Mountain Pine Beetle Monitoring with IoT

E. Rochester, J. Ma, B. Lee, and M. Ghaderi
Department of Computer Science
University of Calgary

Abstract—**Outbreaks of forest pests cause large-scale damages, which lead to significant impact on the ecosystem as well as the forestry industry. Current methods of monitoring pest outbreaks involve field, aerial and remote sensing surveys. These methods only provide partial spatial coverage and can detect outbreaks only after they have substantially progressed across wide geographic areas. This paper presents an IoT system for real-time insect infestation detection using bioacoustic recognition via machine learning techniques. Specifically, we focus on detecting the Mountain Pine Beetle (MPB), which is the most destructive insect of mature pines in western North American forests. We present the design of the system and describe its various hardware and software components. Experimental results collected from a prototype implementation of the system are presented, which show that the system can detect MPB with 82% accuracy. We also demonstrate the applicability of our system in other noise monitoring applications, and report our experimental results on urban noise detection and classification.**

I. INTRODUCTION

A. Background and Motivation

Insect pest outbreaks (such as *Dendroctonus* Erichson, Coleoptera, and Curculionidae), through mortality and growth loss of trees, cause a large-scale damage to western North American forests [1]. Such outbreaks lead to impacts on valuable services provided by the ecosystem, *e.g.*, supply of timber, water purification, and storage of carbon. Thus, for forest pest management planning and modeling purposes, information regarding the areal extent, location and severity of the insect damage is of vital importance. While there are nine major insect pests in Canada, the *Dendroctonus ponderosae*, a.k.a. Mountain Pine Beetle (MPB), is the most destructive insect of mature pines in western North American forests [2]. Specifically, the current MPB outbreak in Canada started in the province of British Columbia in 1990s. Since then, the insect has killed an estimated 58% of the total volume of merchantable pine in the province, and is quickly spreading eastward to other provinces [3].

Currently, pest damage assessment is based on the information collected using field, aerial [4] and remote sensing surveys [5]. The aerial surveys track disturbance in the forest landscape over space and time using maps. The aerial maps are subjective, relatively coarse and require more precise surveys for pest management. Remote sensing is an extension of aerial survey, with aircrafts or satellites being equipped with sensors to allow for quantitative analysis of frequency and extent of the disturbances [6]. Remote sensing has found only limited use in the field of forest health monitoring due to cost and logistics of

acquiring the data [7]. One of the main shortcomings of current forest health monitoring approaches is that they are built around *ex post facto* detection and mitigation strategy, where the attack has already taken place in a wide geographic area.

In this work, we investigate the feasibility of using Internet of Things (IoT) for real-time and low-cost monitoring of forest health. Our goal is to design and build a Low-Power Long-Range (LPWAN) system to detect MPB infestation in early stages. While, we focus on MPB detection, the system can be used for general forest health monitoring as well as other applications in Smart City environments such as noise pollution monitoring. LPWAN technologies offer long-range connectivity for low power and low rate devices, not provided by legacy technologies (*e.g.*, cellular and short range technologies). In this work, we use LoRa [8], one of the leading LPWAN technologies that operates in the unlicensed ISM band, to provide long-range connectivity for our field IoT devices. LoRa networks are designed to provide connectivity for a massive number of IoT devices scattered over a wide geographic area, with devices communicating over distances exceeding 10 Km [9], [10]. Such characteristics make LoRa an ideal solution for remote monitoring applications such as the MPB detection considered in this work.

B. Related Work

In the following, we briefly review the works on MPB detection that are most relevant to our work.

Aerial Imaging. Currently, detection of mountain pine beetle infestation is primarily accomplished through aerial imaging techniques [11]. An aerial view, whether via satellite imaging or aerial photography, provides a means to assess and estimate threat levels of trees by examining the color of the foliage. Although extremely efficient at red-attack and gray-attack stage detection, aerial imaging is limited when applied to green-attack stage detection[1]. Reliable methods to detect green-attack stage can be accomplished by physical observations of the trees inner bark. This would involve going to the trees in question to sample and test their infestation status. However, the amount of resources and time required to perform physical observations at a large scale would be impractical.

Acoustic Sensing. Having the ability to remotely detect trees undergoing green-attack phase would vastly improve forestry and environmental agencies capability to react, control, and minimize MPB infestations [12]. An overview of recent developments in the field of acoustic tools, techniques and applications for cryptic

---

[1]Generally, the foliage fades from green to red to gray following an attack.

insect detection and distribution mapping is presented in [13]. In [14], it is discussed how to use such techniques for farm pest detection, where the system presented builds upon farm pest bioacoustic characteristics to develop an ultrasonic sensor to detect the presence of farm pests.

MPB Bioacoustics. Characterization of MPB bioacoustics is presented in [15]. The authors observed MPB acoustics during stress, male-male and male-female interactions, with sounds attaining significant energy in the ultrasound (peaks at 20, 40 and 60 kHz), but low amplitude of 55 and 47 dB sound pressure levels. They also observed that signal patterns vary among contexts primarily in the proportions of chirp types, where chirps were either simple or interrupted. This work is extended in [16], where the idea that beetles use acoustic emissions from their host tree for host selection is tested. A bark beetle species classifier is introduced in [17], which determines the identity of the signaller based on the input acoustic signal.
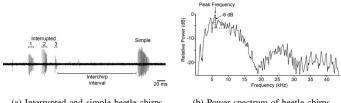
These works on characterizing the bioacoustics of MPB form the basis of our work, where we use these acoustic characterizations to build and test an IoT system for MPB detection.

C. Our Work

In this work, we describe the design and implementation of an IoT system based on LoRa LPWAN technology for MPB biacoutsic sensing and classification. We also report measurement results characterizing the performance of our system. Specifically, we make the following contributions:

- We design and build an IoT system to address the need for *proactive* forest health monitoring. Such a solution is viable due to recent development of LPWAN technologies that allow for low-power communication over long distances, *e.g.*, over 10 km with LoRa. We built a small-scale LoRa network consisting of an end device equipped with an ultrasonic microphone, a gateway that receives compressed sound samples from the end device, and a backend server that collects and analyzes the samples. The server uses different classification algorithms to decide if the received sound samples are generated by a beetle. Our lab experiments a high degree of accuracy (82% accuracy over validation samples) can be achieved using a Support Vector Machine (SVM) classifier.
- We show that the system can be applied not only for MPB detection, but also for detection and classification of urban environment noises. Specifically, we modify the system classifier using an artificial neural network (ANN), which is trained with various types of urban noises, *e.g.*, traffic at busy intersection, quiet office environment, music at the background . We show that, in the lab experiments, the system can accurately detect and classify different types of urban noise with over 90% accuracy for highly distinguishable samples and over 79% accuracy for more cryptic sound samples.

One of the key features of our design is the *local filtering* implemented on the IoT device itself. Specifically, once a noise signal is detected by the device, it is passed through a simple filter to

Fig. 1: Measured MPB bioacoustics [15].

(a) Interrupted and simple beetle chirps.   (b) Power spectrum of beetle chirps.

decide if the captured signal is worth further processing on the backend server. The local filtering helps substantially reduce the amount of energy consumed for long-range communication.

D. Paper Organization

An overview of MPB bioacoustic characteristics is presented in II, followed by a short review of LoRa technology in Section III. Our system design is presented in Section IV. Experimental results and their analysis are presented in Section V. Section VI concludes the paper.

II. MOUNTAIN PINE BEETLE

Mountain Pine Beetle (MPB) is the most destructive insect of mature pines in western North American forests. Start of the MPB infestation season ranges, depending on the overall weather conditions of the year, with warmer years seeing infestation start as early as the beginning of the spring [18]. MPBs emerge from their host tree in the search of another suitable tree to settle in. Female species of MPB burrow into the bark of the tree to burrow galleries within the bark of the tree, creating an egg storage place. Meantime male MPBs roam in the search of potential mate, lured by the pheromones emitted by female beetles [19]. Eventually, when the mate is found and successfully courted, female beetle lays the eggs in the created gallery. During this cycle, MPBs spread species of fungus within the tree, causing tree to lose its nutrients to the beetles and fungus [20]. Unless tree is able to somehow defend itself from the MPB and the fungus infestation, it inevitably dies from malnourishment.

A. Infestation Stages

There are three main stages used to categorize pine trees undergoing MPB infestation: Green-attack, red-attack, and gray-attack [21]. Green-attack stage is defined as the period where a tree may still be alive and actively combating beetle infestation as its leaves are still green in color. If the tree loses against its attackers, then leaves will remain green for a period of time until the tree falls into the red-attack stage. In this stage, the tree has lost against its invaders and the leaves turn red as nutrients steadily deplete leading into gray-attack phase. The last stage leaves the tree barren as its leaves have fallen due to the extent of deterioration already experienced.

B. Acoustic Characteristics

MPB bioacoustics are characterized using spectral, temporal and amplitude properties. In the measurement study presented in [15], it is reported that male MPB emitted chirps measured to be 55.4 and 47.1 dB sound pressure levels for 2 cm and 4 cm sound levels, respectively. In addition, the frequency of the

chirps were measured to be in the range of 6.5-75 kHz. A similar measurement study was conducted in [16] to characterize the bioacoustics of the same species, but recorded slightly different frequencies in ranges of 2.3-56.6 kHz. Based on the differences between the two results, there may be some potential variation even among the same species of beetles. According to these studies, beetles emit bioacoustics from audible to ultrasonic range. There are also two distinct patterns of chirps: simple and interrupted chirps, as shown in Fig. 1(a). These chirps are uniquely distinguishable by the pattern in chirp production, as shown in Fig. 1(b), which is visualized in the frequency domain by the relative power plot of the sound samples, as measured and presented in [15].

III. LoRa Technology

LoRa (Long Range) is an LPWAN technology developed by Semtech Corporation [22]. The LoRa design efficiently trades data rate for communication range, enabling it to be a compelling technology for large scale deployment of IoT devices over wast geographic areas. LoRa networks have three components, namely the physical (PHY) layer, link layer, and the network architecture [23]. A brief overview of each component is provided below.

A. PHY Layer

At the physical layer, LoRa implements Chirp Spread Spectrum (CSS) with integrated Forward Error Correction (FEC) [8]. LoRa networks operate in unlicensed ISM frequency band, which for North America is the frequency band $902 - 928$ MHz with center frequency of 915 MHz. For this band, the LoRa specifications define 64 non-overlapping uplink and 8 downlink channels. LoRa supports multiple spreading factors (SFs) and coding rates (CRs) for end devices. LoRa's spreading factors are orthogonal, which allow multiple devices to transmit simultaneously with different spreading factors over the same channel. The LoRa specifications for North America define four SF values, namely $\{7, 8, 9, 10\}$. Different SFs result in different transmission rates (see Table I), thus affecting transmission time of the message. LoRa also implements a form of FEC, which permits the recovery of the information in case of corruption of messages due to interference. Applying FEC requires additional coding data to be included in each transmitted packet, where the amount of coding data is determined by the coding rate. Depending on which CR is selected, one may attain an additional robustness in the presence of interference, with the available options being $\{4/5, 4/6, 4/7, 4/8\}$.

TABLE I: LoRa achievable data rates.

Data Rate	SF	bit/s
0	10	980
1	9	1760
2	8	3125
3	7	5470

B. Link Layer

LoRa MAC layer distinguishes between three end-device classes, namely class A, B, and C, where B and C class devices are required to be compatible with class A devices. Class A

Fig. 2: MPB detection system architecture.

devices are optimized for power consumption, where a device receives downlink messages only immediately after an uplink transmission, by opening two short receive windows. In addition to the two receive windows defined for class A devices, class B devices open extra downlink receive windows at scheduled times, where time is synchronized with beacons transmitted by the gateway. Class C devices, on the other hand, continuously keep the receive window open, only closing the window when transmitting.

C. Network Architecture

LoRa relies on a star topology, in which end devices directly communicate with a few *gateways* in a single-hop manner. The gateways are in turn connected to a network server and use TCP/IP protocols to communicate with the server. Each end-device may adjust its data rate manually or using adaptive data rate (ADR) [23]. Since end devices broadcast their messages, the same message may be received by multiple gateways who will forward the message to the network server, where the redundant messages are filtered. Within this network architecture, the network server is also responsible for security, diagnostics and, if so desired, acknowledgements [23].

IV. System Design

The conceptual architecture of the system for MPB detection is presented in Fig. 2. In the following, we will describe the hardware and software components of the system, and describe how the actual detection is implemented.

A. Hardware Specifications

The system consists of an IoT end device and a LoRa gateway, as depicted in Fig. 3. The end device consists of a micro-controller that drives a LoRa transceiver module for network communication and an ultrasonic microphone to capture sound samples.
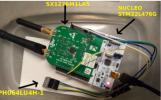
Lora Gateway. The gateway is a custom-built LoRa gateway operating in the 915 MHz ISM band. It is powered by Raspberry Pi 3 Model B that runs Raspbian operating system and is connected to a certified 8 channel concentrator board (see Fig. 3(a)). We use the free crowd sourced network server hosted by The Things Network (TTN) [24] and then use the MQTT feeds to retrieve the device data from the server.

IoT Device. The end device consists of three components, namely:
- NUCLEO-STM32L476G micro-controller,
- SX1276MB1LAS LoRa shield,
- SPH064LU4H-1 ultrasonic microphone.

Beetles communicate via ultrasound frequencies, *i.e.*, above 20 kHz. Thus, special hardware is required to capture their communications. The off-the-shelf microphones are predominantly

(a) Raspberry Pi LoRa gateway.   (b) IoT device with LoRa shield.

Fig. 3: Hardware components of the system.

(a) Arduino audio player.   (b) Audio player circuit schematics.
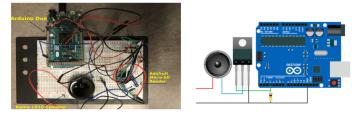
Fig. 4: Audio player with Kemo L010 speaker.

built to support sound waves of only up to 20 kHz frequency. The SPH064LU4H-1 microphone is capable of capturing sound waves from 100 Hz to 80 kHz [25], which is more than enough to cover the frequency spectrum of high pitched beetle chirps. The SX1276MB1LAS shield is fitted with the SX1276 transceiver, which includes the LoRa long range modem. The application running on the NUCLEO-STM32L476G micro-controller enables us to collect sound samples from the microphone, and then filter and compress them before transmission to the backend server for further analysis and classification.

B. Software Specifications

Various software modules used to build the monitoring and detection applications, running on the IoT device and backend server, are described next.

Monitoring Application. Our main goal for the end device was to develop a working low-power node to capture ultrasound signals, which could be installed in a remote forest location for real-time monitoring. To this end, we used the STM library [26] to develop a monitoring application on the end device. The application acquires the pulse density modulation of the microphone signal using a digital filter for Sigma-Delta modulators interface function of the microphone itself, and outputs its frequency characteristics using FFT. It then runs a simple filtering on the acquired output based on a threshold to decide if the microphone output should be transmitted to the backend server for further analysis, or discarded. In total the microphone collects 1022 data points, to cover the frequencies from 6.5 to 75 kHz. The data points are converted from the original 32 bit floats to 8 bit integers to reduce the size of samples. Then, they are passed through a compression algorithm, which on average compressed every data sample to 80% of its original size.

Communication Module. In order to send the samples to the backend server, the device uses the LoRaWAN library for STM32 device series found in [27]. The large amount of data points meant that, due to the LoRaWAN maximum packet size restriction of 242 bytes [23], we divided the samples into 4 packets which were sent back-to-back. Because of long transmission times, the application is implemented as a multi-threaded application that is able to sample the environment and send packets simultaneously. Specifically, one thread is used for data sampling and compression, with abnormal data (*i.e.*, data that passed the basic threshold filter for further processing) after compression being added to a circular queue. As soon as the queue is full, sampled data will be discarded. Another thread

is used to continuously check if there is any data available for transmission, fetch it from the queue and transmit it.

Classification Algorithm. After the sound samples are received by the TTN server, we use MQTT protocol to retrieve and save the samples in a database. The samples are then fed into our classifier to detect the type of the sound signal. Since in MPB detection, only one specific type of sound needs to be detected, we used the one-class support vector machine (SVM) in the Scikit-learn machine learning library [28]. The detection results are then visualized in a web portal created using the Django framework [29].

V. MEASUREMENTS AND EXPERIMENTS

In this section, we present a summary of our experiments with the proof-of-concept prototype system. The experiments were conducted in a lab environment, where it was possible to control the amount of background noise affecting the system. Two sets of experiments were conducted: (i) MPB detection experiments, and (ii) urban noise detection and classification experiments.

A. MPB Detection

In order to reproduce MPB chirps, we designed a custom audio player platform for generating and playing sound samples in the ultrasound frequency range. The sound samples are generated in accordance with the MPB acoustic characteristics identified in [15]. The reason for designing a custom player platform is that typical off-the-shelf speakers (*e.g.*, those in home audio systems) cut out at frequencies lower than what is needed to reproduce beetle communication sounds. There are some ultrasound speaker options, but such speakers and audio cards are usually custom-built and come at a steep price. Not only our Do-It-Yourself (DIY) audio player system is low cost, it can also be operated on batteries as it is built on an Arduino platform. This means that, in the future experiments, we can take the player and the IoT device to the wild for testing and measuring in a more realistic environment.

Audio Player. The playback system is based on an Arduino DUE, equipped with an external SD card reader, which is attached to a Kemo L010 speaker (a low-cost piezo speaker) able to generate ultrasound frequencies of up to 60 kH. Since Arduino DUE can only output signals at 5 V, the system includes a metal-oxide-semiconductor field-effect transistor (MOSFET) to control the speakers with higher voltage than that viable from the Arduino pins, through an external 120 VAC/16 VDC power source. The resistor is a pull-down resistor used to lower the
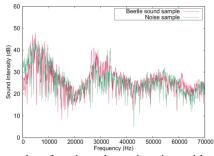
Fig. 5: Examples of random ultrasonic noise and beetle sound.

Fig. 6: ANN classification results for urban noise.

voltage to ground the line if Arduino is not driving it up. The schematics of the speaker control circuit are shown in Fig. 4(b).
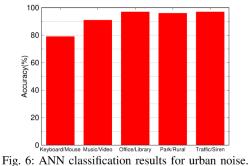
Sound Generation. To create artificial beetle chirps, we used the Python's wave library [30] to create audio files containing MPB-like sounds. We replicated the sound signatures by adding together multiple sine waves with different frequencies. The generated audio files needed to have frequencies of up to 60 kHz, thus the sampling rate of the audio was set to 196 kHz. Clipping technique was used to further increase the sound intensity of the audio. After the necessary sound files were generated, they were uploaded to the micro-SD card and played on the Arduino using the circuit shown in 4(a). On the Arduino micro-controller, the libraries Audio [31] and SD [32] were used to read and play the wave files stored on the SD card.

Beetle Detection. For beetle sound detection and classification, 700 sound samples of beetle noise were played through the Arduino player and were then captured by the IoT end device. Also, around 100 samples of non-beetle ultrasound noise were generated and collected as well. The latter were used for validation purpose. The SVM model was defined using degree 3 polynomial kernel and parameter $\nu = 0.01$ for lower bound of the fraction of support vectors. The SVM was tuned using the first 100 beetle sound samples as the training data. The remaining samples were then used as test data. With these parameters, we reached an accuracy of 82% for correctly detecting beetle sound samples out of the mix. Fig. 5 shows an example sound sample for beetle noise as well as non-beetle ultrasound noise.

Baseline Classifier. To create a base line for comparison of our results, on the same sound samples we used a non-machine learning classification approach. In this approach, we average the data points at the same frequency in all training samples and then use the average as the reference classification value. Specifically, for classification, we calculate the absolute difference between an unclassified sample sound level and the average reference value. If the difference is smaller than a pre-specified threshold (in our experiments, we chose the threshold to be 0.1), the sample is classified as beetle sound. This approach achieved an accuracy of 70% for the same sample set. As shown in Table II, SVM approach achieves higher accuracy for detecting and classifying MPB sound samples.

TABLE II: MPB sound classification accuracy results.

Classifier	Accuracy(%)
SVM	82%
Baseline Classifier	70%

B. Urban Noise Classification

Our system, while initially designed for beetle detection, can be used in other sound monitoring applications such as urban noise pollution monitoring. Urban noise can be defined as unwanted loud or annoying sound from various sources. It has harmful effects on the health and well-being of urban citizens. Individuals exposed to urban noise sometimes experience sleep disturbance and performance impairment [33]. Long time exposure to severe noise pollution causes progressive hearing loss, mental disorder, high heart rate and blood pressure which can potentially lead to cardiovascular diseases [34]. A system for real-time data collection and analysis over wide geographic area helps the city planners gain insights into the severity of urban noise pollution at different times and locations.

TABLE III: Sound distribution among different noise classes.

Sound Class	# of sound samples (%)
Office/Library	25.3
Traffic/Siren	24.5
Park/Rural	18.4
Music/Video	18
Keyboard/Mouse	13.9

Experiment Setup. For urban noise classification, there was no need to collect data in the frequencies above 20 kHz, since that is the limit of human hearing. As such, we collected only 240 data points per sample in the range from 100 to 17.5 kHz, transmitting them to the network server only if a certain threshold of sound intensity was surpassed. For classification, an Artificial Neural Network (ANN) was built using Tensorflow, an open source machine learning framework [35]. The constructed ANN was a multilayer perceptron with 2 hidden layers, where each layer had a Sigmoid activation function. Input layer had 240 nodes, corresponding to the number of data points per sample. Each hidden layer had the same number of nodes and output size of 5, where sound samples were classified into one of 5 urban sound categories. The categories/sound environments used in this experiment are: Traffic/Siren, Park/Rural area, Music/Video, Keyboard/Mouse, Office/Library. A total of 1219 sound samples were collected and used for ANN training. Table III shows the noise sample distribution among different classes. The variability in the training data samples came not only from playing different sound samples, but by varying the distance between the speaker and the microphone.

Noise Classification. The initial results for noise classification

had below 50% accuracy, *i.e.*, worse than random guessing, with misclassification of samples happening on a deterministic basis. We identified two reasons for such a low classification accuracy. The first reason was the similarity of sound acoustic signatures. The input to the ANN includes sound intensity levels at different frequencies of the sound sample. Thus, there are cases when two distinct and unique sources generate a sound sample with the same intensity at similar frequencies. As such, it would be impossible to distinguish between the two sources. For this reason, we assumed that such similar noises would be categorized into the same category. The second reason was the present of samples, in which multiple noises were present simultaneously. When used for training, these samples result in low classification accuracy, and when used for validation, they result in misclassifications. As such, in this experiment, we manually dealt with such sound samples. After we modified our system to account for the above sample types, the average accuracy for the validation set reached 92%. Fig. 6 shows the classification accuracy for each of the noise categories. Fig. 7 shows a snippet of the user interface for the Django application that visualizes the collected and classified data results.

Fig. 7: Data visualization and classification portal.

VI. CONCLUSION

In this paper, we presented the design and evaluation of an IoT bioacoustic monitoring system for MPB detection. Experiments conducted in a lab environment confirmed the potential of such a system for both MPB detection and urban noise pollution monitoring. Specifically, we observed that the current system is very effective in detecting a specific noise signature with a unique acoustic signature (achieved accuracy of 82%) which makes it a promising solution for MPB detection. In the future, we plan to conduct further experiments, first in the lab with live beetles, and subsequently in the wild.

REFERENCES

[1] V. A. Saab *et al.*, "Ecological consequences of mountain pine beetle outbreaks for wildlife in western north american forests," *Forest Science*, vol. 3, 2014.

[2] J. White *et al.*, "Detecting and mapping mountain pine beetle red-attack damage with spot5 10- m multispectral imagery," *BC Journal of Ecosystems and Management*, vol. 7, 2006.

[3] N. R. Canada, "Mountain pine beetle (factsheet)." [Online]. Available: https://www.nrcan.gc.ca/forests/fire-insects-disturbances/top-insects/13397

[4] C. C. of Forest Ministers, "Forest pest monitoring in canada: Current situation, compatibilities, gaps and proposed enhanced monitoring program," Accessed: December, 2018. [Online]. Available: http://www.ccfm.org/pdf/PestMonitoring_2012_en.pdf

[5] R. Hall *et al.*, "Remote sensing of forest pest damage: A review and lessons learned from a canadian perspective," *The Canadian Entomologist*, vol. 148, 2016.

[6] N. G. McDowell *et al.*, "Global satellite monitoring of climate-induced vegetation disturbances," *Trends in Plant Science*, vol. 20, 2015.

[7] C. Rullán *et al.*, "Remote monitoring of forest insect defoliation," *Forest Systems*, vol. 22, 2013.

[8] LoRa Alliance. [Online]. Available: https://www.lora-alliance.org

[9] A. M. Yousuf *et al.*, "Throughput, coverage and scalability of LoRa LPWAN for Internet of Things," in *in Proc. IWQoS*, 2018.

[10] ——, "A low-cost LoRaWAN testbed for IoT: Implementation and measurements," in *in Proc. WF-IoT*, 2018.

[11] C. Senf *et al.*, "Remote sensing of forest insect disturbances: Current state and future directions," *Int J Appl Earth Obs Geoinformation*, vol. 60, 2017.

[12] M. A. Wulder *et al.*, "Challenges for operational detecion of mountain pine beetle green attack with remote sensing," *The Forestry Chronicle*, vol. 85, 2009.

[13] R. Mankin *et al.*, "Perspective and promise: a century of insect acoustic detection and monitoring," *American Entomologist*, vol. 57, 2011.

[14] A. S. R. M. Ahouandjinou *et al.*, "Smart environment monitoring system by using sensors ultrasonic detection of farm pests," in *in Proc. BioSMART*, 2017.

[15] A. Fleming and Lothers, "Acoustics of the mountain pine beetle (dendroctonus ponderosae) (curculionidae, scolytinae): sonic, ultrasonic, and vibration characteristics," *Canadian Journal of Zoology*, vol. 91, 2013.

[16] M. Kaiser, "Acoustics of mountain pine beetle and lodgepole pine," *Masters Thesis. University of Calgary.*, 2014.

[17] A. Lindeman, "Acoustic signalling in the destructive bark beetle genus dendroctonus (curculionidae:scolytinae) with emphasis on dendroctonus valens," *Doctoral Thesis, Carleton University.*

[18] J. R. Safranyik, L., "Emergence patterns of the mountain pine beetle from lodgepole pine," *Canadian Forest Service*.

[19] C. C. Chiu *et al.*, "Monoterpenyl esters in juvenile mountain pine beetle and sex-specific release of the aggregation pheromone trans-verbenol," *Proceedings of the National Academy of Sciences*, vol. 115, 2018.

[20] L. Safranyik *et al.*, "An interpretation of the interaction between lodgepole pine, the mountain pine beetle and its associated blue stain fungi in western canada," *D.M. Baumgartner, ed. Management of lodgepole pine ecosystems.*, 1975.

[21] L. . N. R. O. Ministry of Forests, "Biology of the mountain pine beetle," Accessed: December, 2018. [Online]. Available: https://www2.gov.bc.ca/assets/gov/farming-natural-resources-and-industry/forestry/forest-health/bark-beetles/biology_of_the_mountain_pine_beetle.pdf

[22] Semtech Corporation. [Online]. Available: http://www.semtech.com/

[23] Semtech, "LoRaWAN specification v1.0.2." [Online]. Available: https://www.lora-alliance.org/technology

[24] The Things Network. [Online]. Available: https:/thethingsnetwork.org

[25] P. D. sheet: SPH0641LU4H-1. [Online]. Available: https://www.mouser.com/ds/2/218/-746191.pdf

[26] L. software expansion for STM32Cube. [Online]. Available: https://github.com/y2kblog/NUCLEO-L476RG_DFSDM_PDM-Mic

[27] [Online]. Available: https://www.st.com/en/embedded-software/i-cube-lrwan.html

[28] S. learn Machine Learning in Python. [Online]. Available: https://scikit-learn.org/

[29] D. W. Framework. [Online]. Available: https://www.djangoproject.com/

[30] P. D. wav. [Online]. Available: https://docs.python.org/2/library/wave.html

[31] A. D. Audio. [Online]. Available: https://www.arduino.cc/en/Reference/Audio

[32] A. D. SD. [Online]. Available: https://www.arduino.cc/en/Reference/SD

[33] D. Halperin, "Environmental noise and sleep disturbances: A threat to health?" *Sleep Science*, vol. 7, 2014.

[34] S. A. Stansfeld and M. P. Matheson, "Noise pollution: non-auditory effects on health," *British Medical Bulletin*, vol. 68, 2003.

[35] "Tensorflow website." [Online]. Available: https://www.tensorflow.org/