Media Engineering and Technology Faculty
German University in Cairo

Intelligent Pillow

Bachelor Thesis

Author: Mariam El-Sherif
Supervisors: Dr. Wael Abouelsaadat

Submission Date: 30th July, 2020

Media Engineering and Technology Faculty
German University in Cairo

Intelligent Pillow

Bachelor Thesis

Author:	Mariam El-Sherif
Supervisors:	Dr. Wael Abouelsaadat

Submission Date: 30th July, 2020

This is to certify that:

(i) the thesis comprises only my original work toward the Bachelor Degree

(ii) due acknowlegement has been made in the text to all other material used

Mariam El-Sherif
30th July, 2020

Acknowledgments

I would like to begin by offering my sincere thanks to Dr. Wael Abouelsaadat for his guidance. Despite the challenging circumstances, he has managed to offer direction and valuable advice at every stage.

I would like to express my deepest gratitude to my mother, who always brings tranquility to my life. To Nadia, who encourages me to achieve things I wouldn't believe possible. To my grandmother, who inspires me to work hard and appreciate the small pleasures in life. To my beloved partner, who loves and supports me unequivocally.

Finally, I would like to acknowledge my friends and colleagues. Thank you for coming on this journey with me.

Abstract

Hearing impaired individuals are unable to respond to auditory alerts during emergency situations, making them vulnerable to danger. This project introduces a solution in the form of an intelligent hearing pillow. The pillow listens to the room during sleeping hours and detects when a noise indicating danger occurs, such as a fire alarm. It responds by producing haptic feedback in order to rouse the user. It contains an embedded system connected to an external device running the main algorithm. The system's microcontroller unit (MCU) transmits the sound, which is detected using a microphone as a sensor. The haptic feedback is produced by a vibration motor connected to the MCU. The pillow's algorithm utilises an artificial neural network (ANN) model to classify the sounds in its environment. The machine learning algorithm is able to classify each sound into specific categories with an accuracy of 85.7%.

Contents

Chapter 1

Introduction

1.1 Thesis Motivation

House fires occur most often during the evening hours, when home residents are sleeping [1]. A reliable alert system is therefore essential in order to avert danger to residents of the household. Alerts for an emergency situation, however, often take the form of an audible warning: sirens, commands, alarms. This causes a significant segment of the population to be helpless in case of an incident.

Over half of the male population over the age of 65 suffers from hearing impairment [2]. During waking hours, a variety of options exist for alerting a person without involving sound. For instance, their phone may deliver haptic feedback by vibrating in response to an alert sent via text message. However, when a person is asleep, the brief, distant vibration of their phone may not be enough to rouse them. As a result, a more dependable option is needed to guarantee safety for a hearing impaired individual.

1.2 Thesis Aim

This project seeks to improve emergency response among the hearing impaired population by ensuring that should an incident occur while they are sleeping, they can be adequately roused and notified. It takes the form of a pillow that is able to capture the sounds in the room, and vibrates when there's a loud, unfamiliar noise. When an alert is sounded to warn people about an emergency, the pillow will react to the noise by vibrating until the alert ceases. The vibration's purpose is to wake the user. This enables them to respond to the alert as needed, helping ensure their safety.

1.3 Thesis Organisation

This thesis consists of five chapters, including the introduction. Each chapter tackles one stage of the project's development.

Chapter 2: Background and Related Works lays the foundation of the project. Section 2.1 discusses the nature of hearing impairment, and goes on to explore machine learning technology. Section 2.2 examines some projects that share either purpose or theme with this one. First, it explores other technologies developed to assist the hearing impaired population. After that, it goes on to discuss other technologies that use a pillow as a medium.

Chapter 3: Implementation discusses the process of gathering and refining data later used to train the machine learning algorithm. Section 3.1 introduces the vision for the full design and implementation. Section 3.2 describes the process of selecting audio files to create a training dataset, and Section 3.3 explains how these audio files were edited to mimic the sounds produced by a person's environment during a night of sleep.

Chapter 4: Algorithm demonstrates how the machine learning algorithm responsible for the classification of the dataset was created. Section 4.1 identifies and defines the features extracted from each audio file in the dataset and explains how these features are extracted. Section 4.2 discusses the machine learning model. It illustrates the type of model used as well as outlining the other tools that were required to build it.

Finally, Chapter 5: Conclusion concludes the thesis and displays its results. Section 5.1 reveals the outcomes of the project. Section 5.2 discusses the limitations the project. Lastly, Section 5.3 recommends areas for further research.

Chapter 2

Background and Related Works

2.1 Background

This section introduces two key topics that establish the foundation of this project. It discusses the definition and varieties of hearing impairment. It also briefly defines machine learning, which is further discussed in Chapter 4.

2.1.1 Hearing Impairment

As illustrated by Morton [2], hearing impairment occurs in at least half of males over 65 years of age. It's defined as any loss of hearing exceeding 25 decibels (dB) [2]. The following are some ways it is categorised:

- It may be described as prelingual, occurring before a child has learned to speak, which may impact speech, or postlingual, occurring after the acquisition of language.
- It may be described as conductive, caused by a blockage in the outer ear, often due to a rupture of the eardrum, or sensorineural, caused by a problem in the inner ear.
- It may be described as genetic, caused by heredity, or acquired, caused by the environment.
- It may be referred to as significant impairment, which occurs at 25 dBs, or profound deafness, which occurs at over 80 dBs.

The incidence of hearing impairment at birth has been reduced in recent years, due to the gradual reduction of inbreeding. However, electronic noise pollution and presbycusis, which refers to hearing loss that occurs gradually as one ages, are increasing as the population ages, increasing the frequency of late-onset hearing impairment.

Late-onset hearing impairment's incidence increases almost 36-fold from age 20 to 80, likely due to heredity, presbycusis, infection, and environmental noise pollution. It can, however, often be ameliorated by a hearing aid, unlike prelingual hearing impairment, which is commonly more difficult to treat.

2.1.2 Machine Learning Technology

According to Mahana [3], machine learning is a technology aimed towards mining information from data by learning to recognise patterns and make intelligent decisions based on that data. Therefore, it mainly involves the design of algorithms that enable computers to evolve based on data.

There are three general types of machine learning, listed below:

- Supervised learning, which involves running training data pre-labelled with intended results to give feedback on learning progression.
- Unsupervised learning, which has no particular desired output, but attempts to find hidden structure in unlabelled data.
- Reinforcement learning, which teaches the machine to learn behaviour from the feedback given by its environment.

Audio signal classification, which is the ultimate goal of this project, aims to extract certain features from an audio file and use those features to pinpoint its class. This is generally a pattern recognition problem. Pattern recognition refers to the scientific discipline whose aim is to classify objects into categories [3].

Classification is a function of data mining that falls under supervised learning. It assigns items in a group to particular categories or classes. It is, therefore, an example of pattern recognition. The aim of this discipline is the design of a classifier. A classifier is an algorithm able to take features of an object as input, and outputs the label indicating which class the object should belong to. This process is shown in Figure 2.1

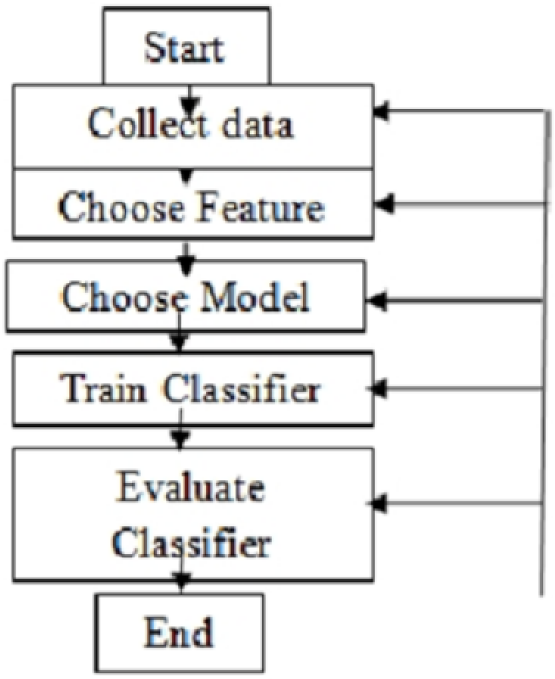


Figure 2.1: A Classifier's Design Cycle (from [3])

The first step of the process is collecting sufficient data in order to successfully train the classifier. This amount varies depending on the complexity of the data. Then, the classifier must have a particular feature to classify. This feature will be the variable when the classifier is being evaluated. Next, a machine learning model needs to be chosen. There exists a variety of machine learning models for different purposes. Therefore, the model is chosen based on the chosen dataset and feature.

Once these decisions have been made and the data prepared, the model can begin training. The amount of time a model takes to train will vary depending on the size of the dataset. When the training is complete, the model can be tested using a testing dataset. A testing dataset is a set of classified data that is run through the classifier to evaluate its accuracy and overall performance in a controlled environment. Accuracy can be improved by further training the classifier using a larger dataset.

2.2 Related Works

This section discusses other projects that share either purpose or medium with this project. Subsection 2.2.1 examines other technologies developed for the purpose of assisting the hearing impaired population. Subsection 2.2.2 analyses examples of projects which used a pillow as a medium to create a biomedical device.

2.2.1 Assistive Technologies for Hearing Impaired

There are many examples of technologies developed with the aim of assisting hearing impaired individuals perform every day tasks. Some examples are outlined below.

Haptic Communication Robot for Urgent Notification

Furuhashi [4] developed a robot inspired by hearing dog behaviour. A hearing dog will alert its owner to an emergency signal by bumping into them to get their attention. If the owner is asleep, they will continue to do this until they wake up. The robot attempts to mimic this behaviour by using wheels to approach and bump into the user.

Upon hearing an alert, it immediately locates the user and proceeds to bump into them. If the user is currently on a chair or bed, the robot will bump into it instead. It offers an improvement upon the previous methods for haptic notification, such as mobile tools vibrating nearby. The robot can produce more haptic feedback than a smaller device would be able to achieve, making it more likely to direct the user's attention to the alert promptly.

Figure 2.2: Design of Furuhashi's Hearing Dog (from [4])

Hearing Robot Using Visual Communication Signals of Hearing Dogs

Similar to the previous technology, Koay [5] developed a hearing robot inspired by the behaviour of hearing dogs. However, in addition to using haptic feedback to alert the owner of an alert, it can also lead the owner to the source of the sound using behaviours of hearing dogs that imply intent.

The design of robots usually focuses on mimicking human-human interaction. As a result, robots are taught how to imitate human behaviour. In this instance, however, using human-dog interaction as an inspiration is beneficial because humans generally attribute intentions to dogs when interacting with them. Therefore, dogs are not expected to explicitly state their intentions using verbal cues.

Hearing dogs' behaviour when a doorbell rings usually follows this general layout:

- First, the dog will try to attract its owner's attention by touching the person, gently at first. If unsuccessful, they will continue nosing or pawing at them until they've gotten their attention. This behaviour is referred to as attention seeking.

- Then, the dog will lead the person to the source of the sound by walking towards it. Periodically, the dog will look back to confirm the owner is still following. If the owner is currently looking at the dog trying to understand its intent, the dog will walk ahead towards the target while looking back in an attempt to communicate its intent. If the owner has stopped paying attention and is distracted by something else, the dog will return to the owner and try to attract their attention again, using the attention getting behaviour described in the first step. The dog will continue to cycle through these actions until the owner has reached the desired target and then it'll stop nearby. This behaviour is known as leading.

- Finally, once the owner has arrived at the target location, the dog will direct the owner's attention towards the target object by looking at the owner then at the object to indicate its desire for the owner to interact with this object. This behaviour is known as gaze alternation.

Using these guidelines, Koay developed a behavioural strategy flowchart, shown in Figure 2.3. In addition to the hearing robot's logic, visual behaviour sequences were also added. The hearing robot is designed to employ attention seeking behaviour by physically interacting with the owner to draw attention, looking back with physical cues as an aide to leading the owner towards a target, and gaze alternation to ensure the owner can interpret the hearing robot's target.

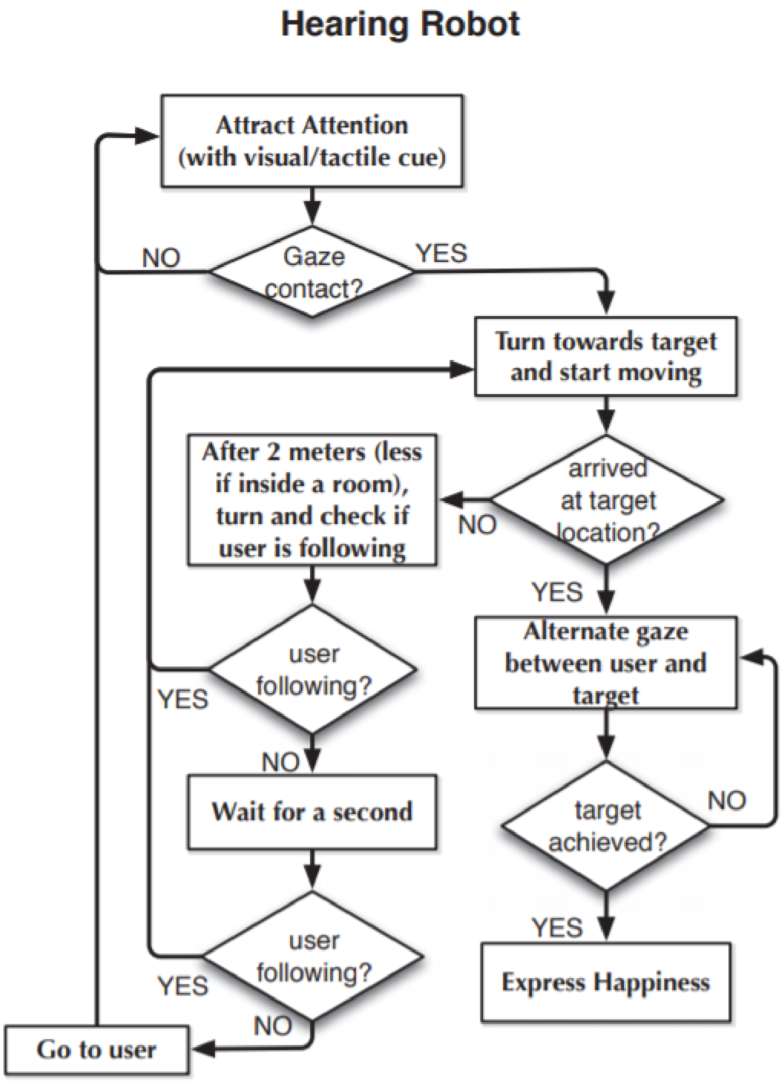


Figure 2.3: Flowchart Describing Dog-Inspired Behavioural Strategy (from [5])

The steps illustrated in Figure 2.3 can be grouped into three key sequences. The first sequence is attention getting, using visual or tactile cues. The second sequence is leading, by moving ahead of the owner and checking if they're following. Finally, the third sequence is showing target, done by showing target object, and then performing gaze alternation.

VisualLink

Hong [6] developed a software that allows elderly people with hearing impairment to receive phone calls. It uses a basic interface to alert the user to a call, transcribe the call's text using speech recognition, and transmit multimedia to the elderly on a large screen such as a television.

The device alerts the user to a call by displaying an alert on a television screen, as well as vibrating using a haptic device attached to their clothing. When the user answers the call, it begins to transcribe the caller's words on the television screen for the user to read. The caller can include visual multimedia in advance to be displayed during the call, or they can add it while already in the call. This process is illustrated in Figure 2.4.

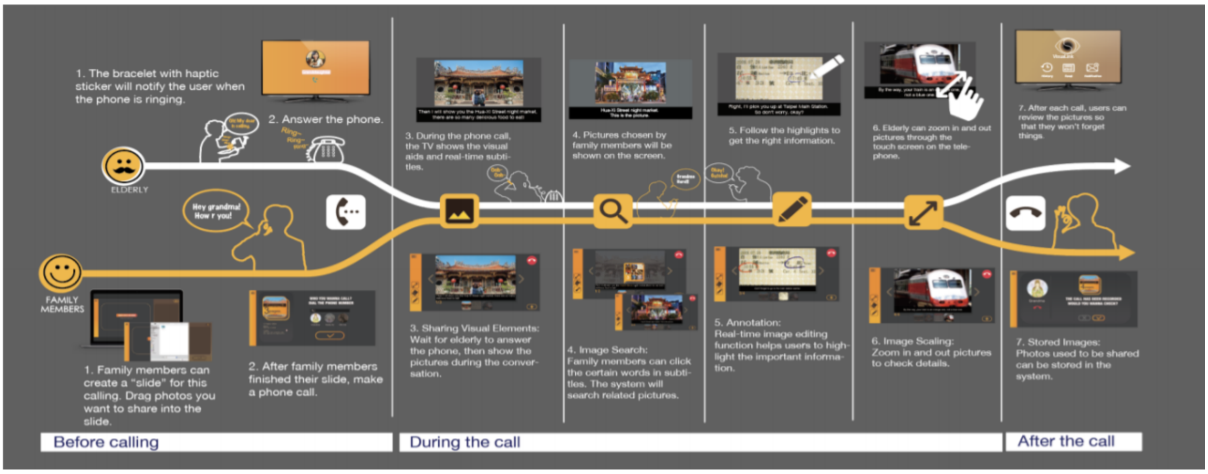


Figure 2.4: VisualLink's Behaviour Before and During a Call (from [6])

The three main components of the system are:

- A conferencing system reminiscent of a traditional telephone. It uses speech recognition to capture the words of the caller and caption them. It also directs the call, subtitles, and received multimedia to the television.
- A sticker which can be attached to a watch or bracelet. Its role is the delivery of haptic feedback when a user is being alerted of a phone call.
- A call authoring interface which allows the caller to select visual multimedia to be added to play either immediately or later on during the call.

Mobile Tool for Hearing Impaired at Emergency Situation

This mobile application, developed by Hosono [7], was created to assist deaf, language impaired, or non-native people when reporting emergencies. It was proven to be three times quicker than using text messages to report a similar emergency.

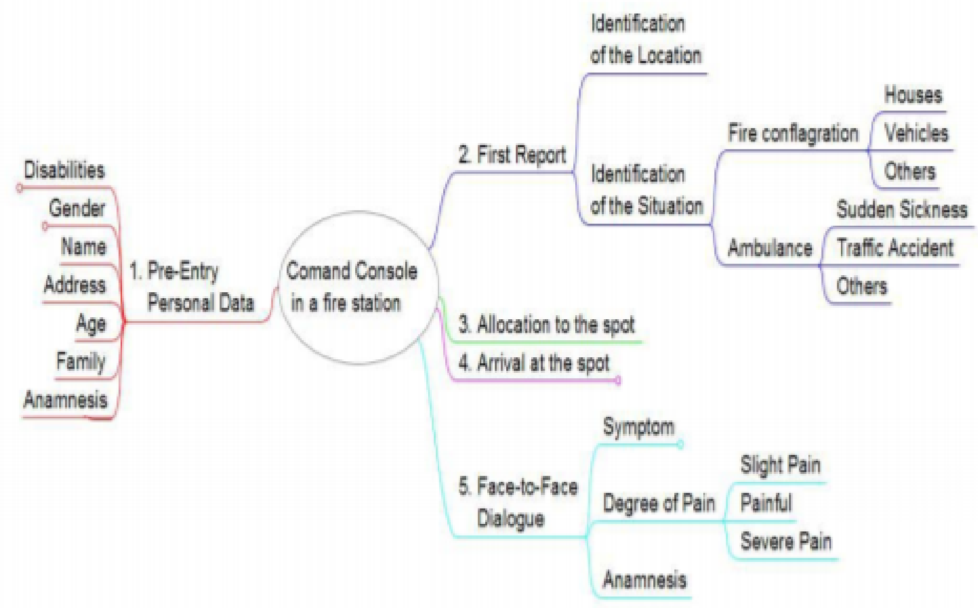


Figure 2.5: Dialogue Process at Emergency Situation (from [7])

The human-centred design process focused on pre-existing emergency dialogue. The emergency dialogues to a local fire station were observed and used to chart the process of an emergency dialogue, shown in Figure 2.5.

The application consists of five main interfaces. The first interface is the pre-registration interface, which acquires the user's information to register within the system prior to the occurrence of an emergency situation.

The second interface is the exact moment of emergency report. This interface aims to report immediately to the nearest fire station.

The third interface is the dispatch of a fire engine or ambulance to location. During dispatch, the users can communicate with the application while they wait for the emergency vehicle's arrival.

The fourth interface is the approach to the reported location. This sends the user visual and auditory signals to alert them that the emergency vehicle is nearing arrival to their location.

The fifth and final interface is dialogue on the spot. This interface is used as a face-to-face interaction tool to facilitate communication with emergency personnel who are unfamiliar with sign language. The user can use icons or pictograms to assist with communication with the emergency workers.

2.2.2 Pillow-based Devices

Different approaches have been explored in the endeavour towards an intelligent pillow, predominantly for biomedical purposes as outlined below.

Mimo Pillow

Chen [8] created a soothing pillow for neonates called the Mimo pillow. Its purpose was to provide a way for medical procedures to be performed that may be uncomfortable or painful without causing great distress to the infant. To achieve this, the infant's mother records her heartbeat using a simple pulse oximeter, which is then transferred to the pillow. When placed in the incubator with the infant, the pillow need only be turned on to assist the procedure by beating gently, simulating the mother's heartbeat. This proved to be an improvement on the current researched methods for soothing neonates, which can be difficult to achieve within the confines of an incubator. The pillow is made of a material that is easy to clean and comfortable, for ease of use in the context of a hospital.

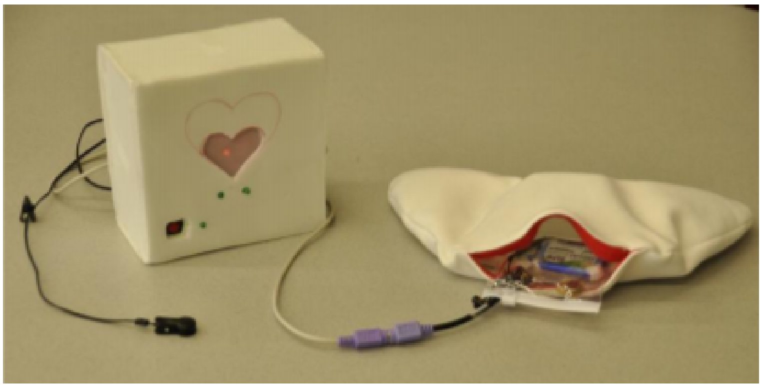


Figure 2.6: Mimo Recorder (left) and the Mimo Pillow (right) (from [8])

Intelligent Pillow With Maternal Temperature Monitor

A similar project by Nandhini [9] sought to develop a pillow to soothe neonatal infants by improving upon the Mimo pillow above. In addition to the heartbeat vibrations, Nandhini's pillow also simulated the mother's basal body temperature. This helps mimic the infant's sensation of being held by its mother during stressful procedures at times when the mother could not be present due to incubation restrictions. The pillow creates the simulation by taking the mother's temperature in addition to the heartbeat recording, and then adjusts the pillow's temperature whenever it differs from the reading received from the mother.

Sensor Pillow System

Several other pillows were developed with the purpose of monitoring hospital patients' health. Those either paved the way for further research that created ways to react to the readings, or were used simply for ease of doctor-patient interaction.

Among the earliest of these health monitoring pillows is Harada's [10] sensor pillow system, designed to track the rate of respiration and body movement during sleep. He used a sheet of pressure sensors below the pillow attached to an interface box to detect all movements during sleep, shown in Figure 2.7. He then analysed the pattern of movement that occurs during inhalation and exhalation to derive the rate of respiration. Other, larger movements were counted as turns. When compared to counting turns using a video recording, the pillow detected 20 out of 23 and 9 out of 10 turns in two different experiments. Respiration was similarly tested by comparison to a medical device. Where the sensor pillow detected 5600 breaths, the medical device recorded 5479 in one experiment, while the other experiment counted 3130 and 3167 respectively. This level of accuracy without the need for medical devices that may restrain the patient was an unprecedented achievement of this research.

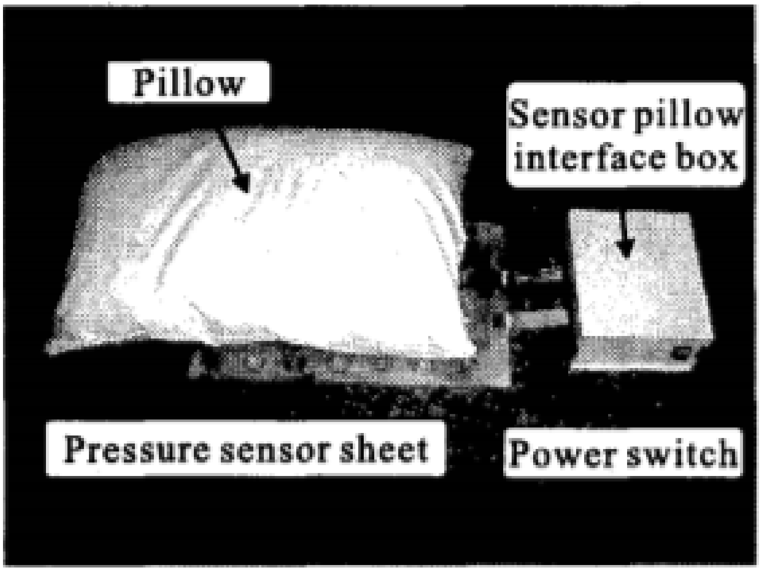


Figure 2.7: Structure of Sensor Pillow System (from [10])

Cardio-Respiratory Monitor Pillow

A similar health monitoring pillow by Lokavee [11] expanded upon the previous work by adding a cardio-respiratory function. Additionally, the pillow could now monitor several different postures as opposed to simply count turns. It used a similar pressure sensor system, but integrated into the pillow itself. It transmits the data read from the pillows wirelessly to an external device via Bluetooth. The device then processes the data to determine the current sleep posture observed by the pillow.

Figure 2.8: Integrated Pressure Sensor Array (from [11])

Posture Monitoring Pillow

Boomidevi [12] also developed a pillow for posture monitoring during sleep. The system for detection of turns is very similar to Lokavee's version; however, this pillow can communicate the output using either Zigbee, for close range transmission, or GSM, for long-range transmission, making it more versatile for use either at the hospital or at home. A medical professional can then use the pillow to observe the patient's health remotely in real time.

Heart Rate Monitor Pillow

One of the challenges faced by healthcare practitioners is monitoring heart rate without restricting patients excessively. Yin [13] developed a method to remedy this problem by using a pillow that monitors the patient's heart rate by simply lying on it, without the need for a finger clip or ECG electrodes. It utilises a fibre optic sensor system to detect heartbeats without the need for copper wires. The fibre optic sensors can create a photoplethysmogram (PPG), which is an optically obtained measure of the change in blood volume. This data then goes through signal processing for refinement and outputs the heart rate complete with a visible dicrotic notch, as well as the rise of the systolic peak and slow fall of the diastolic foot, shown in Figure 2.10 in comparison to results from a commercial heart rate monitor in Figure 2.9.

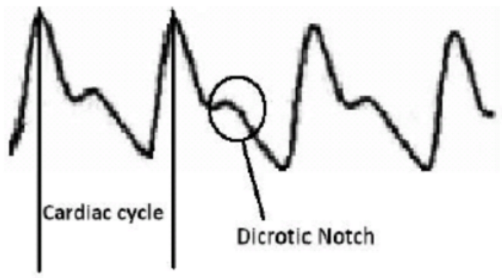


Figure 2.9: PPG Signals from Commercial Heart Rate Monitor (from [13])

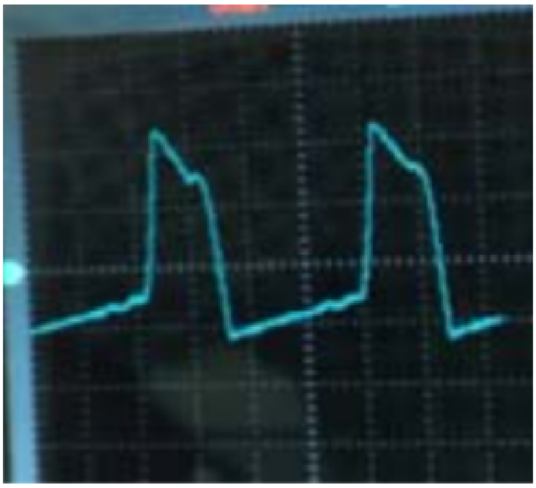


Figure 2.10: Fibre Optic PPG Signals (from [13])

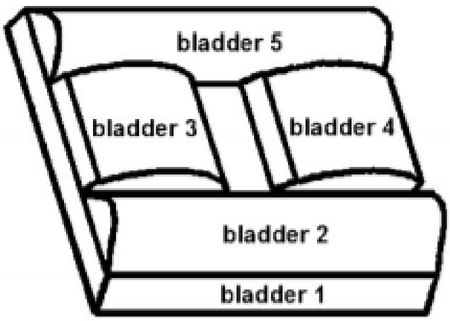
Intelligent Middle-Ear Hearing Device Charging Pillow

Lim's [14] pillow was made specifically for reducing emissions when charging a middle-ear hearing device during sleep. Traditionally, a middle-ear hearing device is charged using several coils implanted in a pillow, all constantly active to account for the person's movement during sleep. As a result, the charging process (which must occur nightly) can cause the user to be exposed to high emissions on a regular basis. Lim's approach to this problem utilises a magnet within the hearing device and a Hall effect sensor within the pillow. The Hall effect sensor outputs a certain amount of voltage dependent on how far the magnet is. This system detects the relative location of the hearing device at each point, and can then selectively activate the primary charging coil closest to the hearing device, thus reducing emissions caused by the other coils.

Auto-Adjustable Pillow for Sleep Apnoea

Another pillow that addresses a specific health concern was researched by Zhang [15] to detect and treat episodes of sleep apnoea. Sleep apnoea is a sleep disorder characterised by pauses in breathing (or periods of shallow breathing) during sleep [15]. It is conventionally treated using a continuous positive airway pressure (CPAP) machine. In addition to causing a lot of noise, this device requires the use of a full face mask. As a result, it can cause difficulty sleeping for many sufferers of sleep apnoea. Zhang's pillow uses a pulse oximeter, which measures the level of arterial oxygen saturation, to detect sleep apnoea events. The data read is transmitted via Bluetooth to a smartphone-based controller which detects drops in blood oxygen saturation, and sends a command to the pillow when needed. When the pillow receives the command, it adjusts the position of the patient's head to clear the air passage using multiple inflatable bladders controlled by its

MCU. When it's been cleared, the pillow returns to monitoring for a new sleep apnoea event.



(a) Pillow Structure

(b) Real Pillow

Figure 2.11: Sleep Apnoea Pillow (from [15])

The bladders are arranged as demonstrated in Figure 2.11. While the ideal shape the pillow adjusts to should differ from person to person, the most important factor is the relative height of the back side of the head compared to the neck. This has the most impact on a person's angle of airflow. As such, when a sleep apnoea event is detected, bladders 2 and 5 are the most helpful in reopening airflow. The process of adjusting the pillow involves some trial and error: the pillow must try different combinations of shapes and check for a response from the oximeter. Some particular combinations were shown to have the highest effectiveness. The pillow will cycle through these combinations until airflow has reopened.

Pillow With Multiple Flexible Actuators

Kuramoto [16] created a general-purpose version of Zhang's pillow. The pillow adjusts to aid head and neck support during sleep. It contains eight flexible actuators, and two different kinds of pressure sensors for increased sensitivity to both major and minor movements. The sensors detect current posture and determine centre of mass, which is then compared to the target centre of mass. When a discrepancy occurs, the valve pressure is adjusted until the target is achieved. This target can be input by the user by measuring the level of actuator pressure at which they feel most comfortable, and retained by the pillow long-term.

IoT-based Smart Pillow for Sleep Quality Monitoring

Veiga [17] developed a pillow for the improvement of ambient assisted living (AAL) environments. This multi-purpose pillow tracks body temperature, room luminosity, humidity, sound, and vibration to be forwarded to and assessed by medical personnel to determine sleep quality. It also helps aid sleep quality when interfacing with an AAL environment. It adjusts the air conditioning when body temperature increases and humidity is detected, indicating sweat, automatically shuts the blinds when high room luminosity is detected to avoid sleep disruption, and reacts to vibrations and sound indicating snoring by briefly playing the sound of a fly to induce a change in sleep position.

2.2.3 Taxonomy of Pillow-based Devices

Below is a taxonomy of all the papers previously mentioned. Each paper is categorised according to its purpose, method of evaluation, type of feedback, type of sensor, and type of actuator.

The purpose refers to the purpose of the pillow, which is further sub-categorised into neonatal intensive care unit (NICU), referring to pillows made specifically for neonates, health monitoring, referring to pillows made to monitor certain aspects of a patient's health, and intervention, referring to pillows which detect some input and react accordingly.

Methods of evaluation refer to whether the research was evaluated subjectively, via a query or questionnaire or objectively, using statistical analysis of some sort. Some papers that did not undergo any evaluation are listed under 'none'.

The type of feedback refers to whether the pillow reacts to stimuli automatically, or whether it waits for a user response before performing any action. Some papers that do not perform any feedback are listed under 'none'.

The types of sensor refers to the devices used by the pillow for input. The subcategories covered within the scope of this taxonomy are force-sensitive resistors (FSR), which are used to detect pressure, pulse oximeters, which read a person's oxygen saturation, hall effect sensors, used to measure the magnitude of a magnetic field, and temperature sensors. One final subcategory contains multiple sundry sensors used collectively for detecting luminosity, humidity, and micro-vibrations.

The types of actuators refers to the method of output the pillow has. This category is divided into inflatable bladders, speakers, vibration motors, and electric coils. Some papers which developed pillows with no output method are listed under 'none'.

	Purpose			Feedback			Sensor Type					Actuator Type					Evaluation		
	NICU	Monitor	Intervention	Manual	Auto	None	FSR	Oximeter	Hall	Temp	Misc	Bladder	Speaker	Motor	Coil	None	Objective	Subjective	None
[8]	X			X				X						X			X		
[9]	X				X			X		X			X						X
[10]		X				X	X									X	X		
[11]		X				X	X									X	X		
[12]		X				X		X								X	X		
[13]		X				X	X									X			X
[14]			X		X				X						X		X		
[15]			X		X			X				X					X		
[16]			X		X		X					X						X	
[17]			X		X					X	X					X			X

Figure 2.12: Taxonomy of Pillow-based Devices

2.3 Summary

This chapter outlines the most crucial facets of this project's background, as well as other works created for similar purposes. Section 2.1 tackles two of the primary aspects of this project: hearing impairment and machine learning technology.

Hearing impairment is a topic at the heart of this project, as the aim of the project is to assist hearing impaired individuals when reacting to an emergency. Hearing impairment is defined as any hearing loss exceeding 25 dBs. It's particularly common in the elderly population, occurring in over half of men over the age of 65.

Machine learning technology is used extensively in the algorithm, covered in Chapter 4. The technology's aim is to identify patterns, and use that information to make intelligent decisions. There are three main types of machine learning: supervised learning, unsupervised learning, and reinforcement learning. The primary difference between supervised and unsupervised learning is whether or not the data is originally labelled with a desirable outcome. Reinforcement learning, on the other hand, teaches the machine to learn behaviours based on feedback received from its environment.

Section 2.2 discusses completed projects that share either purpose or theme with this project. Subsection 2.2.1 discusses projects with a shared purpose: assisting the hearing impaired population, while Subsection 2.2.2 covers some examples of projects that used a pillow as a medium for their device.

Two of the assistive technologies discussed in Subsection 2.2.1 aimed to mimic the behaviour of hearing dogs. Furuhashi [4] created a device that uses wheels to roll towards and bump into its owner when an alert is sounded. Koay [5] created a similar, but more complex, version. In addition to alerting its owner to the occurrence of a sound, it is also able to lead them towards the source of the sound.

Hong [6] and Hosono [7] created phone-related tools to assist hearing impaired people. Hong's tool transcribes a phone call onto a large TV screen to help with communication, in addition to providing the user with a token that delivers haptic feedback to alert them to an incoming call. Hosono, on the other hand, created a mobile tool to be used in case of emergency. The tool, which takes the form of a mobile application, allows the user to report an emergency faster than they otherwise could using text messages. It also assists with communication with emergency personnel by using pictograms.

Subsection 2.2.2 introduces a variety of pillow-based devices. Among them, Chen [8] and Nandhini [9] created pillows for neonates that aimed to simulate a mother's heartbeat and temperature, which would calm them during uncomfortable procedures where their parent cannot be present.

Harada [10], Lokavee [11], Boomidevi [12], and Yin [13] created pillows that would monitor the user's health. Harada's tracked respiration and counted turns during sleep. Lokavee's and Boomidevi's monitored the user's posture during sleep, in addition to counting turns. Finally, Yin's monitors a patient's heart rate without the need for restrictive tools.

Lim's [14] pillow had a unique use: charging middle-ear hearing devices without exposing the user to high emissions during sleep. It activates only the coils closest to the user's hearing device, rather than all coils at all times.

Zhang [15] and Kuramoto [16] created pillows that could be adjusted automatically. Zhang created a pillow that would detect sleep apnoea episodes and adjust the pillow automatically to a new position to stop them. Kuramoto's pillow is originally adjusted to a certain setup, and can re-adjust itself when the pressure changes to keep the user in the most comfortable position.

Finally, Veiga's [17] pillow tracks many aspects of the state of the room and the user, and uses that information to make changes to other devices in the room. For instance, it tracks body temperature, and can use this information to adjust the room's air conditioning.

Chapter 3

Design and Implementation

3.1 Prototype Concept

This project's implementation consists of two aspects: the sensors and actuators, and the software. Due to the COVID-19 pandemic and the resultant difficulty in sourcing hardware components, only the software component has been completed. The hardware aspect outlined below is therefore purely theoretical.

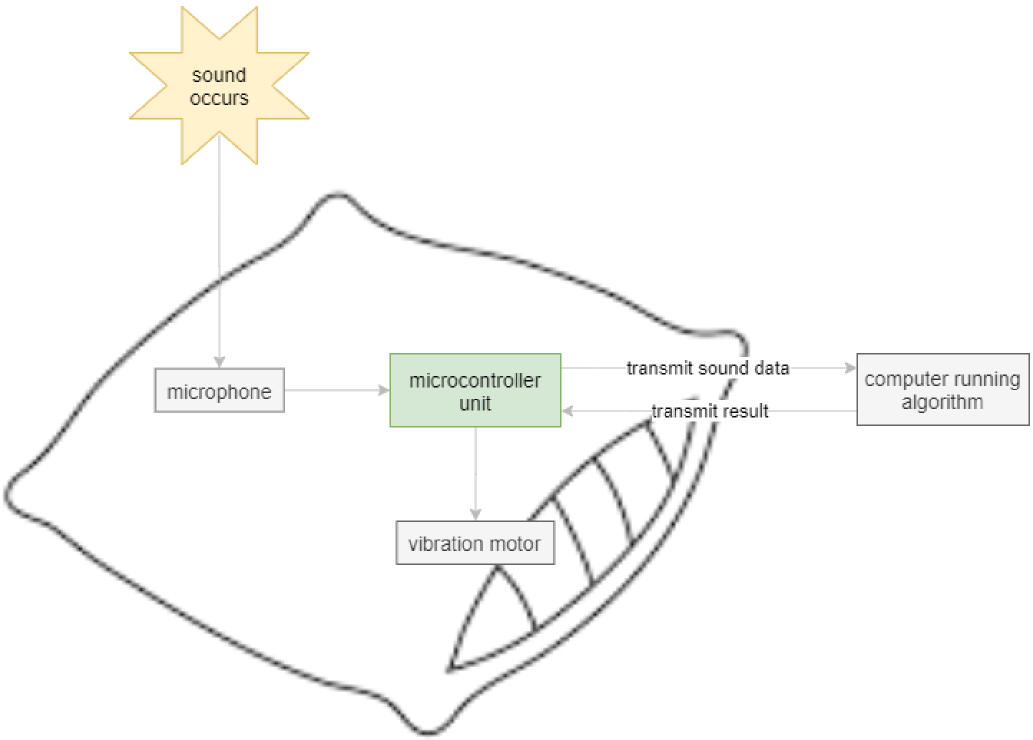


Figure 3.1: Architecture of Pillow Prototype

3.1.1 Sensors and Actuators

The pillow contains a highly sensitive microphone to detect the noises in the room. This microphone is embedded into the side of the pillow for maximum sound detection. It is attached internally to the microprocessor that transmits the data in real time to the device running the classification algorithm externally. This process is demonstrated in Figure 2.11.

After the sound has been run through the algorithm, the response is returned to the microprocessor. The microprocessor is also attached to a vibration motor that acts as an actuator. When the algorithm is unable to classify the current sound in the room, the actuator is turned on for one minute to alert the user of an unfamiliar sound. If the unfamiliar sound continues, the one minute timer is reset until the sound ceases.

The microphone is connected to an Arduino MCU. The microphone's positive terminal is connected to one of the MCU analog terminals, while the negative terminal is connected to the ground (GND). This transmits one channel of audio, which is sufficient for the purposes of this project.

The vibration motor is also connected to the MCU. Its power requirements, however, exceed what can be provided by the MCU itself, so it must also be connected to an external power source. The motor's connection to the MCU should be through a pulse-width modulation (PWM) pin in order to be able to control the intensity of the vibration.

The MCU connects to the computer through Wi-Fi. Therefore, a Wi-Fi module must be connected to the Arduino board. The Wi-Fi module is connected through the RX and TX pins, which enable the board's serial port. This allows the board to transmit information to the Wi-Fi module serially.

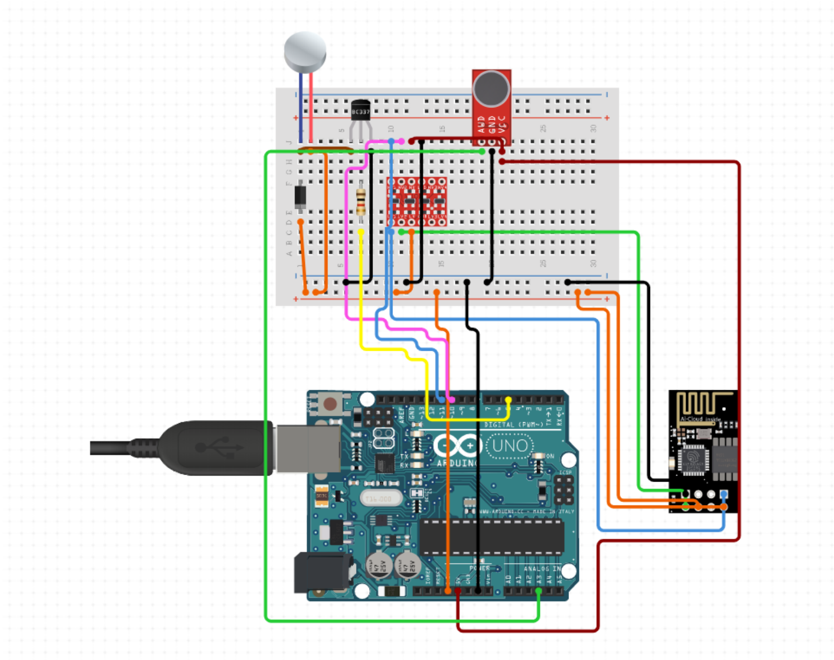


Figure 3.2: Prototype Circuit Diagram

3.1.2 Software

Due to the software's complexity, it must be run on an external device rather than within the pillow. This is to keep the hardware in the pillow to a minimum to maintain usability and comfort.

The software listens to the sounds and divides them into snippets every few seconds, which are then classified into different types of familiar sounds, such as a door opening or closing. As long as all snippets received are classifiable, no alert is sent to the pillow besides an acknowledgement of receipt. However, when the software receives a sound it is unable to classify, it will send an alert to the pillow to enable the vibration motor. If the following snippet of sound is similarly unfamiliar, it will continue to transmit these alerts until the sound ceases. The algorithm is illustrated in further detail in Chapter 4.

3.2 Sound Acquisition

Recording sound accurately in a room requires a powerful microphone. However, due to the inability to source hardware, recording was not possible. Instead, I created various simulations of what sounds might occur during a typical night of sleep.

3.2.1 Ambient Sounds

The base of each simulation is the sound of a person sleeping. This sound, along with all the others used for the simulations, were sourced from freesound.org, which is a collaborative database of sound clips for various purposes. It contains audio of a man breathing in and out during sleep for 28 seconds. This clip is repeated throughout each simulation. While the repetition may not perfectly capture the variety of sounds a person might produce during their hours of sleep, it suffices for the purpose of demonstrating the sound of a person sleeping through the night.

In addition to the ambient sound of the person him- or herself asleep, the ambient surrounding noises also require consideration. The sound of one's environment may vary from one day to another. Therefore, a few different ambient audio clips were selected to cycle between. For instance, on a rainy day, the sound of rain might be quite loud and overpowering, while otherwise on a clear day, you can hear the sound of cars on a nearby street. As such, simulations were produced containing several different possibilities for background noise. Other sources of ambient noise may include people chattering outside, particularly if one does not live alone, or a stormy evening.

3.2.2 Unique Sounds

This concludes the part of the simulations containing ambient sounds; however, some noises needed to be considered uniquely as nonthreatening sounds that occur commonly during the evening, but may be loud enough to trigger the pillow's alert system. For instance, hearing your neighbours open or close their front door or their dog bark outside are sounds that may be frequent to some households but are no cause for the pillow to react.

Each of these categories of sounds that occur uniquely throughout the night require multiple different variations. After all, the sound of a neighbour opening and then shutting their front door would not be identical every single night. Additionally, it could be a different door altogether at a different point during the evening. As a result, these variations were used to ensure that the pillow could recognise and categorise the act of opening and shutting a door conceptually, rather than as a unique noise that must be repeated exactly in order to be recognised.

The example of the door above can be split into two parts: the sound of the act of opening the door, and then the subsequent closing. As a result, they have each been placed into their own class. Each of these sounds can vary based on the weight of the door, the might of the person performing the act, and the speed at which it occurs. As a result, while finding sounds for several different doors is necessary, it's also helpful to alter each audio file to produce a slower or louder version.

This idea extends to other sounds you might hear during the evening. The sound of car horns is loud enough to trigger the pillow to react, but shouldn't be classed as a threatening sound. Therefore, the algorithm must be taught to recognise them as nonthreatening. For each class, this is achieved by introducing the algorithm to a large number of variations. In the case of the car horn, it may vary based on different car models, distance from the pillow, and whether the car is moving.

Similarly, the noise caused by an aeroplane flying above the user's home may be heard during sleeping hours. This sound also varies based on factors such as the kind of aeroplane and its current altitude. Finding or creating these varieties is therefore essential to ensuring the algorithm is capable of identifying aeroplanes in a general sense.

The final class considered is the sound of a barking dog. This sound is particularly crucial if the user owns a dog. A dog barking at the door, for example, would certainly alert the pillow due to its volume. However, even if the user does not own a dog, it's likely that a dog may be heard nearby and alert the pillow regardless. This is especially the case considering that dogs are frequently startled by sounds and scents during the night. The sound of a dog's bark varies depending on its size, breed, and age, so these factors must all be considered and taught to the algorithm accordingly.

3.3 Sound Editing

As outlined above, some of the sound editing was done to the audio files themselves to create different varieties of the same sound, such as a slower or quicker version, somewhat louder or quieter, or more or less clear to simulate it being closer or further away. This is done by adding effects to the sound file using Audacity. There are many options for many different purposes, but primarily, the required effects were either amplification, a change in speed or tempo, or a change in pitch. Adding some echoes or reverb also sometimes creates a believable variety for a sound file.

The majority of the sound editing work, however, occurred when putting these sounds together into clips that simulate a full night's sleep. This was done by putting the ambient noises down as a baseline, and matching their lengths to one another first. Each simulated night contains two ambient sound files: one of the sleeping subject's breathing, and one of the ambient sounds surrounding them. Each of them is looped until it reaches a specific length, and then clipped to remove the excess time. Depending on each audio file, it may be necessary to change the volume or pitch of the file to ensure they match one another and could believably have been recorded using the same microphone.

The unique sounds are then placed over the ambient sound file. In some cases, certain unique sounds ought to be associated with each other. For instance, a door closing may need to be placed subsequent to it opening. After that, the volume of each sound may need to be altered slightly to both fit the subsequent sounds and to fit the overall sound file. The sounds are spaced out to some degree to ensure they're not chained one after another.

Figure 3.3 illustrates the process outlined above. It's split into three sections; two are ambient, and one contains unique sounds. The top section contains the sound of a sleeping person. Each vertical line represents the cutoff where the sound repeats itself. Below that, the middle section contains the scattered unique sounds. This illustration does not capture the full length of the clip; accordingly, the unique sounds take up a significant amount of space. If the full clip was visible, they would appear much smaller and more dispersed. Finally, the bottom section shows the ambient sound added to create a believable atmosphere. In this particular clip, the ambient audio file contains rain drizzling outside.

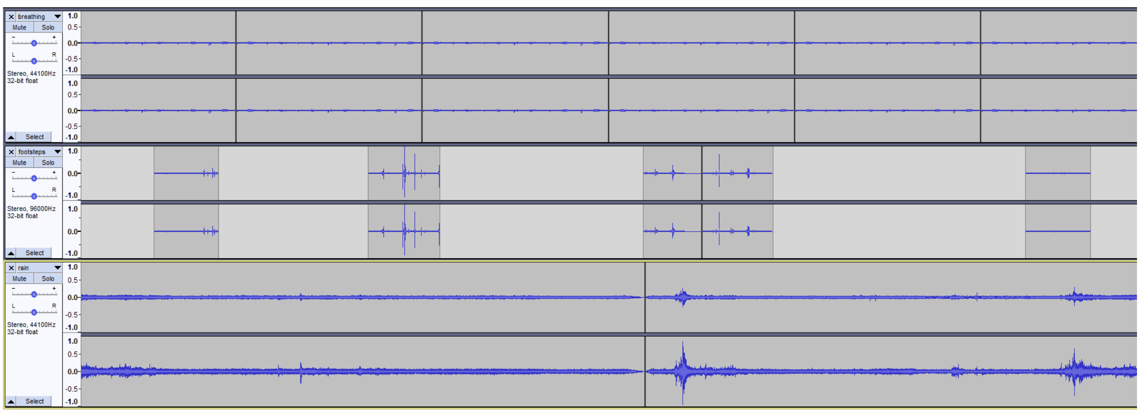


Figure 3.3: Example of a Layered Audio File Using Audacity

The final step is to listen to the parts of the full seven-hour sound file containing unique sounds and review how it sounds. It should contain limited repetition, and the sounds should feel cohesive. This final review also checks that there is no issue with one sound being too loud and overpowering the other sounds, or too quiet to be heard.

When the sound file is completed and reviewed, it's imported as an MP3 file. This is both to make it more widely readable and to compress it. Along with all the other simulations, it is stored in a folder that will be used later as training data.

3.4 Summary

This chapter discusses the process of gathering, refining and editing the audio files later used as the project's machine learning training dataset. It also outlines the overall vision for this project's design and implementation.

In Section 3.1, the project's components are introduced, including both hardware and software aspects. The hardware aspect of the pillow's construction required some components, which could not be sourced due to the widespread closures caused by the COVID-19 pandemic.

Besides the pillow itself, the project called for a highly sensitive microphone to capture the sounds in the user's room and a microprocessor able to transmit data wirelessly to the device running the classification algorithm. Additionally, a vibration motor was required to act as an actuator, and alert the user when an unfamiliar sound is detected.

The software aspect briefly outlines the algorithm's purpose and behaviour, discussed further in Chapter 4. The algorithm, which runs on a device separate from the pillow due to its complexity, divides the audio received into brief fragments, each only a few seconds long. It then classifies each into a category. When a section of audio is not classifiable, it sends an alert to the pillow's MCU, which activates the vibration motor.

Section 3.2 explains the process of compiling and categorising the audio files contained in the training dataset. The first category compiled is ambient sound files. Those files contain the sound of a sleeping person, in addition to the fundamental ambience of their surroundings, such as the sound of cars outside or a downpour of rain.

The second category contains sounds that occur uniquely during the night, but are considered nonthreatening. For instance, the sound of a neighbour's front door shutting may cause a significantly loud noise, but this noise is no cause for an alert. Each type of unique sound is categorised along with other similar types, within categories such as *door opening* or *dog barking*.

Section 3.3 demonstrates how the audio files above were assembled into a complete sound clip which mimics the noises heard during a typical night of sleep. The first step was to establish the fundamental atmosphere using ambient sounds. The sound of a sleeping person was layered with the ambient sound of a surrounding environment, such as the distant sound of a busy motorway outside. This layered file repeats itself throughout the clip. After that, unique sounds may be added to the layered clip at random intervals. Finally, the unique portions must be reviewed to ensure their volume and pitch conform to the background audio.

Chapter 4

Algorithm

The sound samples outlined in the previous chapter were pre-classified: that is to say, we already know what category each one falls under. However, in order for the pillow to be effective, it must be able to identify whether or not a new sound sample is familiar. It does this by checking whether it matches any categories it's currently aware of. This chapter will discuss how the pillow's system learns to identify different categories of audio.

4.1 Audio Signal Feature Extraction

Because audio signals are made up of a variety of features, certain characteristics have to be extracted before it can be processed into categories. Extracting these characteristics is done by processing the audio signal into a spectrogram, shown in Figure 4.1. This can be done using Librosa, a Python library used for audio analysis. Once the audio file has been imported into it, a spectrogram can be generated using a simple function call.

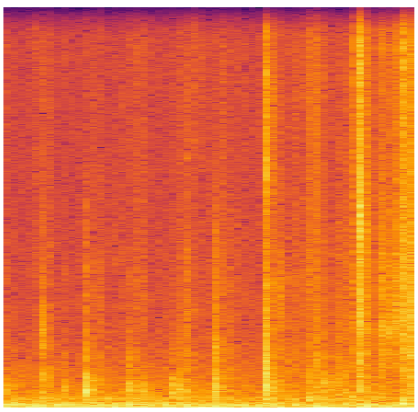


Figure 4.1: Example of a Spectrogram Generated Using Librosa

After the spectrograms have been created, the following step involves extracting features from each one into a comma-separated values (CSV) file. Each feature will be extracted into a separate column. As a result, the CSV file must be prepared with a header containing each column name. The following are the features that must be extracted, as defined by Theimer [18]:

- **Spectral Centroid:** This feature denotes the location of the centre of gravity of the amplitude spectrum of sound, similar to a weighted mean of the frequency throughout the file.
- **Spectral Rolloff:** This feature represents the frequency below which a certain percentage, generally 80% is of the signal's energy is contained.
- **Spectral Bandwidth:** This feature expresses the extent from the centre frequency that the power transfer function covers, illustrated in Figure 4.2.

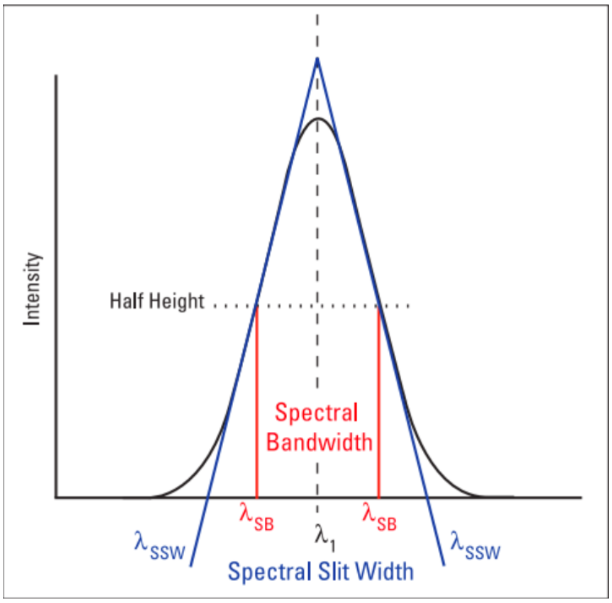


Figure 4.2: Spectral Bandwidth Visualisation (from [18])

- **Zero-Crossing Rate:** This feature measures the high frequency content of a sound.
- **Mel Frequency Cepstral Coefficients (MFCCs):** This feature expresses the logarithm of the amplitude spectrum, computed on the Mel bands rather than the Fourier spectrum. It helps describe the general shape of a spectral envelope.
- **Chroma Vector:** This vector encapsulates the full range of spectral components belonging to a particular pitch class. This helps describe similarity between audio files.

- **Root Mean Square (RMS):** This feature expresses a normalised measure of the sound's signal energy during a specific window of time.

Alongside the features outlined above, the CSV header also contains the original name of the audio file, whose purpose is to identify it. Once the CSV file has been created and populated with the header, the features can be extracted into it.

In order to extract the features, the algorithm must loop over each spectrogram. While iterating over the files, Librosa functions can be used to extract each feature from each one. Once all of the features of an audio file have been extracted, it can be inserted into the CSV file as a new row.

When the full dataset has been loaded into the CSV file, it requires some pre-processing before it can be used. The pre-processing involves encoding the audio file name into numeric form, scaling the features, and splitting the data into a training set and a test set. The test set will be used later to test the model's accuracy at classifying an audio file.

4.2 Machine Learning Model

4.2.1 Artificial Neural Networks (ANNs)

According to Wang [19], an artificial neural network is a machine learning model inspired by the human brain's interconnected nature. The human brain's neurological structure contains hundred of billions of neurons, each processing data in parallel.

As shown in Figure 4.3, an artificial neural network is made up of three primary components: the input layer, the hidden layer(s), and the output layer.

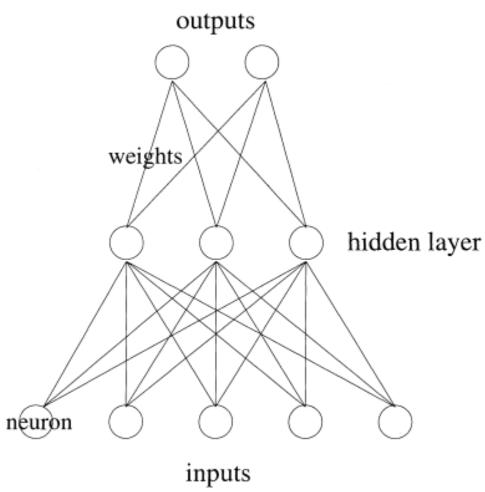


Figure 4.3: Architecture of a Neural Network (from [19])

The output of a given node is determined by the network's activation function. An activation function also helps introduce boundaries to the value of each output, which allows the network to produce more harmonious results. There are many different functions that may be used as an activation function, such as the sigmoid (or logistic) function, the arc tangent function, and the hyperbolic arc tangent function. However, for the purposes of this project, the activation functions used are the rectified linear unit function (ReLU) and the Softmax function.

ReLU is defined by Ramachandran [20] as $f(x) = \max(x, 0)$. It is regarded as the most successful and most commonly used activation function. This is because ReLU facilitates network optimisation when compared to other functions referenced above.

According to Low [21], the Softmax function is used to determine whether a training sample belongs to a particular class based on its weight and net input. Because the aim of this project is to classify sounds into several different classes, the final layer of the model uses Softmax as an activation function.

4.2.2 Additional Libraries

The model built for this project was constructed using Keras, which is an application programming interface (API) designed to support neural network architecture. Keras can run on top of a few different library, but the library used for this project was TensorFlow. TensorFlow is an open-source library available for Python that is used for machine learning applications.

Besides these frameworks, the implementation of this project required the use of a few more libraries. Chief among those is NumPy, a library supporting large, multidimensional arrays, Pandas, a library for data manipulation and analysis, and Matploblib, a library for plotting visualisations.

4.3 Summary

This chapter discusses the development of the classification algorithm for the project. This algorithm receives the audio stream from the pillow's MCU, and classifies the sound fragments. If a loud sound fragment is not classifiable, it notifies the MCU.

Section 4.1 discusses how the audio files are processed into data the algorithm can utilise. The first step of the process is converting the each audio file into a spectrogram, depicted above in Figure 4.1.

Once this is done for each file, features can be extracted from each spectrogram into a CSV file. Each spectrogram's features are extracted into one row, containing the following features: file name, spectral centroid, spectral rolloff, spectral bandwidth, zero-crossing rate, MFCCs, chroma vector, and RMS. Finally, the CSV file is pre-processed to prepare for use in the training to follow.

Section 4.2 describes the machine learning model used to develop the algorithm. This project used an artificial neural network to classify the contents of the CSV file above. Artificial neural networks consist of a network of nodes, shown above in Figure 4.3. Each node's output is determined by the network's activation function. For the purposes of this project, the activation functions used were ReLU and Softmax.

The section also discusses some of the key libraries used in the development of the algorithm. Keras, an API which supports neural network architecture, was used to construct the model described above. Additionally, NumPy, Pandas, and Matplotlib each had a key role in the data processing involved in the algorithm's development.

Chapter 5

Conclusion

5.1 Achievements

The project's algorithm, explained thoroughly in Chapter 4, aimed to categorise night-time household sounds into their respective classes using an artificial neural network model. After a training period, the algorithm was tested using a subset of the data composed of 20% of the audio files in each class. It achieved an accuracy of 85.7% when sorting them into their appropriate classes.

Additionally, the algorithm is capable of isolating noises louder than a certain decibel volume from a sound file, and saving them to disk to be used for testing. The algorithm should then be able to identify the class of each noise with the aforementioned accuracy.

5.2 Limitations

One of the project's limitations is the scarcity of training data. Each class was trained using a dataset between 25 and 50 files large. The model might have achieved higher accuracy had it been presented with a larger training dataset.

The model is also triggered by any sound above a certain decibel volume, except for ones it's been trained to recognise. Accordingly, the model may need further training to recognise other categories of sounds that are nonthreatening despite being loud.

Additionally, due to the widespread closures caused by the COVID-19 pandemic, the project's full vision could not be achieved. Hardware components could not be sourced in order to assemble the pillow's internal architecture. As a result, the algorithm is not connected to a tangible system.

5.3 Future Work

Further support is needed for hearing impaired individuals during emergency situations. The purpose of the intelligent pillow is to rouse the user in case of emergency. However, once the user is awake, a system should be put in place to notify them about the nature of the emergency.

A mobile application delivering reliable notifications whenever an incident is reported in the surrounding area would be an adequate step towards ensuring safety for the hearing impaired population. The application should also be able to guide the user about the correct emergency response towards different types of incidents.

The user should be able to receive live instructions from the application as the situation progresses. If they have questions, concerns, or further information regarding the incident, they should be able to contact an emergency worker via text directly through the application.

The application may also be used as a networking tool to be used by neighbours and friends, making it possible for them to be notified when a particular user is in an area where an emergency has occurred. Additionally, the user may use the networking tool to signal to anyone connected to them through the application that they are in need of aid. They may also use this tool to reassure their friends and family when they have reached safety.

Appendix

Appendix A

Lists

MCU	microcontroller unit
ANN	artificial neural network
GND	ground
dB	decibels
PWM	pulse-width modulation
CPAP	continuous positive airway pressure
PPG	photoplethysmogram
AAL	ambient assisted living
NICU	neonatal intensive care unit
FSR	force-sensitive resistors
CSV	comma-separated values
MFCCs	Mel Frequency Cepstral Coefficients
RMS	Root Mean Square
ReLU	rectified linear unit function
API	application programming interface

List of Figures

Bibliography

[1] Erin Mack Ashley. *Waking effectiveness of emergency alerting devices for the hearing able, hard of hearing, and deaf populations.* PhD thesis, 2007.

[2] NE Morton. Genetic epidemiology of hearing impairment. *Annals of the New York Academy of Sciences*, 630(1):16–31, 1991.

[3] Poonam Mahana and Gurbhej Singh. Comparative analysis of machine learning algorithms for audio signals classification. *International Journal of Computer Science and Network Security (IJCSNS)*, 15(6):49, 2015.

[4] Michihiko Furuhashi, Tsuyoshi Nakamura, Masayoshi Kanoh, and Koji Yamada. Haptic communication robot for urgent notification of hearing-impaired people. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 429–430. IEEE, 2016.

[5] Kheng Lee Koay, Gabriella Lakatos, Dag Sverre Syrdal, Márta Gácsi, B Bereczky, Kerstin Dautenhahn, Adám Miklósi, and Michael L Walters. Hey! there is someone at your door. a hearing robot using visual communication signals of hearing dogs to communicate intent. In *2013 IEEE symposium on artificial life (ALife)*, pages 90–97. IEEE, 2013.

[6] Hsiang-Ting Hong, Tzu-Yu Su, Po-Hsun Lee, Ping-Chun Hsieh, and Mian-Jhong Chiu. Visuallink: Strengthening the connection between hearing-impaired elderly and their family. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 67–73, 2017.

[7] Naotsune Hosono, Hiromitsu Inoue, Miwa Nakanishi, and Yutaka Tomita. Urgent mobile tool for hearing impaired, language dysfunction and foreigners at emergency situation. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 413–416, 2014.

[8] Wei Chen, Sidarto Bambang Oetomo, Daniel Tetteroo, Frank Versteegh, Thelxi Mamagkaki, Mariana Serras Pereira, Lindy Janssen, and Andrea van Meurs. Mimo pillow—an intelligent cushion designed with maternal heart beat vibrations for comforting newborn infants. *IEEE journal of biomedical and health informatics*, 19(3):979–985, 2014.

[9] P Nandhini[1], J Bethanney Janney, and Sindu Divakaran. Design of an intelligent pillow with maternal temperature and heartbeat vibrations for comforting nicu infants. *Indian Journal of Science and Technology*, 9(33):1–5, 2016.

[10] Tatsuya Harada, Akiko Sakata, Taketoshi Mori, and Tomomasa Sato. Sensor pillow system: monitoring respiration and body movement in sleep. In *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113)*, volume 1, pages 351–356. IEEE, 2000.

[11] S Lokavee, N Watthanawisuth, JP Mensing, and T Kerdcharoen. Sensor pillow system: Monitoring cardio-respiratory and posture movements during sleep. In *The 4th 2011 biomedical engineering international conference*, pages 71–75. IEEE, 2012.

[12] R Boomidevi, R Pandiyan, and N Rajasekaran. Monitoring cardio respiratory and gesture recognition system using sensor pillow system. *Int. Res. J. Eng. Technol.(IRJET)*, 2(8):83–87, 2015.

[13] Everlyn Chiang Hai Yin and V Muthukumar. Intelligent pillow for heart rate monitor. *International Journal of Electrical and Computer Engineering (IJECE)*, 3(6):791–796, 2013.

[14] Hyung-Gyu Lim, Dong-Wook Kim, Myoung-Won Lee, Jang-Woo Lee, Sang-Hyo Woo, Min-Woo Kim, Eui-Sung Jung, Ki-Woong Seong, Jyung-Hyun Lee, and Jin-Ho Cho. Intelligent pillow type wireless charger for fully implantable middle ear hearing device with a function of electromagnetic emission reduction. In *2008 Second International Symposium on Intelligent Information Technology Application*, volume 3, pages 835–838. IEEE, 2008.

[15] Jin Zhang, Qian Zhang, Yuanpeng Wang, and Chen Qiu. A real-time auto-adjustable smart pillow system for sleep apnea detection and treatment. In *2013 ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 179–190. IEEE, 2013.

[16] Akisue Kuramoto, Wataru Inoue, Yasuhito Otake, Hitoshi Kimura, Norio Inou, Sacha Krivokuca, Tomu Ichikawa, Hiroyuki Ono, and Naoto Sekiyama. Development of an intelligent pillow with multiple flexible actuators. In *The 16th International Conference on Biomedical Engineering*, pages 1–6. Springer, 2017.

[17] Alejandro Veiga, Laura Garcia, Lorena Parra, Jaime Lloret, and Vivian Augele. An iot-based smart pillow for sleep quality monitoring in aal environments. In *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 175–180. IEEE, 2018.

[18] Wolfgang Theimer, Igor Vatolkin, and Antti Eronen. *Definitions of audio features for music content description.* Tech. Univ., Fak. für Informatik, 2008.

[19] Sun-Chong Wang. Artificial neural network. In *Interdisciplinary computing in java programming*, pages 81–100. Springer, 2003.

[20] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[21] Jia Xin Low and Keng Wah Choo. Classification of heart sounds using softmax regression and convolutional neural network. In *Proceedings of the 2018 International Conference on Communication Engineering and Technology*, pages 18–21, 2018.