sumner and A. R. Palmer, "Auditory nerve fibre responses in the ferret," *Eur. J. Neurosci.*, vol. 36, no. 4, pp. 2428–2439, 2012.

[99] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[100] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous systems," 2015.

[101] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[102] B. Fontaine, H. Peremans, and B. Schrauwen, "Bat echolocation modelling using spike kernels with support vector regression," in *Proceedings of the 15th European Symposium on Artificial Neural Networks. p. 367-372 2007*, 2007, pp. 367–372.

[103] S.-C. Liu, N. Mesgarani, J. Harris, and H. Hermansky, "The use of spike-based representations for hardware audition systems," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems.* IEEE, 2010, pp. 505–508.

[104] A. Jiménez-Fernández, E. Cerezuela-Escudero, L. Miró-Amarante, M. J. Domínguez-Morales, F. de Asís Gómez-Rodríguez, A. Linares-Barranco, and G. Jiménez-Moreno, "A binaural neuromorphic auditory sensor for fpga: A spike signal processing approach," *IEEE Trans. Neural Netw.*, vol. 28, no. 4, pp. 804–818, 2016.

[105] J. P. Mitchell, G. Bruer, M. E. Dean, J. S. Plank, G. S. Rose, and C. D. Schuman, "Neon: Neuromorphic control for autonomous robotic navigation," in *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS).* IEEE, 2017, pp. 136–142.

[106] M. B. Milde, H. Blum, A. Dietmüller, D. Sumislawska, J. Conradt, G. Indiveri, and Y. Sandamirskaya, "Obstacle avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic processing system," *Front. Neurorobot.*, vol. 11, p. 28, 2017.

[107] X. Yin, P. Qiu, L. Yang, and R. Müller, "Horseshoe bats and old world leaf-nosed bats have two discrete types of pinna motions," *J. Acoust. Soc. Am.*, vol. 141, no. 5, pp. 3011–3017, 2017.

[108] P. Qiu and R. Müller, "Variability in the rigid pinna motions of hipposiderid bats and their impact on sensory information encoding," *J. Acoust. Soc. Am.*, vol. 147, no. 1, pp. 469–479, 2020.

# Appendices

# Appendix A

# First Appendix

## A.1 Location Data

Table A.1: Location data.

| Track number | Location | Number of echoes |
|:---:|:---:|:---:|
| 1 | Stadium Woods | 2,940 |
| 2 | Stadium Woods | 5,460 |
| 3 | Stadium Woods | 9,660 |
| 4 | Grove Lane | 5,460 |
| 5 | Grove Lane | 8,820 |
| 6 | Corporate Research Center | 4,200 |
| 7 | Inventive Lane | 9,660 |
| 8 | Inventive Lane | 4,200 |
| 9 | Corporate Research Center | 2,520 |
| 10 | Inventive Lane | 2,520 |

## A.2 Classification Accuracy with Data Size

Table A.2: Classification accuracy with different dataset sizes.

| Dataset Size | AUC (mean) |
|:---:|:---:|
| 54,600 | 0.56 |
| 2,920 | 0.59 |
| 420 | 0.79 |

## A.3    Correlation of Simulated Echoes
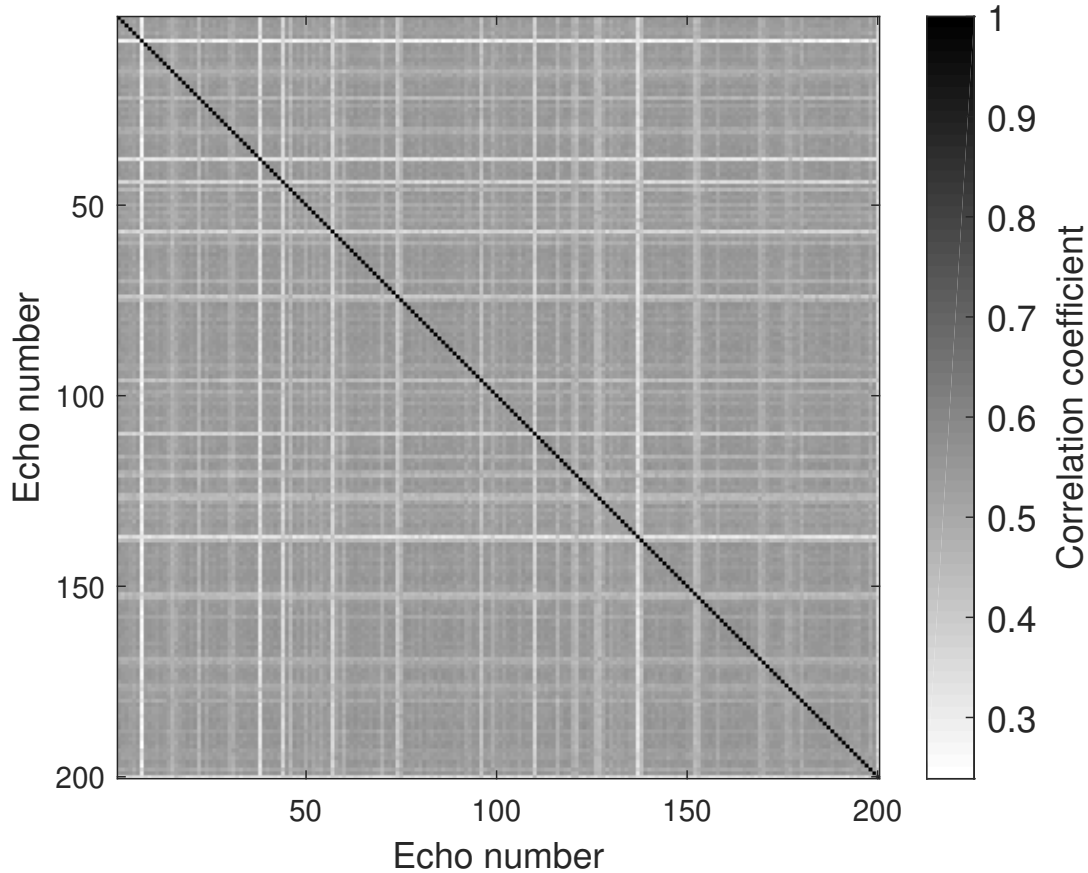


Figure A.1:   The similarity between simulated "echoes" was estimated through the calculation of the correlation between the recordings. The correlation coefficient was calculated between 200 simulated echoes generated through a convolution of the emitted pulse with artificial impulse responses

The baseline correlation was validated by a numerical experiment A.1, where we generated artificial foliage impulse responses from point reflectors. The amplitudes of point reflectors were drawn from a gaussian distribution and their timing was based on a poisson distribution. The correlation between these impulse responses was very low (0.17). The correlation between these artificial impulse responses convolved with the emission pulse, to generate simulated "echoes", and the average correlation was found to be 0.51, which matches our

data.

# Appendix B

# Auditory Model Code (MATLAB)

## B.1   Master file

```matlab
%% Master
%% Loading Data Set
%load('full_data.mat') %TODO for Full data

data_range = 'Mix';
BM_model = 1;
SR_model = 2;

for q = 1:2 %Two Loops for Static and Dynamic sets
    if q == 1
        file = full_sta;
        count = 1;
    else
        file = full_dyn;
        count = 2;
    end
    %% Center Frequencies Setting
    min_f       = 20e3;      % Minimum Frequency
    max_f       = 45e3;      % Maximum Frequency
```

```matlab
nChan           = 100;          % Number of channels
fs              = 400e3;        % Sampling rate Fs in Hz
[fc, ERB]       = fc_ERB_hb(min_f, max_f, nChan);


subt = 1;
rg1 = [10802 + (subt*210) : 10802+((subt+1)*210 -1)]; %static range
rg2 = [10911 + (subt*210) : 10911+((subt+1)*210 - 1)];%dynamic range
if q == 1
    data_mat    = double(file(6401:10000, rg1));
else
    data_mat    = double(file(6401:10000, rg2));
end
nData           = size(data_mat,2);
N               = 3600;
t               = 1/fs:1/fs:N/fs;


%% Models' Parameters
% BM's Parameters
leveldBSPL  = 50;
coef_b          = 1.019;        % Increasing or decreasing of b means increasing
    or decreasing ERB
coef_c          = -3;           % Controls the degree of asymmetry in gammachirp,
    when c=0, gammachirp=gammatone
coef_GTGC   = [coef_b, coef_c];
coef_d          = 3.1; %4.2040 3.1;    % Linear gain coef 1
coef_e          = -0.6; %-0.4791;  % Linear gain coef 2
coef_f          = 1.17274;  % Non-Linear bandwidth broken stick nonlinearity
    coef 1
coef_g          = 0.0113;       % Non-Linear bandwidth broken stick nonlinearity
    coef 2
coef_h          = 1.4030;   % a1 Broken-stick nonlinerity cofficients
```

```matlab
coef_i       = 0.8192;    % a2 Broken-stick  nonlinerity  cofficients

coef_j       = 1.6191;    % b1 Broken-stick  nonlinerity  cofficients

coef_k       = -0.8187;   % b2 Broken-stick  nonlinerity  cofficients

coef_l       = -0.6021;   % c1 Broken-stick  nonlinerity  cofficients

coef_m       = 0;         % c2 Broken-stick  nonlinerity  cofficients

coef_DRNL    = [coef_d, coef_e, coef_f, coef_g, coef_h, coef_i, coef_j,
    coef_k, coef_l, coef_m];

BM_par       = [N, fs, nChan, leveldBSPL, coef_GTGC, coef_DRNL];


% SRM's Parameters

V_rest       = 0;         % resting membrane potential [V]

V_reset      = 0;%-1e-6;     % after-potential hyperpolarization [V]

VarTheta     = 13.5e-6;      % spike threshold [V] %GC 13.5e-6; %DR

G_m          = 1.67e+2;   % membrane conductance [S] (MAP parameter human)

Tau_m        = 0.0065;    % membrane time constant [s] (MAP parameter for
    human) 0.0065 for Kernel

SRM_par      = [N, fs, nChan, V_rest, V_reset, VarTheta, G_m, Tau_m];


% Entropy's Parameters

dt       = 0.5e-3;           % Time bin for spike train entropy calc %TODO

T        = 9e-3;          % Word length for spike train entropy calc TODO if
    adding zeros 14

ENT_par = [N, fs, nData, dt, T];


%% Model

parfor i = 1:nData
    data_mat(:,i)          = data_mat(:,i) - mean(data_mat(:,i)); % Zero
        mean correction

    data           = commonbandpass(min_f, data_mat(:,i), max_f, fs);  % Low
        Pass Filter

    normalizer  = sqrt(sum(data.^2));
```

```matlab
    data          = data./normalizer;


% Basilar Membrane Displacement Function
    [BM_out(:,:,i)]      = BM_BOX(BM_model, data, BM_par, fc, ERB);  % 1: GT
        , 2: GC, 3: DRNL
    [IHC_out(:,:,i)]    = IHC_BOX(BM_out(:,:,i), fs, nChan);
    temp = IHC_out(:,:,i);
    for j = 1:nChan
        div = sqrt(sum(temp(:,j).^2));
        temp(:,j) = temp(:,j)./div;
    end


    %% Zeroing ISI start to eliminate low freq content at start if signal
    temp(1:900,1:35) = zeros(900,35);
    IHC_out(:,:,i) = temp;


    % Spiking Activity Function
    [SRM_out(:,:,i), Tspikes(:,i), V1(:,:,i)]  = SRM_BOX(SR_model,
        IHC_out(:,:,i), SRM_par);       % 1: Leaky IAF, 2: Kernel
end


fc = fc;


if count == 1
    SRM1 = SRM_out;
    [w1, train1, ST1] = ISI_func_ST(SRM1);
    V_st = V1;
else
    SRM2 = SRM_out;
    [w2, train2, ST2] = ISI_func_ST(SRM2);
    V_dy = V1;
```

```matlab
    end


end


s1 = sum(SRM1);
s1 = reshape(s1,nChan,nData);
s1 = sum(s1')./nData;


s2 = sum(SRM2);
s2 = reshape(s2,nChan,nData);
s2 = sum(s2')./nData;


if size(w1,2)-size(w2,2) == 0
    same = 1
elseif size(w1,2)-size(w2,2)>= 0
    w2 = [w2 zeros(size(w2,1),size(w1,2)-size(w2,2),size(w2,3))];
else
    w1 = [w1 zeros(size(w1,1),size(w2,2)-size(w1,2),size(w1,3))];
end
%% File Prep for CNN
nChan = 100;
nData = length(w1);
slength = size(w1,2);
slength2 = size(ST1,1);


st = reshape(w1, nData, nChan, slength);
st_label = ones(nData,1);


dy = reshape(w2, nData, nChan, slength);
dy_label = zeros(nData,1);
```

```matlab
data = [st ; dy];
label = [st_label; dy_label];
%% Reshaped output - 2
for i = 1:nData
    b1(i,:) = reshape(w1(:,:,i)',1,nChan*slength);
    b2(i,:) = reshape(w2(:,:,i)',1,nChan*slength);
end


for i = 1:nData
    g1(i,:) = reshape(ST1(:,:,i),1,nChan*slength2);
    g2(i,:) = reshape(ST2(:,:,i),1,nChan*slength2);
end
```

## B.2   Best Frequency Distribution

```matlab
function [fc , ERB] = fc_ERB_hb(min_f, max_f, nChan)
% Generate the Q value plot as per [Acoustic flow...parameters]
% Constants:
fr  = 4e4;  % foveal freq. [Hz] - Change to 40 later
q0  = 10;   % ration of maximum (qr=400) to minimum (q0=10) -10dB - filter
    quality
qr  = 400;


% Q -10dB values and ERB
% Parameters:
m   = [3.850547786150128e+18 3.051156234741211e+03];    % (f<=fr   f>fr)
f0  = [min_f max_f];


% f <= fr
```

```matlab
q10_low      = @(fc)(qr-q0).*(m(1).^((f0(1)-fc)./(f0(1)-fr)) - 1)./(m(1)-1) +
    q0; %for fig
% f > fr
q10_high     = @(fc)(qr-q0).*(m(2).^((f0(2)-fc)./(f0(2)-fr)) - 1)./(m(2)-1) +
    q0; %for fig


% Calculate Step Factor
% Formulas: Q=fc/BW=fc/ERB, now, i/ERB=1/fc*Q. and step=(integration of i/ERB)
    /no of channel


% factor to adjust for 10dB bandwidth (0.5) -> 3 dB bandwidth -> ERB  (0.887):
kappa    = 2*.887;
% f <= fr
invERB_1     = @(fc)(kappa./fc).*((qr-q0).*(m(1).^((f0(1)-fc)./(f0(1)-fr)) - 1)
    ./(m(1)-1) + q0);
% f > fr
invERB_2     = @(fc)(kappa./fc).*((qr-q0).*(m(2).^((f0(2)-fc)./(f0(2)-fr)) - 1)
    ./(m(2)-1) + q0);


step_factor = (integral(invERB_1,f0(1),fr)+ integral(invERB_2,fr,f0(2)))/nChan
    ;


% Calculate the actual centre Frequency fc
% Find center frequency for each channel index nummerically:
LMAX     = 100;                % max number of iterations:
i_ch     = 0:nChan-1;
fc       = zeros(nChan,1);   % preallocate center freqs

for i = 1:nChan
  fc(i) = f0(1);           % start bisecting at f0_l
  df    = f0(2) - f0(1);
```

```matlab
    for l = 1:LMAX
        df      = df/2;            %half of df
        lb_mid  = fc(i) + df;
        if lb_mid > fr
            G_mid   = integral(invERB_2,lb_mid, f0(2))./step_factor - i_ch(i);
        end
        if lb_mid < fr
            G_mid   = ((integral(invERB_2,fr, f0(2)) + integral(invERB_1,lb_mid,
                fr))./step_factor) - i_ch(i);
        end
        if (G_mid>0)
            fc(i)   = lb_mid;
        end
        if abs(df)<=eps
            break
        end
    end
end
fc  = flipud(fc);
fc  = fc';


% split fc in fc <= fr & fc > fr:
idx_l   = find(fc <= fr);
idx_h   = find(fc >  fr);


%fc <= fr
invERB_1    = (kappa./fc(idx_l)).*((qr-q0).*(m(1).^((f0(1)-fc(idx_l))./(f0(1)-
    fr)) - 1)./(m(1)-1) + q0);
% f > fr
invERB_2    = (kappa./fc(idx_h)).*((qr-q0).*(m(2).^((f0(2)-fc(idx_h))./(f0(2)-
    fr)) - 1)./(m(2)-1) + q0);
```

```matlab
%invERB2 placed first because the number of channels are assingned from 3000
    to 1 (not 1 to 3000) in the paper
ERB       = 1./[invERB_1 invERB_2];
end
```

## B.3  Basilar Membrane Model

```matlab
function [BM_out] = BM_BOX(model, data, BM_par, fc, ERB)
% Parameters
N             = BM_par(1);
fs            = BM_par(2);
nChan         = BM_par(3);
leveldBSPL    = BM_par(4);
coef_b        = BM_par(5);
coef_c        = BM_par(6);
coef_d        = BM_par(7);
coef_e        = BM_par(8);
coef_f        = BM_par(9);
coef_g        = BM_par(10);
coef_h        = BM_par(11);
coef_i        = BM_par(12);
coef_j        = BM_par(13);
coef_k        = BM_par(14);
coef_l        = BM_par(15);
coef_m        = BM_par(16);

BM_out        = zeros(N, nChan);    % Empty BM Output Array
t             = 1/fs:1/fs:N/fs;     % Time axis
```

```matlab
if model == 1 || model == 2
    % Gammatone / Gammachirp
    data             = leveldBSPL*data*0.00014*20;   %%20 has been multiplied
        to make the amp same as DRBL
    [GT_GC_out]      = GT_GC_Func(model, t, fs, nChan, fc, ERB, coef_b, coef_c,
         data);
    BM_out           = GT_GC_out;
elseif model == 3
    % DRNL
    data          = leveldBSPL*data*0.00014;
    [DRNL_out]   = DRNL_Func(fs, nChan, fc', ERB', coef_d, coef_e, coef_f,
        coef_g, coef_h, coef_i, coef_j, coef_k, coef_l, coef_m, data);
    BM_out        = DRNL_out;
end
end
```

## B.3.1   Linear Models

```matlab
function [GT_GC_out] = GT_GC_Func(model, t, fs, nChan, fc, ERB, coef_b, coef_c
    , data)
imp      = zeros(length(t),nChan);
n        = 4;
tpt      = (2*pi)/fs;
gain     = (((coef_b^2).*ERB.*tpt).^n)./6;  %%See fast gammatone ()

for i = 1:nChan
    env      = gain(i)*(fs^3)*t.^(n-1).* exp(-2*pi*coef_b*ERB(i)*t);
    if model == 1
        carrier      = cos(2*pi*fc(i)*t);
```

```matlab
    elseif model == 2
        carrier      = cos(2*pi*fc(i)*t+coef_c*log(t));
    end
    imp(:,i)    = env.*carrier;
end
GT_GC_out = fftfilt(imp,repmat(data,1,nChan));
```

## B.3.2  Non Linear Model

```matlab
function [DRNL_out] = DRNL_Func(fs, nChan, fc, ERB, coef_d, coef_e, coef_f,
    coef_g, coef_h, coef_i, coef_j, coef_k, coef_l, coef_m, data)
for i=1:nChan
    lin_fc           = fc(i);
    lin_bw           = ERB(i);
    lin_gain         = 10.^(coef_d+coef_e*log10(fc(i)));
    lin_lp_cutoff    = lin_fc;
    nlin_fc_before   = lin_fc./1;     % 0.97%In Brian toolbox it same as lin_fc
    nlin_fc_after    = nlin_fc_before;
    nlin_bw_before   = lin_bw./(coef_f.*fc(i).^coef_g); %bandwidth of nonlinear
        part before broken stick nonlinearity. Note nonlinear bandwidth is
        always lower than linear bandwidth. defaults value is [1.17274   0.0113
        ]
    nlin_bw_after    = nlin_bw_before;
    nlin_lp_cutoff   = nlin_fc_before;

    % Broken-stick nonlinearity coefficients: a, b, c
    nlin_a  = 10.^(coef_h+coef_i*log10(fc(i)));
    nlin_b  = 10.^(coef_j+coef_k*log10(fc(i)));
    nlin_c  = 10.^(coef_l+coef_m*log10(fc(i)));     %the results is 0.25 which
```

```matlab
    satisfy Brian's Toolbox and others


% Compute gammatone coefficients for the linear stage
[GTlin_b,GTlin_a]    = FOD_GF(lin_fc, 2, fs, lin_bw);


% Compute coefficients for the linear stage lowpass, use 2nd order
    Butterworth.
[LPlin_b,LPlin_a]    = butter(2, lin_lp_cutoff/(fs/2));


% Compute gammatone coefficients for the non-linear stage
[GTnlin_b_before, GTnlin_a_before]   = FOD_GF(nlin_fc_before, 3, fs,
    nlin_bw_before);
[GTnlin_b_after,GTnlin_a_after]      = FOD_GF(nlin_fc_after, 3, fs,
    nlin_bw_after);


% Compute coefficients for the non-linear stage lowpass, use 2nd order
    Butterworth.
[LPnlin_b,LPnlin_a] = butter(2, nlin_lp_cutoff/(fs/2));


% -------------- linear part ----------------
y_lin   = data.*lin_gain;


% Gammatone filtering
y_lin   = filter(GTlin_b, GTlin_a, y_lin);


% Multiple LP filtering
lin_nlp = 4;


for j = 1:lin_nlp
    y_lin    = filter(LPlin_b, LPlin_a, y_lin);
end
```

```matlab
    % ------------ non-linear part --------------
    y_nlin  = filter(GTnlin_b_before, GTnlin_a_before, data);


    % Broken stick nonlinearity
    y_nlin  = sign(y_nlin).*min(nlin_a*abs(y_nlin),nlin_b*(abs(y_nlin)).^
        nlin_c);
    y_nlin  = filter(GTnlin_b_after,GTnlin_a_after,y_nlin);


    % Then LP filtering
    nlin_nlp    = 3;
    for j = 1:nlin_nlp
        y_nlin = filter(LPnlin_b,LPnlin_a,y_nlin);
    end
    DRNL_out(:,i)   = y_lin + y_nlin;
end
end
```

# B.4   Spike Model

```matlab
function [SRM_out, Tspikes, V] = SRM_BOX(model, IHC_out, SRM_par)
% Parameters
N           = SRM_par(1);
nChan       = SRM_par(3);
SRM_out     = zeros(N, nChan);
V           = zeros(N, nChan);
Nspikes     = zeros(nChan,1);
Tspikes     = zeros(nChan,1);
```

```matlab
if model == 1
    % Leaky Integrate-And-Fire
    for ch = 1:nChan
        [SRM_out(:,ch), V(:,ch), Nspikes(ch), Tspikes(ch)] = SRM_L_func(
            SRM_par, IHC_out(:,ch));
    end
elseif model == 2
    % SRM responses Kernel
    for ch = 1:nChan
        [SRM_out(:,ch), V(:,ch), Nspikes(ch), Tspikes(ch)] = SRM_K_func(
            SRM_par, IHC_out(:,ch));
    end
end

%{
for ch = 1:nChan
    figure(ch);
    subplot(211)
    plot(t,V(:,ch))
    str=[num2str(Nspikes(ch)),' spike(s) '];
    text(t(end)/2.5,(7/8)*(max(V(:,ch))+min(V(ch,:))),str);
    title(['Leaky Integrate-And-Fire, channel #',num2str(ch),', ',num2str(fc(
        ch)/1000),' kHz'])
    xlabel('Time [s]')
    ylabel({'Membrane','Potential','[V]'})
    subplot(212)
    plot(t,IHC_out(:,ch))
    title(['Calcium current form the IHC, channel #',num2str(ch),', ',num2str(
        fc(ch)/1000),' kHz'])
    xlabel('Time [s]')
    ylabel({'Current','[A]'})
```

```matlab
end
%}
end
```

## B.4.1 LIAF Model

```matlab
function [SpMat, V_plot, Nspikes, T] = SRM_L_func(SRM_par, I)
% Parameters
N           = SRM_par(1);
fs          = SRM_par(2);
dt          = 1/fs;              % time step [s]
t_end       = N/fs;             % signal time length [s]
V_rest      = SRM_par(4);
V_reset     = SRM_par(5);
VarTheta    = SRM_par(6);
R_m         = (1/SRM_par(7));    % membrane resistance (1/G_m) [Ohm];
Tau_m       = SRM_par(8);
V_spike     = VarTheta+abs(VarTheta-V_reset);


% Initial Conditions
i           = 1;
V           = zeros(1,N);
V_plot      = V;
SpMat       = V;
V(i)        = V_rest;
V_plot(i)   = V_rest;


% Integration: Tau*dV/dt = V_rest - V(t) + R*I(t)
Nspikes     = 0;
```

```matlab
for k = dt:dt:t_end-dt
    V_inf    = V_rest + R_m*I(i);
    V(i+1)   = V_inf+(V(i)-V_inf)*exp(-dt/Tau_m);


    if V(i+1) > VarTheta
        V(i+1)        = V_reset;
        V_plot(i+1) = V_spike;
        SpMat(i+1)  = 1;
        Nspikes     = Nspikes+1;
    else
        V_plot(i+1) = V(i+1);
    end
    i = i+1;
end


if Nspikes == 0
    T = 0;
else
    T = Nspikes/t_end;
end
end
```

## B.4.2   Kernel Model

```matlab
function [SpMat, V_plot, Nspikes, T] = SRM_K_func(SRM_par, I)
% Parameters
N              = SRM_par(1);
fs             = SRM_par(2);
dt             = 1/fs;                    % time step [s]
```

```matlab
t_end            = N/fs;                     % signal time length [s]
V_rest           = SRM_par(4);
V_reset          = SRM_par(5);
VarTheta         = SRM_par(6);
R_m              = (1/SRM_par(7));    % membrane resistance (1/G_m) [Ohm];
Tau_m            = SRM_par(8);
Tau_refr         = Tau_m*1;                  % refractory time constant [s]
Tau_rec          = Tau_m*1;                  % recovery time constant [s]
V_spike          = VarTheta+abs(VarTheta-V_reset);
Eta_0            = V_rest-V_reset;


% Initial Conditions
i                = 1;
V                = zeros(1,N);
V_plot           = V;
SpMat            = V;
V(i)             = V_rest;
V_plot(i)        = V_rest;
K_Eta            = V';
K_Kap            = zeros(N,1);
Thres            = VarTheta*ones(N,1);


% Integration: V(t) = Eta(t-ti) + int[Kappa(s).I(t-s)ds]
Nspikes          = 0;
ti(1,Nspikes+1)      = 0;
for k = dt:dt:t_end-dt
    t = k-ti(1,Nspikes+1);


    if t >= 0 && ti(1,Nspikes+1)  = 0
        K_Eta(i)      = V_rest-Eta_0*exp(-t/Tau_refr);
        Thres(i)      = VarTheta+(VarTheta-V_reset)*exp(-t/Tau_refr);
```

```matlab
        else
            K_Eta(i)      = V_rest;
        end

        if  t > 0
            K_Kap(i)      = (R_m/Tau_m)*(1-exp(-t/Tau_rec))*I(i)*Tau_m*(1+exp(-t/
                Tau_m)));
        end


    V(i)      = K_Eta(i) + K_Kap(i);

        if V(i) > Thres(i)
            V(i)            = V_reset;
            V_plot(i)    = V_spike;
            SpMat(i+1)   = 1;
            Nspikes      = Nspikes+1;
            ti(1,Nspikes+1)    = k;
        else
            V_plot(i)    = V(i);
        end
        i = i+1;
end

if  Nspikes == 0
    T = 0;
else
    T = Nspikes/t_end;
end
end
```

# Appendix C

# Machine Learning Code (Python)

## C.1  Classifier Model

```python
#Dependencies
import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, Flatten
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.callbacks import ModelCheckpoint
from tensorflow.keras.models import load_model
import scipy.io as spio
import numpy as np
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import tensorflow as tf
from random import randint

# Import Data
# load the data, split between train and test sets
X = spio.loadmat('ISI_track4_subt18_GTK_data.mat', squeeze_me=True)['data_r2']
y = spio.loadmat('ISI_16_blockIHC_track7_subt10_GCK_label.mat', squeeze_me=
    True)['label']
```

```python
#shape before resizing
print(np.shape(X))
print(np.shape(y))


#print(X[500,:,:])


#X = X[:,:,0:20]
print(np.shape(X))


print('cut shape is: ' , np.shape(X))
## Reshape correctly
#X = X.reshape(X.shape[0], X.shape[1] * X.shape[2])


print(np.shape(X))
#print(X[500,:])


seedno = randint(1, 30)


# splitting X and y into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size= 0.15,
    random_state=seedno)


## Standardize Data
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
# Fit on training set only.
scaler.fit(X_train)


# Apply transform to both the training set and the test set.
X_train = scaler.transform(X_train)
```

```python
X_test = scaler.transform(X_test)
#print(X[500,:])


# Neural network
model = Sequential()
model.add(Dense(512, input_dim=1000, activation= 'relu')) #16
model.add(Dropout(0.40)) # no drop
model.add(Dense(256, activation= 'relu'))
model.add(Dropout(0.40)) # no drop
model.add(Dense(256, activation= 'relu')) #32
model.add(Dropout(0.35))
model.add(Dense(64, activation= 'relu')) # 32
model.add(Dropout(0.35))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.30))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation= 'sigmoid'))


model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy
    '])


# simple early stopping
es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=100)
mc = ModelCheckpoint('best_ANN_model_ISI.h5', monitor='val_accuracy', mode='
    max', verbose=1, save_best_only=True)


# Train the model
history = model.fit(X_train, y_train, batch_size=128, shuffle=True,
    validation_split=0.15, epochs=500, verbose=0, callbacks=[es, mc])


plt.plot(history.history['accuracy'])
```

```python
plt.plot(history.history['val_accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['DR Train', 'DR Test'], loc='upper left')
plt.show()

plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# load the saved model
saved_model = load_model('best_ANN_model_ISI.h5')
# evaluate the model
train_acc = saved_model.evaluate(X_train, y_train, verbose=0)
test_acc = saved_model.evaluate(X_test, y_test, verbose=0)
print('train_acc', train_acc)
print('test_acc', test_acc)

from sklearn.metrics import roc_curve
X_test = tf.cast(X_test, tf.float32)
y_pred_keras = saved_model.predict(X_test).ravel()
fpr_keras, tpr_keras, thresholds_keras = roc_curve(y_test, y_pred_keras)
from sklearn.metrics import auc
auc_keras = auc(fpr_keras, tpr_keras)
print(auc_keras)
```

```python
# comparing actual response values (y_test) with predicted response values (
    y_pred)
from sklearn import metrics


#print("rf model accuracy(in %):", metrics.accuracy_score(y_test, y_pred_keras
    ) * 100)


#y_pred_test = bnb.predict(X_test)
#print("Naive_Bayes model accuracy(in %):", metrics.accuracy_score(y_test,
    y_pred_test) * 100)
#joblib.dump(cv.best_estimator_, '../../rf_cv.pkl')
plt.figure(1)
plt.plot([0, 1], [0, 1], 'k--')
plt.plot(fpr_keras, tpr_keras, label='AUC (area = {:.3f})'.format(auc_keras))
plt.xlabel('False Alarm')
plt.ylabel('Hit')
plt.title('ROC curve')
plt.legend(loc='best')
plt.show()
```