DETECTION OF ELEPHANT INTRUSION WITH SEISMIC SENSORS AND MACHINE LEARNING

S.A.D.S. Jayathunga

(148460 J)

Degree of Master of Science

Department of Electronic and Telecommunication Engineering

University of Moratuwa
Sri Lanka

November 2018

DETECTION OF ELEPHANT INTRUSION WITH SEISMIC SENSORS AND MACHINE LEARNING

S.A.D.S. Jayathunga

(148460 J)

Thesis submitted partial fulfillment of the requirements for the degree Master of Science in Electronic Engineering

Department of Electronic and Telecommunication Engineering

University of Moratuwa

Sri Lanka

November 2018

I declare that this is my own work and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date: 21st May 2019

The supervisor/s should certify the thesis/dissertation with the following declaration.

The above candidate has carried out research for the Masters thesis under my supervision.

Name of the supervisor: Dr. M.A.U.K. Premaratne

Signature of the supervisor: Date :

ABSTRACT

Human elephant conflict (HEC) is a severe social issue in several Asian countries. A possible approach to prevent HEC is to identify the presence of elephants remotely, thus people can get precautions. This research introduces a method to detect presence of elephants by acquisition of seismic signals generated by their footfalls. A seismic sensor – Geophone was used to convert seismic waves to analog signal and then it is converted to digital domain. Digital signal processing techniques have been used to develop an algorithm that distinguishingly identifies subsequence signals due to elephant footfalls.

Developing such an algorithm was a major objective of the research. A novel algorithm has been developed based relative harmonic contents of the transient signal generated by elephant footfalls. Machine learning algorithms have been used to get the intuition and obtain this algorithm. It uses features of transient signal generated by a single footfall; thus a detection result is generated for every individual footfall. This makes real-time detection possible.

Data acquisition and recording hardware has been designed. Site recorded data was processed and analyzed offline in MATLAB environment with a laptop computer. Development and testing of the algorithm was done entirely in MATLAB. However algorithm was designed to be implemented with much less computational power in a microcontroller. Therefore the electronic systems which will use this algorithm can be fabricated as portable units and they can be used at HEC affected areas to get elephant intrusion warnings.

Algorithms developed with SVM classification and relative harmonics contents could successfully detect elephant footfalls below average time period of 6s; even when high environmental seismic noise is present. This had been lowered to 3s periods when there is less seismic noise. False detection has average periods of 10s or more.

To my appachchi and amma, Mr. & Mrs. Jayathunga for teaching me to read.

ACKNOWLEDGEMENT

I could do my research because of the grateful help I got from many great people. My family farther, mother, sister and brother-in-law are my source of working aptitude. Inspiration and norm I get from them are the strength to pursue my work. I could extensively benefit from research experience of my sister, Maheshi. She gave me the confidence that I'm working on a worthy research.

I especially thank to my supervisor Dr.Upeka Premarathna for guiding and correcting me throughout this research. Also without his encouraging words this research would not cover extent that I expected. I wish I would have work with him more closely.

I also thank Prof. Rohan Munasinghe for being with our batch and for advises he gave us on conducting the research. As well I thank Dr. Chamira Edusooriya and Dr. Shehan Perera for giving me advices in the reviews which broaden the scope and knowledge.

I acknowledge the Department of Electronic and Telecommunication Engineering of University of Moratuwa including its entire staff for facilitating to conduct the Master degree and research. I would like to thank all my batch mates who cheerfully helped and cooperated in every stage of the research. Lastly I'm grateful to office staff in my division for their understanding and leaving me to work on this research.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Sri Lanka is the home for Asian elephants (*Elephas maximus maximus*) which IUCN has listed as **Endangered** (criteria A2c) in 1986. Having the highest elephant density in the Asia region, Sri Lanka is in a crucial phase of conservation of elephants.

Human Elephant Conflict (HEC) is one of severe social issue Sri Lankan people suffers especially who lives in dry-zone of the country. Statistics show elephants yearly caused human deaths about 73 and several crop raids. As well humans are responsible for more than 135 elephant’s deaths yearly for past several years.

Most of the human deaths due to elephant strikes can be avoided if the presence of elephants can be identified at a far distance to them. People can be informed to get early precaution for protection of their lives and vulnerable properties. This can save human lives in close encounters with elephants which lead to elephant strikes; and thus reduce retaliation attacks on elephants which are the primary reason for elephant killings. Also elephant chasing actions can be initiated if elephant movements are identified near crops or any vulnerable property.

The motivational fact for conducting this research is above mentioned identification of elephants at a far distance can save both human and elephant lives. This will leads to reduce conflict incidences and conserve the endangered species. The methodology exploited in this research is the seismic waves. When humans or any animal move on ground seismic waves are generated because of footfall impacts on ground. For identification of presence of elephants, seismic waves generated by elephant footfall should be recognized distinctly.

The ultimate task is to develop a DSP algorithm and implement it on a MCU for real-time detection of seismic signal generated due to elephant footfalls. Power consumption and cost factors are also considered in the algorithm development stage, which to be makes it possible to implement on available technology and affordable to the community that directly confronting the human-elephant conflict. Recent developments of MCUs would facilitate this paradigm.

Also hope to establish a ground work and present a flexible algorithm that can be modified, developed and implemented even on different hardware to detect and categories various seismic signal sources. Hence features of seismic signal that are exploited in this research can be used to identify many other seismic signal sources by manipulating parameters of relevant features. Also implemented algorithm will be described relating as much as possible to theoretical background. This approach will help anyone to pursue this system further by simulation only.

1.1 Asian elephant population in region

Sri Lankan wild elephant (*Elephas maximus maximus*) is one of the 3 sub species of Asian elephant (*Elephas maximus*). The other two subspecies are;

- Indian elephant (*Elephas maximus indicus*) lives in mainland Asia:
  Bangladesh, Bhutan, Cambodia, China, India, Laos, Malay peninsula, Myanmar, Nepal, Thailand, Vietnam; Sabah (Malaysian part of Borneo) and Kalimantan (Indonesian part of Borneo).
- Sumatran elephant (*Elephas maximus sumatranus*) lives in Sumatra.[1, pp. 8]

According to data of 2011, world Asian elephant population is 43 445 and Sri Lanka has 5879 of them. Over 13% of Asian elephants occur in Sri Lanka which only has 0.4% total land among 14 countries which they occur [2]. Figure 1 shows Sri Lanka has the highest elephant density over land and figure 2 show "people per elephant" ration is lowest in Sri Lanka**.** This leads to recognize Sri Lanka tends to have highest human elephant confrontation.

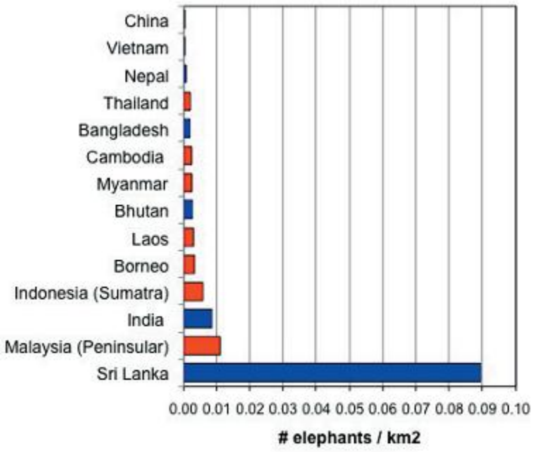


Figure 1. Elephant density in South (blue), Southeast (red) and East (China) Asian range countries [2].

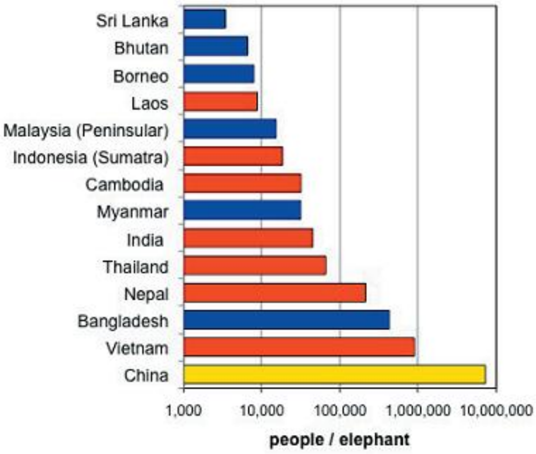


Figure 2. Number of people per elephant in South (blue), Southeast (red) and East (yellow) Asian range countries. Note: logarithmic X-axis [2]

1.2 Sri Lanka cultural association with elephants

Sri Lankan has very close cultural association with elephants which extends back to first kingdom of Sri Lanka, the Anuradhapura era (377 BC–1017 AD). They can be noted in "Sandakadaphana" stone carvings in kingdoms of Anuradhapura, Polonnaruwa and Kandy eras. Also elephants have honorable mention in Lord Buddha's life and army of king Dutugamunu (161 BC to 137 BC).

In kingdom era, captive elephants are used for labor, war, religious and cultural activities. Taming and training of them is still remaining as an ancient treatise. The elephant is considered as a symbol of physical and mental strength, intelligence, responsibility, good luck [7] and considered as sacred animal.

Elephants are the most attractive feature of the Kandy perahera which is the annual ceremonial to exhibit sacred tooth relic of the Lord Buddha. There are about 100 glamorously decorated elephants feature in the Kandy perahera and the tooth relic is also carried by a majestic tusker. Currently use of elephants for labour is drastically reduced and no involvement with war. New trend is to use them in tourism for elephant rides.

1.3 Reasons for extinction of Sri Lankan elephants

1.3.1 Loss of habitat

Expansion of human population and cultivation leads to convert forest lands into human utilization and thus results to shrink and fragment natural habitat of elephants. Sri Lankan elephants are already disappeared from montane zone of Sri Lanka due to extensive clearance of the up-country forests during first half of 19th century. Before that they were found in cold, damp forests of Colombo, Kandy and Ratnapura in between 1669 and 1744 [6].

In 1881 about 84% of the land was covered by closed canopy forest, by 1956 it was declined to 44% and in 1983 it was only 27%. By 2007 it was about 22.2%. Much recent statistics should be much lower as we experienced organize massive forest destruction in “national parks” conserved by Forest Conservation Department. Hence only about one fourth of forest remains today as compared to 137 years ago.

1.3.2 Ranging pattern of elephants

The Asian elephant is recognized as one of the last surviving mega-herbivores in the world. Adult body mass exceeds 1000kg. Normal walking speed of Asian elephant about 1.7 km/h but they can reach up to 21.6 km/h (6 m/s) at full speed [8].

Sri Lanka is an island 65610 km^2 and dry zone is low land of 60% of total which elephants mostly live. As shown in figure 3, covering 13.6 % land area, and 8616 km^2 are preserved as Protected Areas (PA) shown in Figure 3 for wild life [3]. But elephant habitat cannot be restricted to protect areas, in most countries elephants extent their range outside protected areas. Hence most conflicts occur on outside of protected area. The reason for that is optimal habitat for elephants is not undistributed forest but with an intermediate disturbance regime [4]. According to DWC survey 2011, 67.19% elephants live within PA, 29.78% live in forest reservation of Forest conservation department [1, pp. xvii]

Sri Lankan elephants don not have separate wet and dry seasonal ranges and no seasonal long-distance migration. Female groups have small well defined home range of 30 to 140 km^2 in extent to which they show high degree of fidelity they keep to same area year after year. Also males occupy small ranges less than 100 km^2 over most of the year but drastically increase during their *musth* period of about two months about 4 times their range for rest of the year. Females seemed to be much more restricted by management measures such as electric fences. Most crop raiding and conflicts with human in the southern area was contributed by male elephants [5].

1.3.3 Destruction of elephants

Island wide national survey conducted by DWC in 2011 shows elephant population is 5879 [1, pp. 32]. Estimates of elephants from 1951 and 2011 survey results shows growth of elephant population from 1500 to 5879 [2], but this is only a fraction of elephant population 200 years ago.

Before colonial era, elephants were property of king and could not kill or captured without royal permission [1, pp. 11]. But during British rule from 1796 to 1948 elephant population dropped from about 10000 to 2000 due to excessive hunting and loss of habitat. Until 1830 elephant destruction was encouraged by British government. Rewards were paid for elephant killing. 3500 elephants were shot in 3 years up to 1848 in the Northern Province; in Southern province 2000 elephants were killed within four years up to 1855. More than 5000 elephants were eliminated systematically within a period of 10 years. Major Thomas William Rogers was reputed for killing 1400 elephants during 11 years period from 1834 to 1845. Another Captain Gallway has credited slaughter more than half of that and Major Skinner, the commissioner of roads, has killed about same amount. Apart from killing, in between 1863 and 1899 total of 2190 elephants were exported to zoos in USA and Europe [1, pp. 13].

Since Sri Lankan society condemns animal abuse elephant hunting is not considered as a sport and hence it was ended with colonial power. Main reason for current elephant killing is retaliation to crop raid and human killing. Ivory poaching has low significant recently may be due to only about 7.3% of bulls have tusks while in southern India it is high as 90% [6].

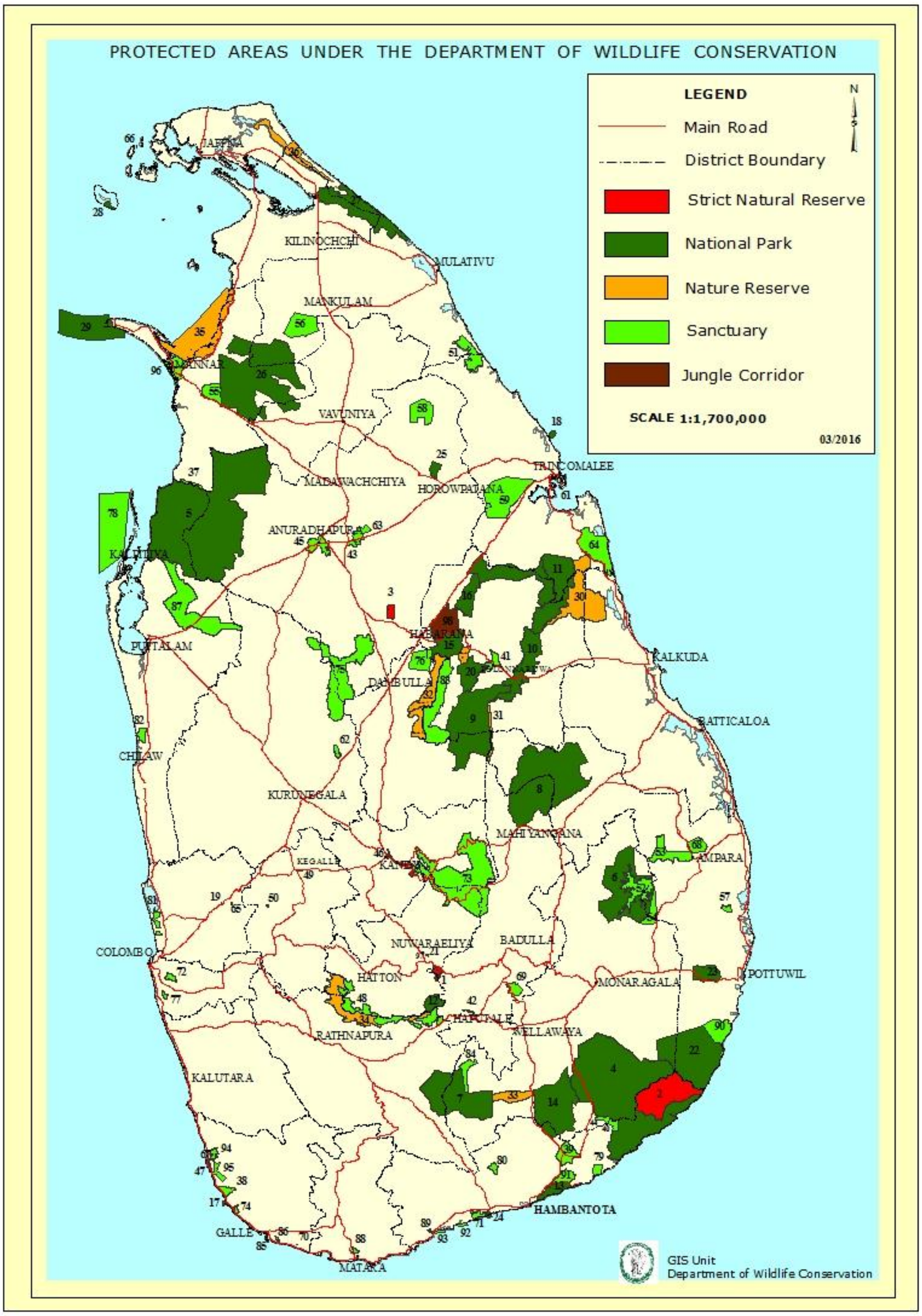


Figure 3: Protected areas under the Department of Wild Life Conservation in Sri Lanka

1.4 Status of Human-Elephant conflict (HEC)

As food and water become scarce in their shrunken habitat, elephants tend to enter villages and raid crops to feed themselves. Elephants are highly like to raid paddy, corn, sugar cane, finger millet; vegetables such as pumpkin, sweet potato beans pulses; fruits such as banana, water melon and mango [7]. Also when they encounter humans or any property on their way as an obstruction, they tend to attack and destroy.

Table 1 shows loss of human and elephant lives during 2012 to 2016 iod due to HEC.

Table 1**:** Loss of human and elephant lives during 2012 to 2016 period due to HEC. Source: DWC performance reports from 2012 to 2016.

http://www.dwc.gov.lk/Aoldsite/index.php/en/performance-reports

Year	**Elephant deaths**					**Human deaths due to elephant attacks**
	Due to intentional killings*	**Due to train and other accidents**	**Due to Natural causes**	**Due to unknown and other causes**	**Total**	
2012	116	28	15	91	250	79
2013	99	17	16	74	206	70
2014	104	19	33	75	231	67
2015	117	18	11	59	205	63
2016	131	29	35	84	279	88
TOTAL	**567**	**111**	**110**	**383**	**1171**	**367**

*Include killing by 1) Gunshot, 2)Electrocution, 3)Poisoned, 4)Hakkapatas (a small pressure mine concealed in fruits or vegetables, which shatters the jaw when bitten)

1.5 Related previous research

For mitigation of HEC, remote identification of elephants can be used as virtual barrier to them. This has been practiced with several techniques;

1) Use of PIR sensors and Infrared thermography [9].
2) Detecting infrasound (14~24Hz) calls of elephants [10],[11]
3) Visual image processing [12]

Also M. S. Nakandala et al. (2014) [9] tried to identify seismic waves of elephant footfalls with correlation coefficients, but abandoned due to high error and in the requirement of high sensitive seismic sensor and sound card like VXpocket V2.

O’Connell-Rodwell et al. (2000) [13] has shown time-domain cross-correlation coefficients function of acoustic and seismic signals when elephant rumble and foot-stomp.

Jason D. Wood et al. (2005) [14] has used recording of seismic signal to differentiate species by counting threshold-limit (0.8) exceed instances of correlation coefficients in 100 consecutive FFT sequences corresponds to 64s long signal; the processing had been done on MATLAB after collecting data at the site.

Succi et al. (2001) [15] has described footstep detection in seismic sensor recording by Kurtosis.

Also Succi et al. (2000) [16] has detected human footsteps in seismic wave recording with use of frequency of peaks and bearing.

Houston et al. (2003) [17] has rejected footstep detection by detection of transient seismic waves corresponding to individual footsteps, reasoning it as error-prone approach. Instead, they used periodicity of footstep on ground, and shown a spectrum analyzing method on envelop-detected (26s long, 40Hz sampling) seismic signal to detect presence of humans.

Koç, G.,et al. (2013) [18] has shown real time detection algorithm that analyses duration of seismic signal above two adaptive thresholds to detect footsteps and vehicle.

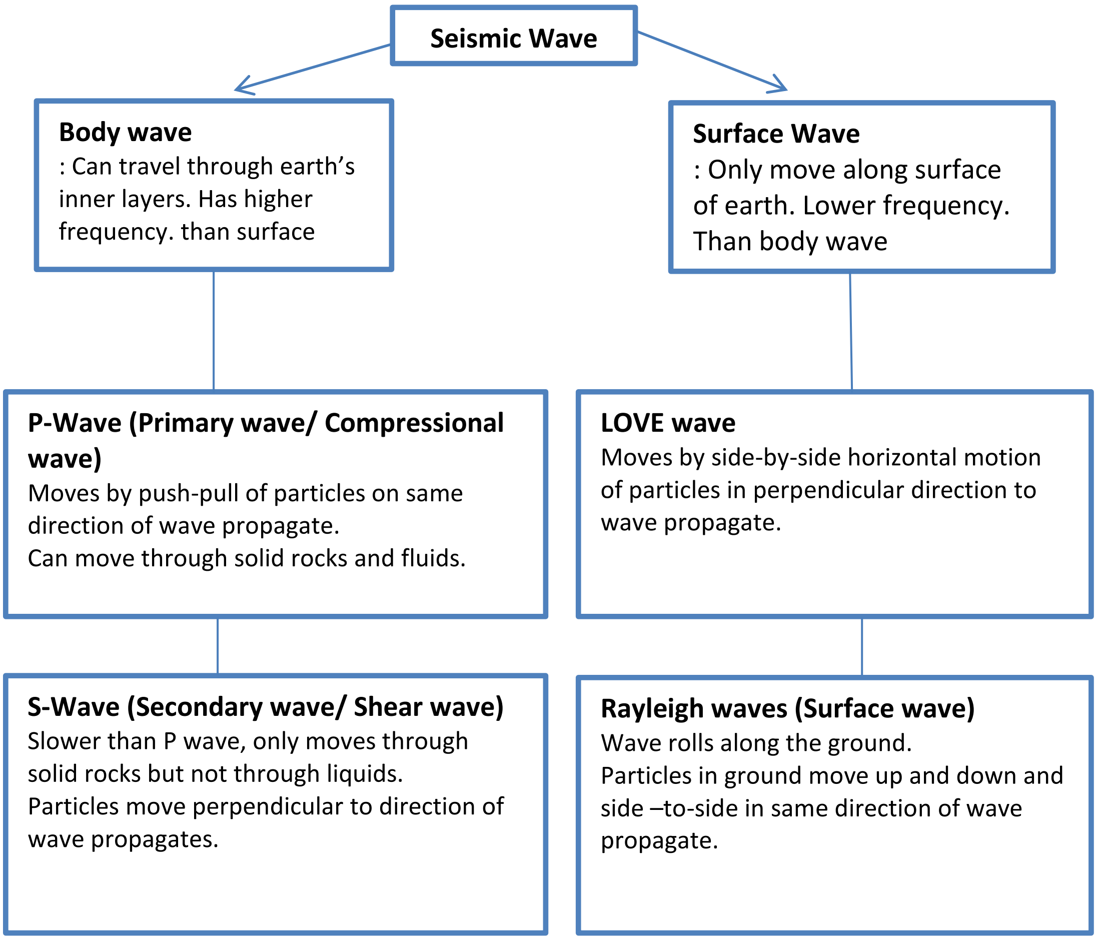
1.6 Objective of the research

In this research, DFT of seismic wave produced by individual footfall of elephant is used to extract spectral features. Time duration for DFT is limited to a single footfall. Then identifiable unique spectral features of footfalls are recognized. Then, those recognized features are used for real-time detection of elephant footsteps in an incoming seismic signal read by a geophone. The real-time identification algorithm is run on a MCU.

CHAPTER 2: BACKGROUND

2.1 Seismic waves

Seismic waves are energy carrying vibrations travel through earth. They can be generated naturally by earthquakes, volcano eruption, magma movements, landslides, or can be generated by man-made explosions, mechanical vibrations on ground. Seismic waves are categorized as follows;



: Can travel through earth's inner layers. Has higher frequency. than surface

: Only move along surface of earth. Lower frequency. Than body wave

P-Wave (Primary wave/ Compressional wave)
Moves by push-pull of particles on same direction of wave propagate.
Can move through solid rocks and fluids.

LOVE wave
Moves by side-by-side horizontal motion of particles in perpendicular direction to wave propagate.

S-Wave (Secondary wave/ Shear wave)
Slower than P wave, only moves through solid rocks but not through liquids.
Particles move perpendicular to direction of wave propagates.

Rayleigh waves (Surface wave)
Wave rolls along the ground.
Particles in ground move up and down and side –to-side in same direction of wave propagate.

67% of impact energy is carry by Rayleigh waves and they diminish in intensity of radius^{-1}. 26% of energy is carried by shear waves and 7% is carried by compressional wave which diminish in radius^{-2} [16].

CHAPTER 3: SIMILARITY SEARCH IN SEQUENCES

3.1 Introduction to time series

Seismic signal acquired in this research is ADC value sequence over time. The task is to identify similarity between previously known value sequences of elephant footfall within captured data. Hence it would be helpful to define the background theory of signal similarity search.

Time series data is sequence of observations over time;

$X = (x_1, x_2, x_3, \dots x_n)$

$x_i \in R^d$, Where $i = 1,2 \dots n$ can be single (d=1) or multidimensional vector.

If X and Y are such sequences given to find similarity, it is importance to define measurement to discriminate the similarity, which often referred as distance between two time series. The most used distance function is Euclidean distance [19].

$$D_{Euclidean}(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Due to higher dimensionality of x_i, direct processing of such raw data to compute distance between sequences would be expensive in time and storage costs. Hence several dimensionality reduction techniques have been developed. Thus f features from data sequences are extracted to represent them in f-dimensional feature space. If F is a such feature extraction fiction, distance between X and Y sequences in reduced space is;

$$D_{Feature}(F(X), F(Y))$$

To guarantee that feature extraction does not result any false dismissals, distance in feature space should match or underestimate the actual distance between two sequences, this is referred as lower bounding lemma.

Lemma 1: To guarantee no false dismissal for range queries, feature extraction function $F(\ )$ should satisfy following formula;[19][20]

$$D_{Feature}(F(X), F(Y)) \leq D_{Euclidean}(X, Y)$$

The measured *distance* between sequences is considered *metric* if it follows following four properties.

1. $D_{Feature}\big(F(X), F(Y)\big) \geq 0$, Non negativity
2. $D_{Feature}\big(F(X), F(Y)\big) = 0$ if and only if $F(X) = F(Y)$, Identity
3. $D_{Feature}\big(F(X), F(Y)\big) = D_{Feature}\big(F(Y), F(X)\big)$, Symmetry
4. $D_{Feature}\big(F(X), F(Y)\big) \leq D_{Feature}\big(F(X), F(Z)\big) + D_{Feature}\big(F(Z), F(Y)\big)$, Triangle inequality

Using metric function is desired since triangle inequality can be used to index measured distances for speed-up search. (eg. GEMINI: Generic Multimedia Indexing)[21]

There are two ways of similarity matching in time series;

1. Whole matching: Given collection of N time sequences $S_1, S_2, \dots S_N$ each of length n and query sequence of Q of same length n. Then the whole matching is done to fined sequences within distance ϵ from Q. Note that both data and query have same data length.
2. Subsequence matching: Given collection of N time sequences $S_1, S_2, \dots S_N$ each of various length n_i , $0 < i < N$; and short query time sequence Q of length $n_q < n_i$ Subsequence matching is finding sequences within distance ϵ from Q and matching start point would be anywhere within any sequences[19][20][21].

3.2 Matric distances

Including Euclidean distance, there are there several other metric distances measuring ways applicable to compare time series data Euclidean space. Consider X, Y time series of length n and x_i , y_i are ith element of corresponding series.

Euclidean distance

$$D_{Euclidean}(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Manhattan Distance

$$D_{Manhattan}(X,Y) = \sum_{i=1}^{n}|x_i - y_i|$$

Maximum distance

$$D_{Max}(X,Y) = \max_{0<i\leq n}|x_i - y_i|$$

Minkowski distance

$$D_{Minkowski}(X,Y) = \sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p}$$

Where p is called the order of Mikowski distance. In fact Manhattan distance is when p=1 and Euclidean distance is when p=2 while maximum distance is when p=∞

Mahalanobis distance

$$D_{Mahanalobis}(X,Y) = \sqrt{(X-Y)W^{-1}(X-Y)^T}$$

W is the covariance matrix [21].

3.3 Similarity measurement

Apart from the distance measured between sequences, similarity measure maps level of similarity to range of number, commonly use [-1, 1] or [0,1] where 1 indicates maximum similarity.

Consider 2 numbers of x and y;

$Similarity\ (x, y) = 1 - \frac{|x-y|}{|x|+|y|}$

Mean similarity defined as

$$Mean\_Simiarity(X, Y) = \frac{1}{n}\sum_{i=1}^{n} Similarity\ (x_i, y_i)$$

Root mean square similarity

$$RtMean\_Simiarity(X, Y) = \sqrt{\sum_{i=1}^{n} Similarity\ (x_i, y_i)^2}$$

Peak similarity

$$Peak\_similarity(X, Y) = \frac{1}{n}\sum_{i=1}^{n}(1 - \frac{|x_i - y_i|}{2\max(|x_i|, |y_i|)})$$

3.4 Variance and Covariance

In statistics and probability covariance and correlation are used to measure the tendency of a data set to deviate from its expected value of long-run average. Consider X time series of length n, hence x_{i} is i$^{\mathrm{th}}$ element of the series.

$$\mathrm{X} = (x_1, x_2, x_3, \ldots x_{\mathrm{n}})$$

Variance $var(\mathrm{X})$ is defined as;

$$var(\mathrm{X}) = \frac{1}{n}\sum_{i=1}^{n}[x_i - mean(\mathrm{X})]^2$$

Which,

$$mean(\mathrm{X}) = \frac{1}{n}\sum_{i=0}^{n} x_i$$

This is a measurement of how breadth data is spread around the mean. Also variation can be used as a feature to compare time series, even when their mean is different.

Also by definition, $var(\mathrm{X}) = Square\ of\ Standard\ diviation\ (X) = \ \sigma\ (X)^2$

Consider another time series Y of length n, which $\mathrm{y_i}$ is i$^{\mathrm{th}}$ element of the series. Then covariance $cov(X, Y)$ is defined as;

$$cov(X, Y) = \frac{1}{n}\sum_{i=1}^{n}[x_i - mean(\mathrm{X})][y_i - mean(\mathrm{Y})]$$

This is accommodating two series in to variance. Hence this generates a value related to how similar both series behave. When both series are higher or lower than their means together, $[x_i - mean(\mathrm{X})]$ and $[y_i - mean(\mathrm{Y})]$ have same sign, hence $cov(X, Y)$ tends to increase positively. Also $cov(X, Y)$ reduces when $[x_i - mean(\mathrm{X})]$ and $[y_i - mean(\mathrm{Y})]$ have opposite signs.

3.5 Correlation and Cross-correlation

Statistical studies Pearson's correlation coefficient, simply **correlation** coefficient $corr(X, Y)$ between two data series X, Y is defined as;

$$corr(X, Y) = \frac{cov(X, Y)}{\sigma\ (X)\ \ \sigma\ (Y)}$$

Or equally,

$$corr(X, Y) = \frac{\frac{1}{(n-1)}\sum_{i=1}^{n}[x_i - mean(\mathrm{X})][y_i - mean(\mathrm{Y})]}{\sqrt{\left(\frac{1}{n}\sum_{i=1}^{n}[x_i - mean(\mathrm{X})]^2\right)}\ \sqrt{\left(\frac{1}{n}\sum_{i=1}^{n}[y_i - mean(\mathrm{Y})]^2\right)}}$$

This is covariance $cov(X, Y)$ normalized by standard deviation of both series, $\sigma\ (X)$ and $\sigma\ (Y)$. Hence the value range of $corr(X, Y)$ is [1,-1]. This means, when both X and Y series are lower or higher than their means *together*, $corr(X, Y)$ tends to increase and achieve +1 as highest, thus characterized by highest positive correlation or similarity. As well, $corr(X,Y)$ is -1 when X and Y are negatively correlated which means they have opposite trends.

Further for *time series* analysis, query sequence Y can be shifted relative to X by t amount of time and Pearson's correlation coefficient can be modified to get time t dependent *cross-correlation* function, $xcorr_{\mathrm{XY}}(t)$.

Consider $\mathrm{X} = (x_1, x_2, x_3, \ldots x_{\mathrm{n}})$ and $\mathrm{Y} = (\mathrm{y}_1, \mathrm{y}_2, \mathrm{y}_3, \ldots \mathrm{y}_{\mathrm{m}}),\ \mathrm{n} > \mathrm{m}$

$$xcorr_{\mathrm{XY}}(t) = \frac{\frac{1}{(m-1)} \sum_{i=1}^{m} [x_{i+t} - mean(\mathrm{X}, \mathrm{t}, \mathrm{m})][y_i - mean(\mathrm{Y})]}{\sqrt{\left(\frac{1}{m} \sum_{i=1}^{m} [x_{i+t} - mean(\mathrm{X}, \mathrm{t}, \mathrm{m})]^2\right)} \sqrt{\left(\frac{1}{m} \sum_{i=1}^{m} [y_i - mean(\mathrm{Y})]^2\right)}}$$

Where,

$$mean(\mathrm{X}, \mathrm{t}, \mathrm{m}) = \frac{1}{m} \sum_{j=t+1}^{t+m} x_j$$

$$mean(\mathrm{Y}) = \frac{1}{m} \sum_{j=1}^{m} y_j$$

Variable t is called time delay or shift. The $xcorr_{\mathrm{XY}}(t)$ value gives cross-correlation between sequence Y with subsequence $\mathrm{X}_{\mathrm{t,m}} = (x_{\mathrm{t+1}}, x_{\mathrm{t+2}}, x_{\mathrm{t+3}}, \ldots x_{\mathrm{t+m}})$ in X.

Hence $xcorr_{\mathrm{XY}}(t)$ can be used to detect similarity between *known sequence* (Y) and a *subsequence* in a longer *unknown sequence* (X), which starts from $(t+1)^{\mathrm{th}}$ location. Range of t is $[(1-m),(n-1)]$. Range of $xcorr_{\mathrm{XY}}(t)$ is [1,-1]. For a particular t', $xcorr_{\mathrm{XY}}(t') = 1$ represents exact matching of known sequence Y to a subsequence in unknown sequence of X, the subsequence start at $(t'+1)^{\mathrm{th}}$ location of X and length is same to Y. Also -1 represents correlated sequences behave with opposite trends or anti-correlation. Detection instances of known sequence (query sequence) in the unknown sequence can be defined by assigning trigger level for $xcorr_{\mathrm{XY}}(t)$ with a tolerance. Detection occurrence time can be obtained by t.

Further, cross-correlation is only a similarity measurement in statistical nature of data series, which does not represent temporal and frequency characteristics of data.

Above definition of cross-correlation in statistical studies is also called as "*cross-covariance"*. But in signal processing, cross-correlation is defined differently by dropping normalizing and not reducing mean. In subsequent matching problem which Y is the query sequence that needs to find the similarity to a subsequences in the sequence X. Hence Y need to be slide along the X and obtain the similarity to subsequence (same to length Y) that start from each data point of X. For this situation, cross-correlation is defined and it's a function of l.

$$xcorr'_{\mathrm{XY}}(l) = \sum_{i=-\infty}^{\infty} x_i y_{(i-l)}$$

The index l is called shift or lag parameter [26, pp. 36]. Same to previous definition, it represents position which first element of Y is aligned relative to X. Substituting $i' = i - l$ definition can be rewrite similarly;

$$xcorr'_{\mathrm{XY}}(l) = \sum_{i'=-\infty}^{\infty} x_{(i'+l)} y_{i'}$$

It can be noted when time series $\mathrm{X} = (x_1, x_2, x_3, \dots x_{\mathrm{n}})$, limits for summation i (or i') is from 1 to n. The length of query series $\mathrm{Y} = (\mathrm{y}_1, \mathrm{y}_2, \mathrm{y}_3, \dots \mathrm{y}_{\mathrm{m}})$, m can be lower or equal to n. When m < n, Y is extended up to length n by padding zeros. As well any element out of index range (i.e. 1 to n) is considered as zero.

$xcorr'_{\mathrm{XY}}$ starts at $l = (1 - n)$ which, $xcorr'_{\mathrm{XY}}(1 - n)$ equals only to multiplication of first element of X (x_1) and last element of Y (y_n). Complete range of l is $[(1 - n), (n - 1)]$ thus $xcorr'_{\mathrm{XY}}$ has $(2n - 1)$ number of terms.

Higher similarity instances between X and Y have relatively higher $xcorr_{\mathrm{XY}}(l)$ values and relative alignment (delay or lag) of sequences can be obtained from l. This would be easy by plotting $xcorr_{\mathrm{XY}}(l)$, which is called "*Correlogram*".

Cross-correlation is used when there is a known signal or template (say sequence Y) and need to find similar sequence to it within an unknown sequence (say in X). This is also called as "*Matched filter"*. The filter coefficients of a matched filter are equal to template values. Higher the cross-correlation values, closer the similarity between template and unknown signal. In radar and sonar systems, received signal from target is delayed version of transmitted signal. Using a proper correlation function round-trip delay of the signal can be calculated thus distance to target can be obtained [27, pp. 239].

If correlated sequences are random noise, $xcorr_{\mathrm{XY}}$ doesn't show any pattern or trend. As well, when sequences have periodical waves, $xcorr_{\mathrm{XY}}$ values get periodical peaks.

Cross-correlation is also usable as a dimensionality reduction tool. Consider there are several sequences of values which represent features of a system, it is important to know cross-correlation between each and every pair of these features. Because when pair of features has high correlation, one of it can be considered as a redundant feature. Thus it can be removed without affecting the system description.

When $\mathrm{Y} = \mathrm{X}$, cross-correlation is called *auto-correlation*, $xcorr_{\mathrm{XX}}$. Auto-correlation is used to detect periodical features of sequences. When a sequence X is time shifted and aligned to a copy of itself in such a way that its periodical waves coincide, that produce higher auto-correlation value than when opposite trends of waves coincide.

CHAPTER 4: DIMENSIONALITY REDUCTION OF DATA

Following is an introduction of several dimensionality reduction techniques that obey lower bounding lemma and address several issues regarding the similarity search.

4.1 Principle component analysis (PCA)

Principle component analysis (PCA) is a commonly used method in Machine learning for dimensionality reduction, data compression, visualization and feature extraction.

PCA is achieved by first transforming data into orthonormal dimensions which have maximum data variance [35, pp.559-577]. For this purpose Singular value decomposition (SVD) is used to identify orthonormal Eigen basis of the data matrix. Thus redundant dimensions can be removed *without loss of any information*; and minimum required number of dimensions can be identified to represent the data.

Consider a column vector in R^{n} dimensional space;

$$\begin{bmatrix} \mathrm{x}_{1,\mathrm{i}} \\ \mathrm{x}_{2,\mathrm{i}} \\ \mathrm{x}_{3,i} \\ \vdots \\ \mathrm{x}_{\mathrm{n}-1,\mathrm{i}} \\ \mathrm{x}_{\mathrm{n},\mathrm{i}} \end{bmatrix}$$

Now consider there are m such column vectors. That is, $i = 1,2 \ldots, m$. This can be presented in a$(n \text{ x } m)$ matrix.

$$\mathrm{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & \cdots & x_{1,\mathrm{m}-1} & x_{1,\mathrm{m}} \\ x_{2,1} & x_{2,2} & x_{2,3} & \cdots & x_{2,\mathrm{m}-1} & x_{2,\mathrm{m}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{\mathrm{n},1} & x_{n,2} & x_{n,3} & \cdots & x_{\mathrm{n},\mathrm{m}-1} & x_{n,\mathrm{m}} \end{bmatrix}$$

This matrix can be interpreted as; rows represent dimensions (or features) and columns represent observations. Thus X has n dimensions and m observations of those dimensions.

Consider *mean of every dimensions as zero*. Hence *covariance* between j^{th} and k^{th} dimensions is $\gamma_{\mathrm{j,k}}$ =$(x_{\mathrm{j,1}}x_{k,1} + x_{j,2}\, x_{k,2} + \ldots + x_{j,\mathrm{m}}x_{k,\mathrm{m}})$.

Then consider ***covariance matrix,*** $\mathrm{C_X}$ that each element represents covariance between each and every dimensions.

$$\mathrm{C_X} = \frac{1}{(n-1)}\,\mathrm{X\,X^T} = \frac{1}{(n-1)}\begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \gamma_{1,3} & \cdots & \gamma_{1,\mathrm{n}-1} & \gamma_{1,\mathrm{n}} \\ \gamma_{2,1} & \gamma_{2,2} & \gamma_{2,3} & \cdots & \gamma_{2,\mathrm{n}-1} & \gamma_{2,\mathrm{n}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \gamma_{\mathrm{n},1} & \gamma_{\mathrm{n},2} & \gamma_{\mathrm{n},3} & \vdots & \gamma_{\mathrm{n,n}-1} & \gamma_{\mathrm{n,n}} \end{bmatrix}$$

which, $\gamma_{\mathrm{j},k}$ is the covariance between j^{th} and k^{th} dimensions (rows in X). Since

$(x_{\mathrm{i,1}}x_{\mathrm{j,1}} + x_{\mathrm{i,2}}\, x_{\mathrm{j,2}} + \ldots + x_{\mathrm{i,m}}x_{\mathrm{j,m}}) = (x_{\mathrm{j,1}}x_{\mathrm{i,1}} + x_{\mathrm{j,2}}\, x_{\mathrm{i,2}} + \ldots + x_{\mathrm{j,m}}x_{\mathrm{i,m}})$

$\equiv \gamma_{\mathrm{i,j}} = \gamma_{\mathrm{j,i}}$ Thus $\mathrm{C_X}$ is a symmetric matrix.

Diagonal represents *variance* of dimensions. Hence relatively higher variance represents more active dimension. As well relatively lower variance represents less active dimension, which may not need measured.

Consider off diagonals of $\mathrm{C_X}$which, represents *covariance* between all pairs of dimensions. Hence, relatively low covariance values in $\mathrm{C_X}$ represents *statistically independent* dimensions. As well, relatively higher covariance values represent *statistically dependent* dimension pairs. Thus one of the corresponding dimensions is redundant and can be removed.

Decision on redundancy of dimensions can be made more rigorously with PCA.

Next step is transform $\mathrm{C_X} = \frac{1}{(n-1)}\,\mathrm{X\,X^T}$ to a diagonal matrix. Hence off diagonal elements are zero thus no relation of relevant dimension pairs and they are not redundant. For this both eigenvalue decomposition and Singular value decomposition (SVD) can be used.

Consider SVD theorem for *any* matrix A ∈ $\mathrm{R^{nxm}}$;

$$\mathrm{A = U\,\delta\,V^T}$$

Where; $\mathrm{U} -$ n x n Orthogonal for real valued A matrices

$\delta -$ n x m Diagonal matrix with real positive values / singular values.

$\mathrm{V} -$ m x m Orthogonal for real valued A matrices [36, pp. 333-341]

Consider above defined dimensional space X and let $\mathrm{A} = \mathrm{X}$;

$$\mathrm{X} = \mathrm{U}\ \delta\ \mathrm{V}^{T}$$

Requirement is to transform existing dimensional space X to *orthonormal dimensional space.* Consider new dimensional space Y.

$$\mathrm{Y} = \mathrm{U}^{\mathrm{T}}\mathrm{X}$$

U^{T} is transpose of U. Now obtain the *covariance matrix of* Y,

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ \mathrm{Y}\,\mathrm{Y}^{\mathrm{T}}$$

Substitute $\mathrm{Y} = \mathrm{U}^{\mathrm{T}}\mathrm{X}$;

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ (\mathrm{U}^{\mathrm{T}}\mathrm{X})(\mathrm{U}^{\mathrm{T}}\mathrm{X})^{\ \mathrm{T}}$$

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ (\ \mathrm{U}^{\mathrm{T}}\mathrm{X}\,\mathrm{X}^{\mathrm{T}}\mathrm{U})$$

Substitute $\mathrm{X} = \mathrm{U}\ \delta\ \mathrm{V}^{\mathrm{T}}$;

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ (\ \mathrm{U}^{\mathrm{T}}\ (\mathrm{U}\ \delta\ \mathrm{V}^{T})(\mathrm{U}\ \delta\ \mathrm{V}^{T})^{\ \mathrm{T}}\ \mathrm{U})$$

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ (\ \mathrm{U}^{\mathrm{T}}\ (\mathrm{U}\ \delta\ \mathrm{V}^{T})\ (\mathrm{V}\ \delta^{\mathrm{T}}\mathrm{U}^{\mathrm{T}})\ \mathrm{U})$$

Since U is a unitary matrix $\mathrm{U}^{T} = \mathrm{U}^{-1}$ hence $\mathrm{U}^{\mathrm{T}}\mathrm{U} = \mathrm{U}^{-1}\mathrm{U} = \mathrm{I}$. Also since V is a unitary matrix $\mathrm{V}^{\mathrm{T}} = \mathrm{V}^{-1}$ therefore $\mathrm{V}^{\mathrm{T}}\mathrm{V} = \mathrm{I}$.

$$\mathrm{C}_{\mathrm{Y}} = \frac{1}{(m-1)}\ \delta\ \delta^{\mathrm{T}}$$

Since δ is a diagonal matrix, $\delta\ \delta^{\mathrm{T}} = \delta^{2}$,

$$\mathrm{C_Y} = \frac{1}{(m-1)}\ \delta^2$$

Since δ^2 is also a diagonal matrix $\mathrm{C_Y}$ is a diagonal matrix.

This results, if Y is taken as the dimensional space, its covariance matrix $\mathrm{C_Y}$ is a diagonal matrix. That means *covariance* of Y dimensional space are all zero. Therefore all dimensions in Y are *orthonormal and no redundancy*.

As a result $\mathrm{Y} = \mathrm{U^T X}$ can be used transform X to *orthonormal dimensional space* Y. U can be obtained by;

$\mathrm{X\,X^T} = (\mathrm{U}\,\delta\,\mathrm{V^T})(\mathrm{U}\,\delta\,\mathrm{V^T})^{\mathrm{T}}$

$\mathrm{X\,X^T} = (\mathrm{U}\,\delta\,\mathrm{V^T})(\mathrm{V}\,\delta^{\mathrm{T}}\mathrm{U^T})$ Since $(\mathrm{V^T})^{\mathrm{T}} = \mathrm{V}$

$\mathrm{X\,X^T} = (\mathrm{U}\,\delta\,\delta^{\mathrm{T}}\mathrm{U^T})$ Since $\mathrm{V^T V} = \mathrm{I}$, as above.

$\mathrm{X\,X^T} = (\mathrm{U}\,\delta^2\mathrm{U^T})$ Since δ is a diagonal matrix $\delta = \delta^{\mathrm{T}}$

$\mathrm{X\,X^T U} = (\mathrm{U}\,\delta^2\mathrm{U^T})\ \mathrm{U}$ Multiplying both sides by U

$\mathrm{X\,X^T U} = \mathrm{U}\,\delta^2$

Since δ^2 is diagonal matrix with eigenvalues, this can be solved as eigenvalue problem of $(\mathrm{X\,X^T})$ and U and δ^2 can be found. Also since $(\mathrm{X\,X^T})$ is a symmetric matrix, its eigenvectors should be orthogonal.

Intension of PCA is to find directions that have largest variation of covariance data (in matrix $\mathrm{XX^T}$). This can be obtained from eigenvectors (in U) corresponding to largest eigenvalues (in δ^2)

Set of data with n dimensional space, has n eigenvector. PCA transformation can be obtained by $\mathrm{Y} = \mathrm{U^T X}$. All those eigenvectors are principal components.

But only few columns in Y which derived from eigenvectors with highest eigenvalues can be selected to represent data. Selection of how many eigenvectors used defines the accuracy. Hence decision of number of eigenvector to be used may be a trial-and-error process to achieve expected accuracy of the data representation.

Steps of PCA using SVD.

Assume input data matrix X is defined above, as column vectors represents observations. Hence a row has values belong to same dimension (or feature) corresponding to all observation.

1. Calculate XX^{T}
2. Find eigenvalues of XX^{T} and then get square root of eigenvalues which are called *singular values.*
3. Compose *diagonal matrix* δ^2, with singular values in diagonal in ascending order.
4. Find eigenvectors of XX^{T} corresponding to all eigenvalues. Total of square of elements in each eigenvector should be 1. This can be achieved by dividing each eigenvector by square-root of total square of elements in the eigenvector itself.
5. Compose U by arranging all eigenvectors as columns of U. (Now since $(\mathrm{X\,X}^{\mathrm{T}}), \mathrm{U}, \delta^2$ all known integrity can be checked by $(\mathrm{X\,X}^{\mathrm{T}})\mathrm{U} = \mathrm{U}\delta^2$ equation.
6. Transform X to orthonormal basis by $\mathrm{Y} = \mathrm{U}^{\mathrm{T}}\mathrm{X}$.
7. Note any rows in Y with all elements are zero or very close to zero. This means dimensions relevant to those rows are redundant. This is the outcome if X not full-row-rank; redundant dimensions are linear combinations of other dimensions. Also note any rows with relatively smaller values than others, this is due to relevant rows in X are having values of linear combination of another rows with offset added. These dimensions (or features) also can be removed without much affect to data set [37, pp.221-229].

4.2 Discrete Fourier transformation (DFT)

DFT is the finite duration sequence of Discrete Fourier Series coefficients [28], [29, pp. 652-674].DFT decompose time series of x_t which $\mathrm{t} = 0{,}1 \ldots, \mathrm{n} - 1$ in to its frequency components of X_f which $\mathrm{f} = 0{,}1 \ldots, \mathrm{n} - 1$. n-point Discrete Fourier transformation is given by;

$$X_f = \sum_{t=0}^{n-1} x_t \exp\left(-\frac{j2\pi f t}{\mathrm{n}}\right) \quad , f = 0,1 \ldots, \mathrm{n}-1$$

Except X_0 which is a real value, X_f will be a complex number that represents *Cosine* and *Sine* waves. Magnitudes of the DFT coefficients, $|X_f|$ and their phase shifts are plotted against frequency, f to represent DFT. For *real valued* x_t, $X_f$is symmetric, hence f can be reduced to $\mathrm{n}/2$. Further since in most physical systems only first few frequencies we are interested, number of f can be further reduced by neglecting higher frequencies. Hence n data points can be represented by reduced dimensional space of frequency.

Fourier representation is the only tool to decompose frequency content of a signal hence it is an indispensable tool for signal analysis. DFT transform time domain information to frequency domain. Hence it can be used to identify features of *periodicity, harmonics, phase shift of frequencies and noise content*. As well mixed signal with different frequencies can be filtered. Physical systems are characterized by their frequency response to input signals. Therefore frequency domain analysis can be used for system identification and modeling. Signal source identification is also possible by correlating frequency properties of signals with frequency properties of physical phenomena which caused to generate them. However frequency domain does not represents temporal information. Hence it is better for *stationary signal* representation.

Energy of the sequence $E(\bar{x})$ defined as sum of energy at every point, hence

$$E(\bar{x}) = ||\bar{x}||^2 = \sum_{t=0}^{n-1} |x_t|^2$$

According to Parseval's theorem Fourier transform preserve Euclidean distance in time or frequency domain [20].

Parseval Theorem: Let $X_f$be the DFT of sequence x_t then,

$$\sum_{t=0}^{n-1} |x_t|^2 = \sum_{t=0}^{n-1} |X_f|^2$$

This means energy of the time domain is same as energy in frequency domain.

Since DFT is linear transformation and shift in time domain only change phase of Fourier coefficients (not the amplitude), according to above Parsevals theorem; [20]

$E(\bar{x}) = \left||\bar{x} - \bar{y}|\right|^2 \equiv \left||\bar{X} - \bar{Y}|\right|^2$

Computational complexity of DFT is $O(n^2)$ but this can be reduced to $O(n \log n)$ using FFT algorithm. Challenge of using DFT dimensionality reduction is selecting correct number of frequencies for accurate reconstruction of original signal [21].

4.3 Piecewise Aggregate Approximation (PAA)

Consider time series X of length , $X = (x_1, x_2, x_3, \dots x_n)$, PAA divides it in to w equal sized m_i $(1 < i \leq w)$ segments and record the mean of each segment $mean(m_i)$ that given by;

$$mean(m_i) = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} m_j$$

Hence, $PAA(X) = \{mean(m_1), mean(m_2), mean(m_3), \dots mean(m_w)\}$

The PAA method is very intuitive and time complexity for calculation is in order $O(n)$[23]. PAA is strongly competitive with DFT, SVD,APCA and DWT in reconstruction accuracy and time complexity [21][23]. Euclidean and other distance measuring techniques can be used to measure similarity between sequences in reduced PAA space.

4.4 Adaptive Piecewise Constant Approximation (APCA)

APCA divides time series in to variable length segments and record mean of each segment and index of last segment endpoint.

Consider time series X of length , $X = (x_1, x_2, x_3, \dots x_n)$, which APCA representation as

$C = (< cv_1, cr_1 >, < cv_2, cr_2 >, < cv_3, cr_3 >, \dots, < cv_M, cr_M >)$, $cr_0 = 0$

Where cr_i is the end point of i^{th} segment and cv_i is the mean of i^{th} segment.

$cv_i = mean\ (x_{cr_{(i-1)}+1}, x_{cr_{(i-1)}+2}, \ldots, x_{cr_i})$

This is same to PAA but using variable segment length hence recording endpoint index of data. Chakrabarti, K et al.(2001) [24] has proposed of $O(n\log(n))$ time complexity that convert the problem first in to wavelet compression problem which optimal solution is known, then convert solution back to ACPA representation with minor modification.

4.5 Piecewise Linear Approximation (PLA)

Consider time series S of length , $S = (s_1, s_2, s_3, \ldots s_m)$, and query to be find similarity there as $Q = (q_1, q_2, q_3, \ldots q_m)$. Then Euclidean distance between them given by;

$$D_{Euclidean}(S, Q) = \sqrt{\sum_{t=1}^{m}(s_t - q_t)^2}$$

PLA approximates time series with line segments $s_t{}' = a.t + b$ $(t \in [1, m])$ where a and b are two coefficients in a linear function such that *reconstruction error* $RecErr(S)$ of S is minimized, $RecErr(S)$is defined by Euclidean distance between approximated and actual time series

$$RecErr(S) \ = \sqrt{\sum_{t=1}^{m}(s_t - s_t{}')^2} = \sqrt{\sum_{t=1}^{m}(x_t - (a.t + b\ ))^2}$$

a and b parameters satisfy following condition;

$$\frac{\partial RecErr(S)}{\partial a} = 0, \quad \frac{\partial RecErr(S)}{\partial b} = 0$$

Solving above 2 equations gives;

$$a = \frac{12\ \sum_{t=1}^{m}\left(t - \frac{n+1}{2}\right) s_t}{n(n+1)(n-1)}\ ,$$

$$b = \frac{6\ \sum_{t=1}^{m}\left(t - \frac{2n+1}{3}\right) s_t}{n(1-n)}$$

$s_t$is the actual time series value at time t. Line $s_t{}' = a.t + b$ is well approximated since a and b are selected to have minimum reconstruction error. If the time series length is larger than query length, problem can be converted to subsequent matching. This can be done by dividing the time series in to equal lengths of m data points which query series has. Consider the time series of $X = (x_1, x_2, x_3, \dots x_n)$ which $\mathrm{n} > \mathrm{m}$, then PLA representation of X as reduced dimensionality space;

$$PLA(X) = (< a_1, b_1 >, < a_2, b_2 >, < a_3, b_3 >, \dots, < a_M, b_M >),\quad which\ \frac{n}{m} = M$$

Time complexity of computation of PLA is $O(n)$. Qiuxia Chen et al. (2007) [25] has proposed indexable lower bound distance function for PLA which calculates lower bound of the Euclidean distance between two time series in reduced PLA space.

CHAPTER 5: SYSTEM DEVELOPMENT

In the perspective of signal processing, objectives of the research can be listed as follows;

1. Record seismic signals generated by elephant footfalls for processing.
2. Identify unique features in the seismic signals of elephant footfall
3. Develop DSP algorithm to detect presence of elephant footfall signals in an incoming signal of seismic waves by matching identified features.
4. Implement the system on a DSP microcontroller based hardware for real-time detection.

These steps are in the order of they perform. Completion of each step took several attempts. Failed occasions are analyzed to find solutions. Outcome of each attempt and implemented modifications were logged for future reference. Above step 1 is described in sections from 5.1 to 5.3. Step 2, 3 and 4 are described in chapter 6.

5.1 Overview of Data acquisition system

Mechanical vibration of seismic waves in soil is converted to electrical signal by Geophone. Then the acquired signal passes through filters and amplifiers before it is fed to Analog to Digital Converter (ADC) for sampling. The sampled ADC values are saved in a memory card for later processing.

The crucial factor for data acquisition is sampling frequency. There are several factors to be concern. Jason D. Wood et al. (2005) [14] have observed frequency range of footfalls of several animals including elephants are in between 4.5Hz to 80Hz. The spurious frequency limit of the used geophone SM-4 is 180Hz. Hence bandwidth is limited to bw_c =180Hz. According to Nyquist theorem sampling frequency would be 360Hz. Therefore much higher sampling frequency $f_s = 1024$Hz is selected.

Software for the data recording system is done in Arduino environment. Considering the speed, availability of code libraries and 12bit ADC, ESP32 processor is selected to implement the system. Operating voltage of the unit is 3.3V. Figure 4 shows signal flow diagram of the data recording system.

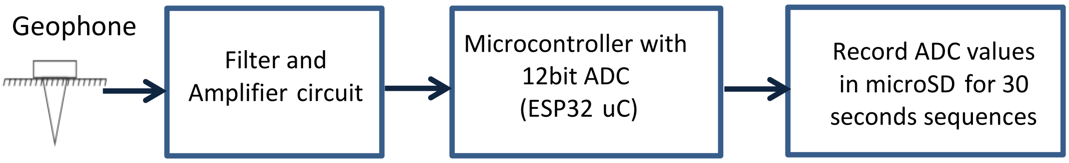

Geophone
Filter and Amplifier circuit
Microcontroller with 12bit ADC (ESP32 uC)
Record ADC values in microSD for 30 seconds sequences

Figure 4: Signal flow chart for data recoding

Hardware design of the data acquisition system is described in Appendix A.

5.2 Seismic sensor – Geophone SM-4

Properties of the used geophone are listed in table 2.

Table 2: Properties of Geophone SM-4 by INPUT/OUTPUT Inc.

Index	Parameter	Value
1	Natural frequency	8 Hz
2	Tolerance	+/- 6.3%
3	Maximum tilt angle	20 degree
4	Typical spurious frequency	>180 Hz
5	Standard resistance	375 Ohm
6	Tolerance	+/- 5%
7	Sensitivity	28.8 V/m/s
8	Tolerance	+/- 5%
9	Maximum coil excursion p.p.	2 mm
10	Moving mass	11 g
11	Diameter	25.4 mm
12	Height	32 mm
13	Weight	74 g
14	Operating temperature	-40 ^{0}C to 100 ^{0}C

5.3 Data recording sessions

I. To obtain the noise characteristics of the circuit, signal input lines were short-circuited (hence grounded) and ADC values were recorded.

II. Seismic wave data generated by elephant footfalls has been collected at Elephant safari at Habarana, Mandala maha viharaya at Ampara, and

Arawwala. At a time a single elephant was let to walk on a straight path. Geophone was fixed to ground such that it covers 5~40m range of the path.

III. Seismic signal was recorded without considering which leg of the elephant caused it.

IV. Also seismic wave data generated by several vehicles like motorbikes, three-wheels, car, van, busses and also human footfalls were recorded by placing the geophone near a roadside.

CHAPTER 6: DETECTION ALGORITHM

Recorded data was imported to MATLAB. They were plot on time domain and frequency domain to identify unique features.

6.1 Identification of circuit noise

From the circuit noise data record, Standard deviation $\sigma_{cct\_noise} = 4.9572.$ was calculated. This value was used to define signal threshold limit that starts ADC, and to distinguish signal and noise in the processing.

6.2 Digital filtering

FIR Hamming window filter was used to filter acquired data in the digital domain. Same to Analog filter 180Hz was selected as cutoff frequency and 15th order filter coefficient are obtained using MATLAB filter design toolbox [26, pp. 305-377].

6.3 Data transformation

Discrete Fourier transformation (DFT) was obtained on time domain data of 1024 samples/sec [26, pp. 59-96, 141-166]. Feature extraction and detection algorithm works basically on DFT data. Since footfall signal of several animals including elephant and human are below 100Hz, interested frequency has been limited to 100Hz in DFT.

6.4 Determination of features and extraction

Features to be extracted are decided by observing time domain plots and frequency domain plots (DFT values). Observing time domain plots, duration of a single footfall seismic wave of an elephant was identified as 300~350ms. Hence time window for a footfall is defined as 500ms (512 samples).

In othere researches which used frequency domain analysiz to identify footfalls have obtained DFT integrating several footfalls same to this. Then a template has been derived.Next this template was used to correlated with frquency data of a unkown signal to find similarity. The main drawbacks of this method are;

1. DFT is done integrating several footfall signals over the time. Hence several seconds of time is required to produce processable data. Thus it is dificult to

produce real-time output for detection of elephants (or the signal source). But this is efficient and have several advantages where real-time detection is not requrired. Eg. For elephant census [14].

2. Results of DFT correlation depends on magnitudes of frequencies of both template and unknown signal. Further, magnitudes of frequencies depend on amplitude of the signal in time domain. Since amplitude of seismic signal (Reigly wave) depends on distnace, detection of footfalls can only be done at a specific distance.

Hence instead of integrating several footfalls, DFT of individual footfalls are analyzed to develop the detection algorithm. The challenge of this method is transient signals of individual footfall are highly vulnerable to noise generated by other sources. Hence, features identification was done only considering relative features of limited bandwidths of above 3 harmonics; hence effect of noise is limited due to limited bandwidth.

Also correlation with a template was not used in time domain. Hence detection is not depending on amplitude of the seismic waves received to the geophone. This is a distinctive advantage of the detection algorithm which has not used before according to the referred researches.

Time domain plot of 10 individual footfall signals is shown in figure 5. DFT values of this signal are plotted in figure 6. Frequency domain plot of 6 data recording sessions are shown in figure 7. Time duration of these recordings are 92s, 66s, 89s, 89s, 23s and 23s consecutively for 1 to 6 spectra. However these recording sessions included noise due to natural causes like draging tree branches or coconut leaves by elephants and human walking.

Observing below 100Hz in figure 6 and 7, dominant frequency range is in between 28~35Hz. Also followings three ranges could be identified as noticeable harmonics;

1. 1st Peak range : 12Hz – 16Hz
2. 2nd Peak range : 28Hz – 34Hz
3. 3rd Peak range : 48Hz – 56Hz

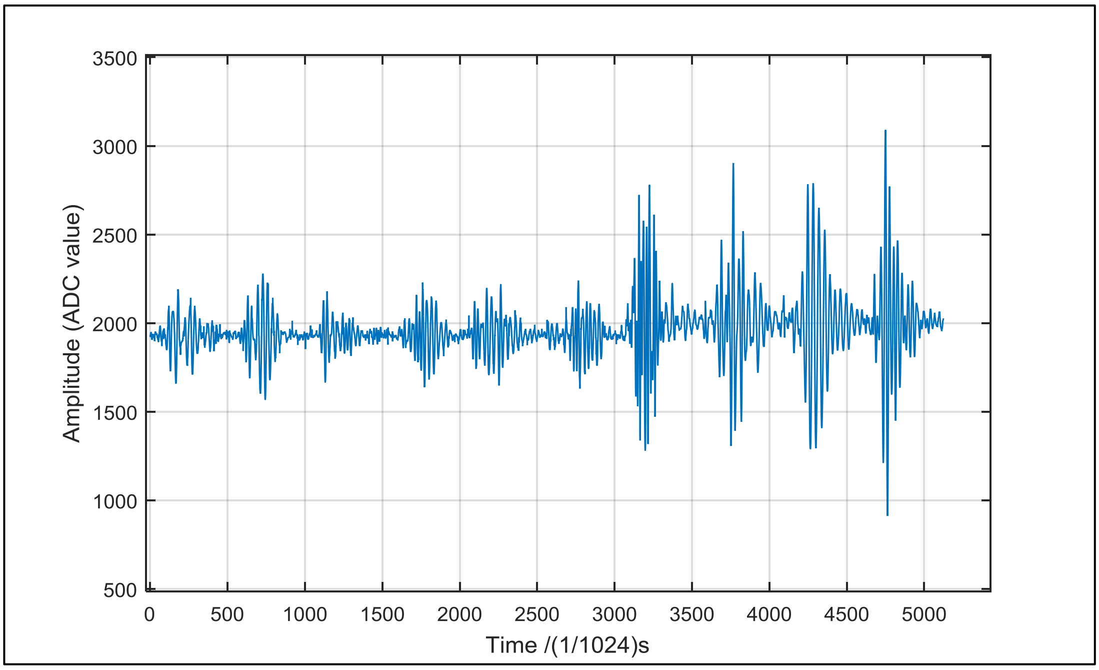


Figure 5: 10 individual elephant footfall seismic signal.

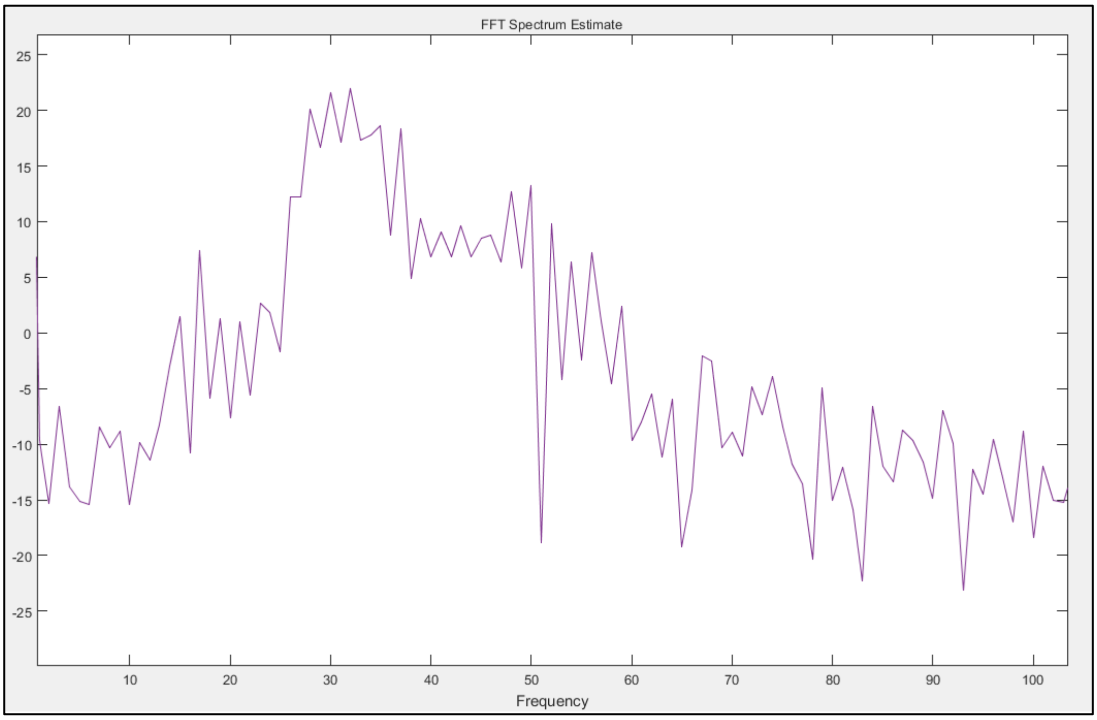
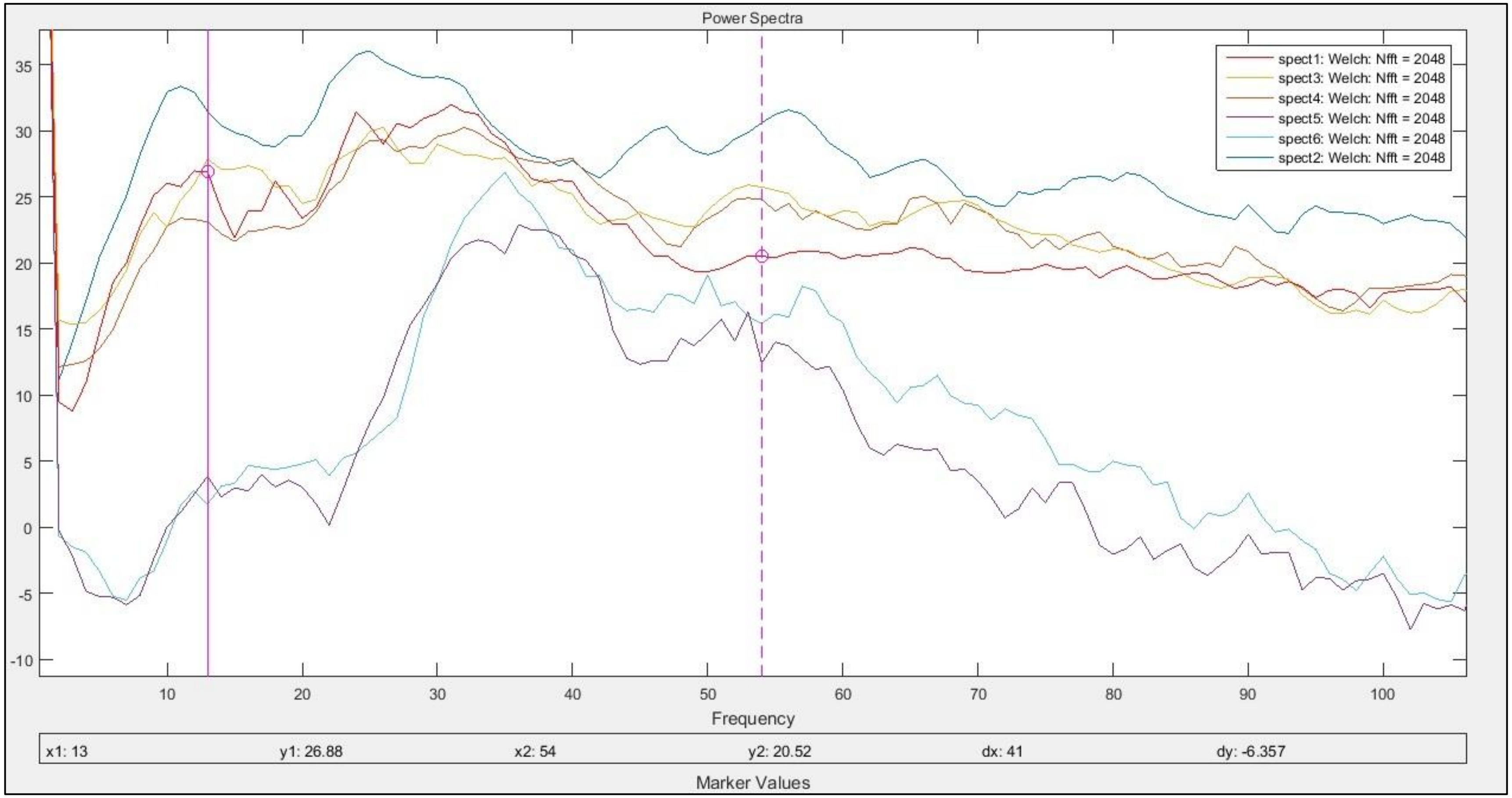


Figure 6: DFT of all 10 elephant footfall seismic signal.



x1: 13 y1: 26.88 x2: 54 y2: 20.52 dx: 41 dy: -6.357

Figure 7: 6 Power spectral plots of seismic waves of elephant walking

Among several data recordings 41 elephant footfall signals are selected for feature extraction. Selection of these signal durations was done by observing time domain plots where no signal anomalies are present. Also environmental condition prevailed in data recoding was considered to select low noise durations

Next, MATLAB code has been written [26,pp. 1-54, 141-166] to extract 12 features from 41 *individual* elephant footfall signals [34, pp. 22-55]. Description of features is shown in table 3. MATLAB code is shown in Appendix E – (I).

Table 3: Description of extracted feature variables.

Index	Feature description	Variable name in MATLAB code	Column number in Appendix B table
1	Number of crossing over a dynamic-threshold level	cutin_count	2
2	Ratio of differences of averaged 3 “Harmonic ranges”	avg_ratio	3
3	Frequency corresponding to maximum DFT value	dft_maxf	4
4	Center of gravity frequency of DFT plot	cog	5
5	Ration of (dft_maxf/cog)	dft_maxf_cog	6
6	Time duration of signal is higher than threshold value	trig_width	7
7	Average difference between raw and filtered signal	filt_diff	8
8	Difference of minimum and maximum signal levels	minMax	9
9	Ration of (minMax/filt_diff)	minMax/filt_diff	10
10	Ration of (minMax/st_div)	minMax/st_div	11
11	Ration of (minMax/trig_width)	minMax/trig_width	12
12	Standard deviation of signal	st_div	13

In order to facilitate further identification of harmonic ranges and any patterns, locations of "Peak frequencies in DFT" and "DFT values" from 2Hz to 100Hz are also listed. Obtained values for each template is shown in Appendix B table 9.

6.5 Dimensionality reduction of data

In order to identify uncorrelated features, correlation matrix for 12 features was obtained. Figure 8 shows color-map of feature correlation matrix. Table 10 in Appendix C shows corresponding values.

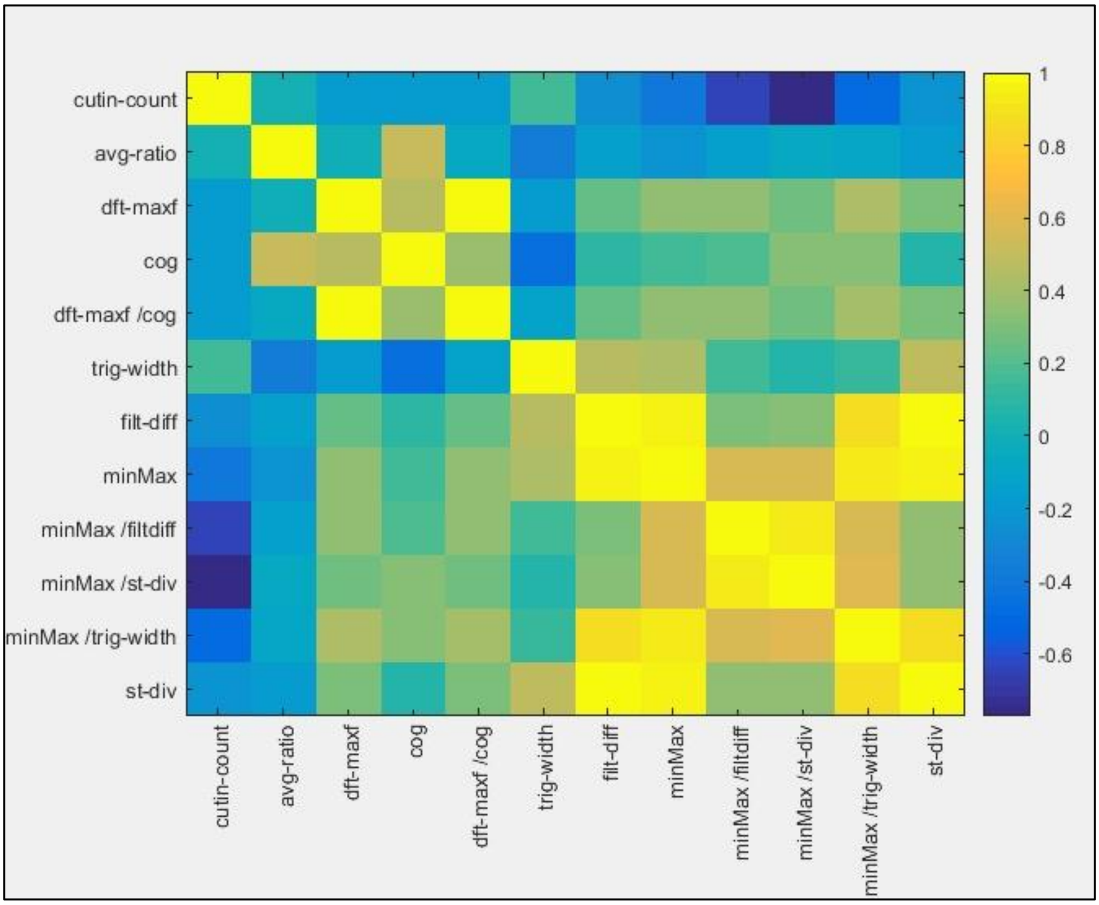


Figure 8: Color map of correlation matrix of features.

Due to high correlation with other features, following 5 features are eleiminated;

1) Dft_maxf (retain dft_maxf /cog)
2) Filt_diff (retain minMax)
3) minMax (retain minMax /trig_width)
4) minMax /filt_diff (retain minMax/st_div)
5) minMax /trig_width (retain st_div)

In order to make detection independent of distance between geophone location and elephant footfall location; features which depend on signal amplitude are removed. Thus, “trig_width”, “minMax /trig_width” and “st_div” are also removed. Hence only following 4 features are selected for system model.

Table 4: Selected feature variables for elephant footfall detection.

Index	Feature variable name in MATLAB code	Index according to feature definition in Table 3
1	Cutin_count	2
2	Avg_ratio	3
3	cog	5
4	dft_maxf /cog	6

Then all 41 footfall signals are used to extract above 4 features, thus 41x4 data matrix “X” was made. This data matrix X was used used to get singular value decomposition (SVD) matrixes ($X = U\,\delta\,V^T$). In order to perform principal component analysis (PCA), X is then tranformed to Y ($Y = U^T X$) which is on orthogonal dimentional space. Values obtained for Y is shown in table 11 in Appendix D. Hence it can be noticed 4th dimension of Y can be disregarded on further processes due to lower values it has.

After removing 5 possible outliers which are reading number 3, 4, 38, 40 and 41 in Y, following ranges of 4 dimesions in Y could be identified in sequence.

1) Min -35.9601, Max -31.9174
2) Min -6.4991, Max 3.9564
3) Min -5.1217, Max 3.7233
4) Min -0.2139, Max 0.8552

6.6 Developing an Algorithm

6.6.1 Use of Principal component analysis (PCA)

Value ranges of dimensions of Y in above (section 6.5) are used to detect footfall signals. Seismic signals in 500ms time durations were captured and 4 feature values were computed. These values are then transformed (by U^{T}) to orthogonal dimension. If these transformed values are within above ranges of Y, the signal is classified as detection.

But this algorithm fails due to high false detections on seismic signals generated by other sources like vehicles, humans and other animals. First 3 transformed feature values are plotted on 3D space with 41 templates which used to develop the model; and found they are closely placed. Hence different algorithm had to be developed.

6.6.2 Use of Support vector machine (SVM)

Classification of acquired signal as an elephant footfall or not is based on extracted 4 feature values shown in Table 4 (section 6.5). Hence this problem can be solved as a binary classification in Machine learning. Also there are 41 template signals of elephant footfalls and several records of various other seismic signal sources. Hence SVM algorithm can be used to train an algorithm to do the classification.

The training data matrix ‘datasvm’ has been composed appending 4 feature values of several data recordings as shown Table 5. 2-D and 3-D Plots of data in any combination of features shows classification could not be done with a simple hyperplane. So, nonlinear SVM was used with Gaussian kernel.

Table 5: Composition of SVM training data set.

Index	**Signal description**	**MATLAB signal record name**	**# data sequences of 500ms**	**Class**
1	Elephant footfall	t40info_4	30	+1
2	2 Cattles and 2 dogs run	cattles2dog	60	-1
3	A girl walks on a tar road	girlwalk_1	34	-1
4	A person walks on bare ground	humanwalk_1	32	-1
5	A motor bike	mbike_1	18	-1
6	Tractor on a tar road	tractor2	65	-1
7	Two three-wheels then a lorry	Two3W_BigLorry	56	-1
8	Two motor bikes then two three-wheels	TwoMbikes_3W	50	-1
	Total number of data points		**345**	

Out of 41 templates of elephant footfall templates, only 30 randomly selected feature vectors were used as training data, remaining 11 were used as test data. MATLAB code for the SVM classification and prediction is shown in Appendix E – (V). Slack variables are not introduced hence hard-margin classification was obtained. Tests were done with every possible combination of 2, 3 and all 4 features to obtain maximum prediction accuracy over test data. Through that, it was found use of 'cutin_count' and 'avg_ratio' features maximize prediction accuracy.

The figure 9 illustrates training data set with SVM decision boundary and support vectors. As shown in figure 10, the closer view of *largest decision region* depicts boundary for 'cutin_count' (Feature 1) and 'avg_ratio' (Feature 2) are 5 ~ 10 and 0.8~3.4 respectively.

This fact helps to develop computationally much simpler classification algorithm than SVM which can be implemented with less computational power. This may leads to compromise the detection accuracy. But requirement of implementing classification algorithm on a portable hardware with low power and low resources compels to develop algorithm further.

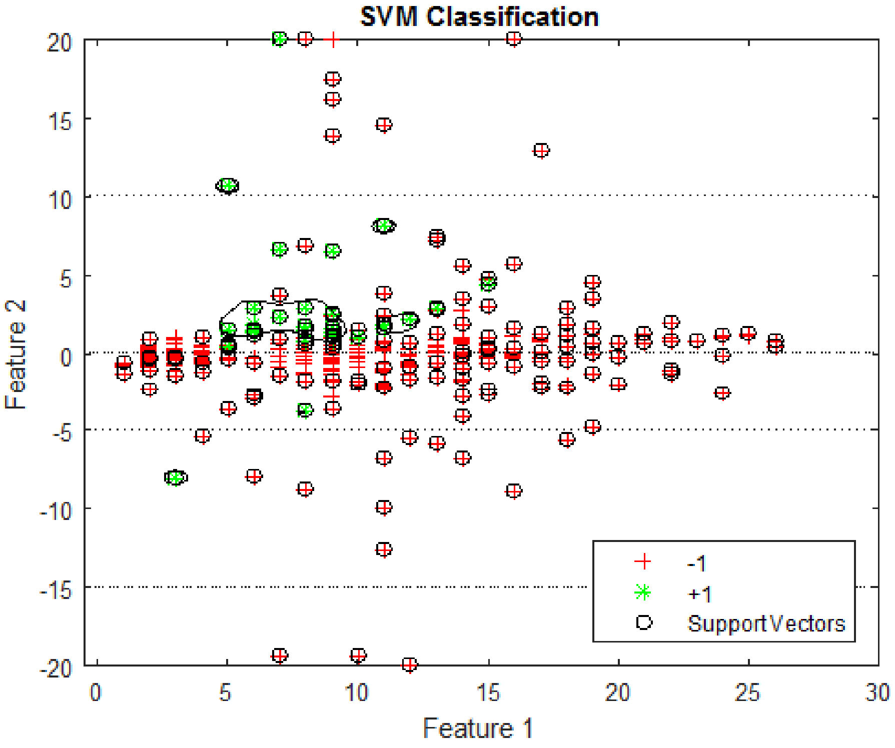


Figure 9: SVM classifier decision boundary.

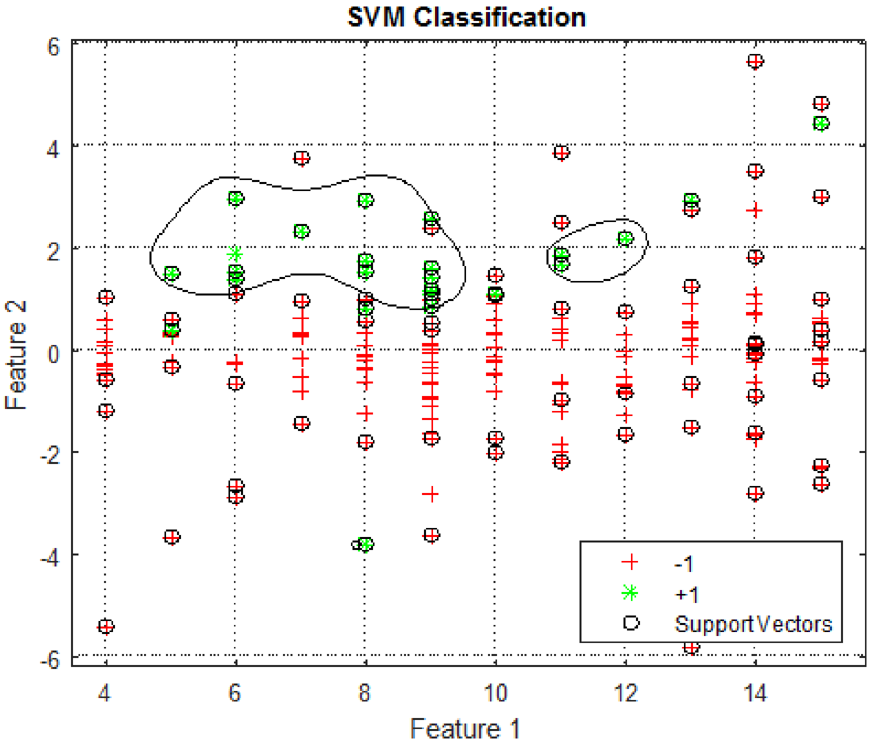


Figure 10: SVM classifier – Details of largest decision region.

6.6.3 Use of relative harmonics content

According to the table 9 in Appendix B, considering only first 10 readings it can be noticed that except one outlier number 6, highest DFT coefficients are in frequency 28Hz, 32Hz, 32Hz, 34Hz, 34Hz, 38Hz, 28Hz, 28Hz and 40Hz respectively. Average of them is 32.6Hz.

1. Previously in figure 7, it was noticed that there are three noticeable peak ranges of DFT values in 12 – 16 Hz, 28 – 34 Hz and 48 – 56 Hz. This is more prominent in figure 6 which DFT values of 10 templates. If these three ranges are arranged on ascending order of their *average* DFT coefficients, it is 1st , 3rd and 2nd ranges. O*rder of average DFT values* of these three ranges is selected as the key feature of elephant footfalls because for any other signal sources this was not observed.

2. As a time domain feature to eliminate false detection due to waves longer than single footfall duration, a dynamic-threshold level was defined as follows;

   (1/6)(Signal maximum – Signal minimum) + Signal mean

   Signal crossing instances over this level is counted. In Appendix B table 9 column “cutin_count” represents this counter and it is in between 5 to 9 inclusive.

3. A minimum level for signal fluctuation is defined to avoid false detection due to mixed seismic waves of several sources at far. This level is defined as 8 x Normal seismic signal noise level recorded in quite environment at night, which is 50 ADC value.

Above 3 features were used as the detection algorithm. A MATLAB code has been written based on them and it is shown in Appendix E - II, III & IV. Since ultimate objective is to implement the algorithm in a microcontroller, above feature 2 and 3 were selected avoiding much higher time complexity calculations like standard deviation, center of gravity of DFT plot. This would make the output faster; reduce power consumption and cost of required signal processor. However DFT calculation cannot be avoided.

6.7 Parameter optimization

In order to verify effectiveness of algorithm described in section 6.6.3, the written MATLAB code (in Appendix E - II, III & IV) has been tested to maximize the detection of elephant footfalls while minimizing false detections.

1. In order to minimize false detection in seismic waves due to other sources, first and third frequency ranges were required to change to 14 – 20Hz and 46 – 52Hz respectively. This was done while keeping maximum detection in 10 templates. However this makes algorithm only detect 9 waves out of 10 templates. Seventh template couldn't be detected. Apparently, because it has anomaly of maximum DFT value at 62Hz. This is out of the range 28 -40 Hz which all other templates fall in. Hence template 7 was avoided considering under the influence of higher frequency noise.

2. After testing the algorithm with several other data recordings of elephant footfalls, it was noted upper limit of the dynamic-threshold crossing instances in time domain was required increase up to 10.

3. Further testing of algorithm *with noise data* of vehicles and human walking showed minimum requirement for signal fluctuation can be reduced to 200 while maintaining false-detections minimum.

   Hence any seismic signal received to the system higher than 200 ADC value is processed to identify elephant footfalls. Since ADC is 12bit, this is 4.9% of ADC full range.

CHAPTER 7: RESULTS

7.1 Detection of the elephant footfalls

Results generated by MATLAB code discussed in section 6.6.3 and 6.7 are shown in table 6 and 7. Table 6 shows detection in data recordings of elephant walking under several environmental noise conditions. Figure 11 to 16 shows time domain plots of 1,2,3,4,7 and 8th data recordings in table 6. Detected sub subsequences of elephant foot falls are marked in red color.

According to table 6, "Average footfall interval" increases up to 6s as environmental noise increases (recording number 2, 4, 9). But even when seismic waves of elephant footfalls and human footfalls are mixed, detection could be done below 3s intervals (recording no 3, 7, 8). Similarly, when there was no artificial seismic noise sources, elephants could be detected in less than 3s (recording number 5 and 6). Thus mixing of human footfall signal has no significant effect on detection. Normally fastest speed of elephant walk is 6m/s [35], thus considering 3s average detection time, elephant can be identified as it walks 18m away.

According to the detected subsequence plots (Figure 11 to 16) it can be noted that detection is independent from amplitude of the signal. As algorithm processes transient signal of an individual footfall waves, this is a remarkable feature of the developed algorithm.

7.2 False detection

Table 7 shows false-detection for various other seismic wave sources. Figure 17 to 20 shows the signal data and false-detected subsequences. They correspond to 7, 8, 9 and 11th data recordings in table 7. False detection rate over time is notably low for common seismic wave sources which may appear near elephant's habitats.

Table 6: Results of elephant footfall detection

Recording No	Signal description	Location	MATLAB Variable name	Number of data samples	Duration /s	# Dynamic - threshold exceeds (c1)	Number of detection	Detection rate /per second	Average footfall interval /s	Remarks
1	Elephant walks with a person	Habarana	data1	94323	92	168	16	0.1737	5.8	
2	Elephant walks dragging coconut leave with a person	Habarana	data2	68073	66	136	17	0.2557	3.9	High ADC saturation
3	Elephant walks with a person	Habarana	data3	91262	89	177	34	0.3815	2.6	
4	Elephant walks dragging coconut leave with a person	Habarana	data4	91262	89	181	16	0.1795	5.6	High ADC saturation
5	Elephant walks	Habarana	data_13_5_1	24018	23	22	10	0.4263	2.3	
6	Elephant walks	Habarana	data_13_6_1	24018	23	36	8	0.3411	2.9	
7	Elephant walks with a human	Arrawwa	Elpt7466	61412	60	102	28	0.4669	2.1	
8	Elephant walks with a human	Arrawwa	Elpt7533	61412	60	107	30	0.5002	2.0	High ADC saturation
9	Elephant walks lot of people around to watch in a temple	Ampara	ph_walk1	272337	266	529	53	0.1993	5.0	High ADC saturation

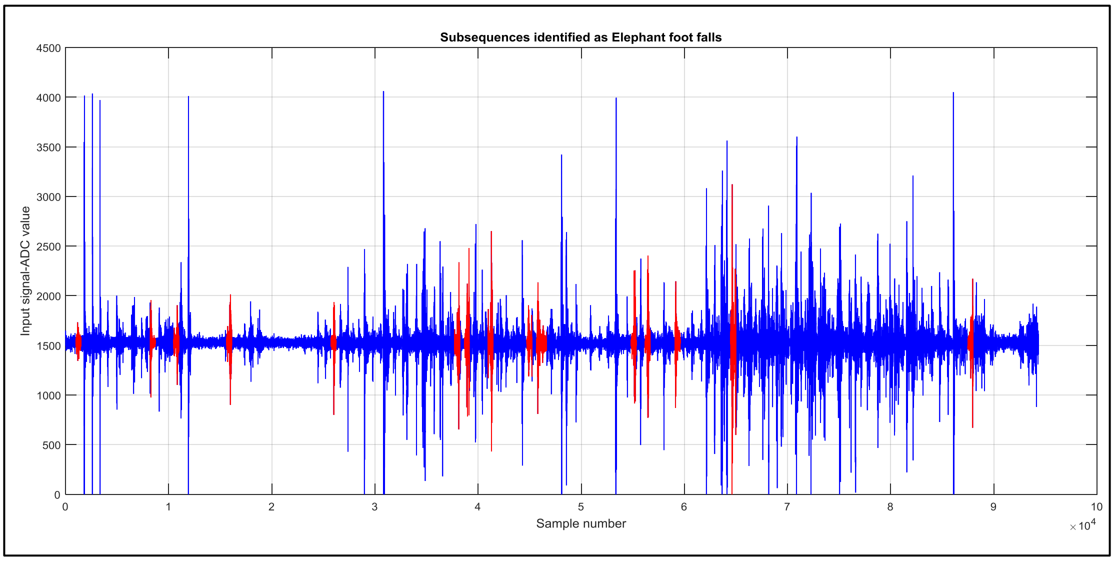


Figure 11: Time domain signal plot of data recording 1 of table 6. Detected subsequences of elephant footfalls are in red color

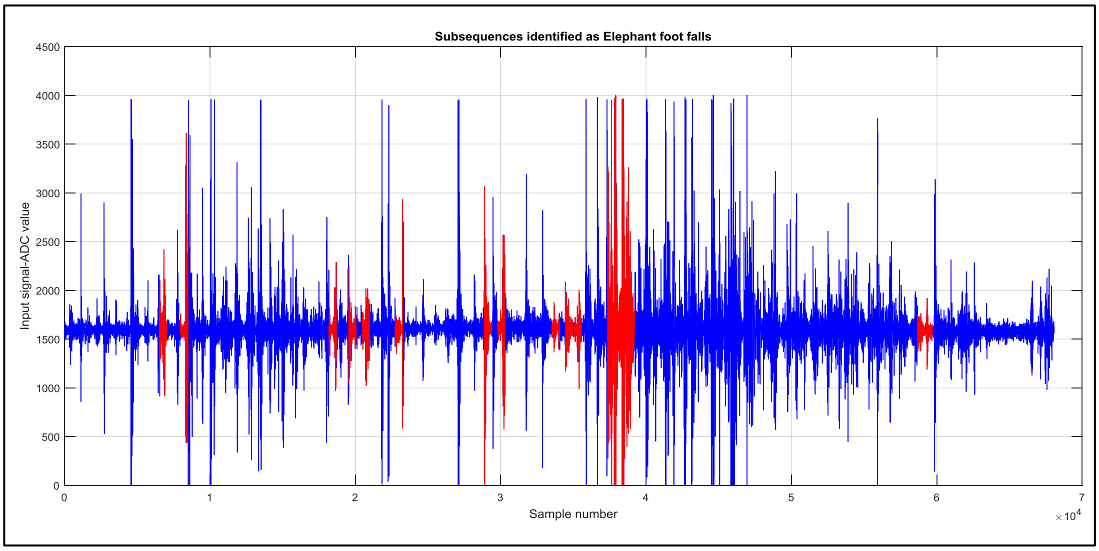


Figure 12: Time domain signal plot of data recording 2 of table 6. Detected subsequences of elephant footfalls are in red color

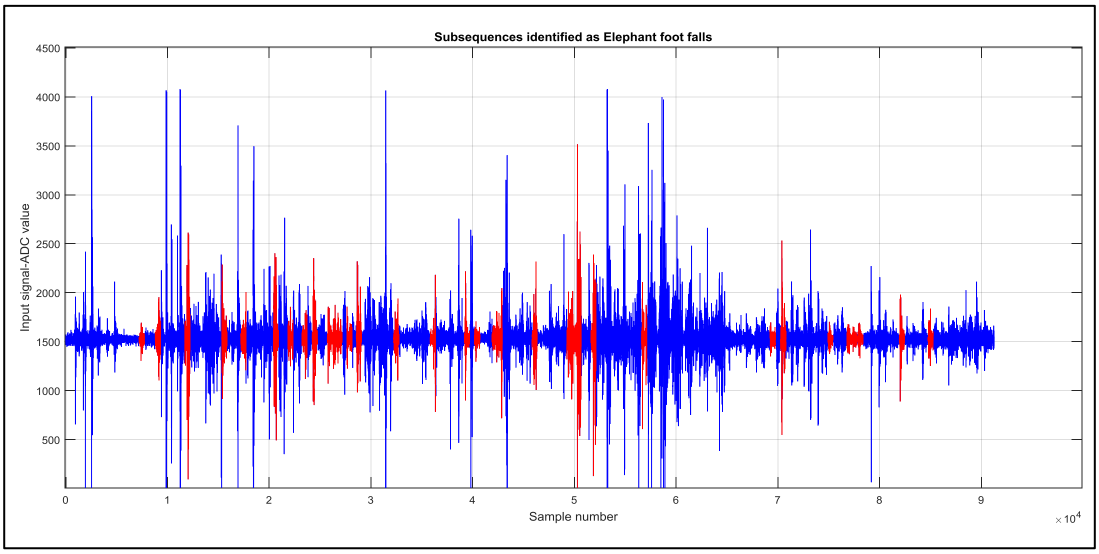


Figure 13: Time domain signal plot of data recording 3 of table 6. Detected subsequences of elephant footfalls are in red color

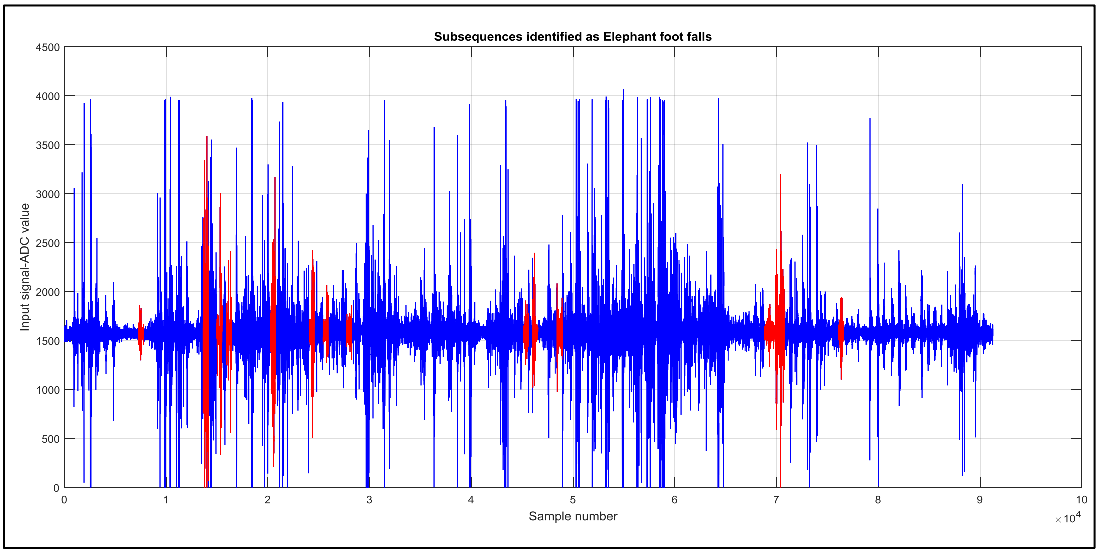


Figure 14: Time domain signal plot of data recording 4 of table 6. Detected subsequences of elephant footfalls are in red color

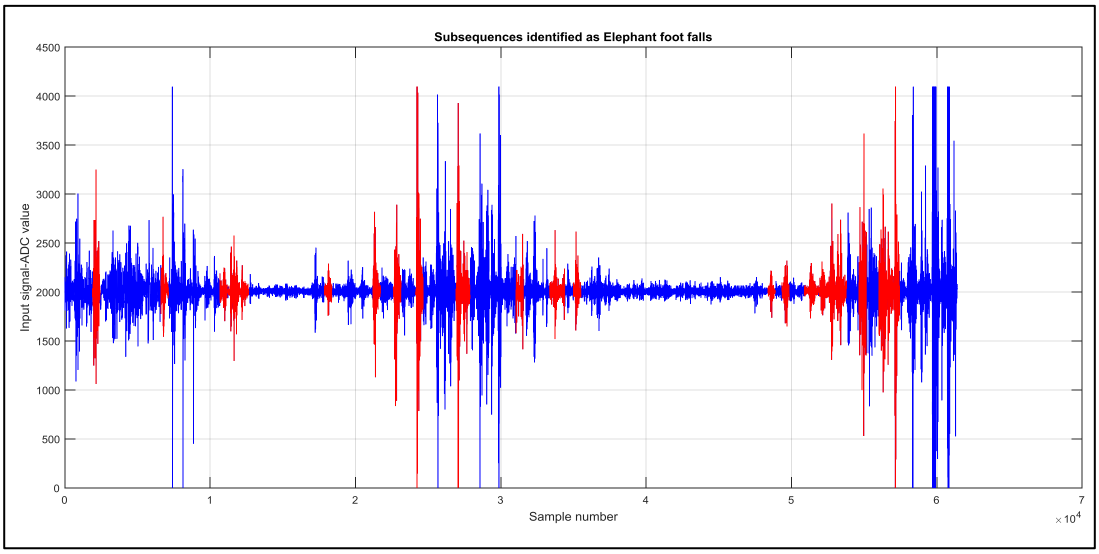


Figure 15: Time domain signal plot of data recording 7 of table 6. Detected subsequences of elephant footfalls are in red color

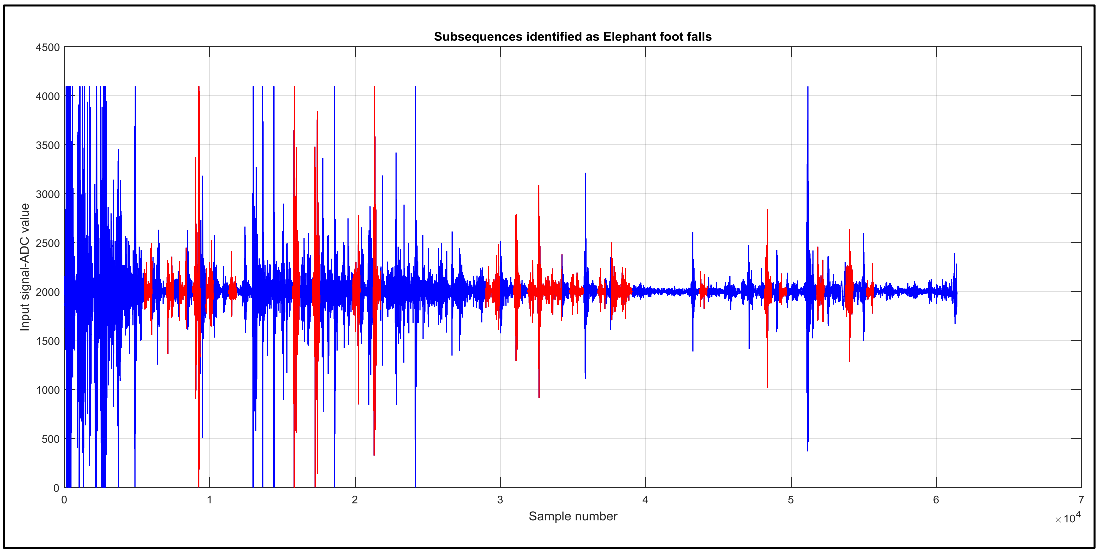


Figure 16: Time domain signal plot of data recording 8 of table 6. Detected subsequences of elephant footfalls are in red color

Table 7: False-detection of elephant footfalls

Index	Signal description	Location	MATLAB Variable name	Number of data samples	Duration /s	# Dynamic-threshold exceeds (c1)	Number of false detection	Detection rate /per second	Average fales detection interval /s	Remarks
1	A girl walks on a tar road	Ampara	girlwalk_1	20481	20	30	0			
2	A person walks on bare ground	Ampara	humanwalk_1	20481	20	22	0			
6	A person walk near, motor bike at far	Ampara	Walk_WithMbikes	30707	30	44	1	0.0333	30.0	
3	A person walk then a orry and a motor bike, minor rain	Ampara	walkLoryMbike_rain	30707	30	41	0			
4	Tractor on a tar road	Ampara	tractor2	30707	30	65	0			
5	Two three-wheeles then a lorry	Ampara	Two3W_BigLorry	30707	30	52	0			
7	Two motor bikes and two threewheels	Ampara	TwoMbike_3W	30707	30	35	1	0.0333	30.0	
8	A motor bike	Ampara	mbike_1	10241	10	16	1	0.1000	10.0	
9	A threewheel	Ampara	threewheel_1	10241	10	15	1	0.1000	10.0	
10	A pearson walk minor rain	Ampara	walk40foot_rain	30707	30	31	0			
11	Two cattles and two dogs run closely	Ampara	cattles_2dog	30707	30	49	2	0.0667	15.0	

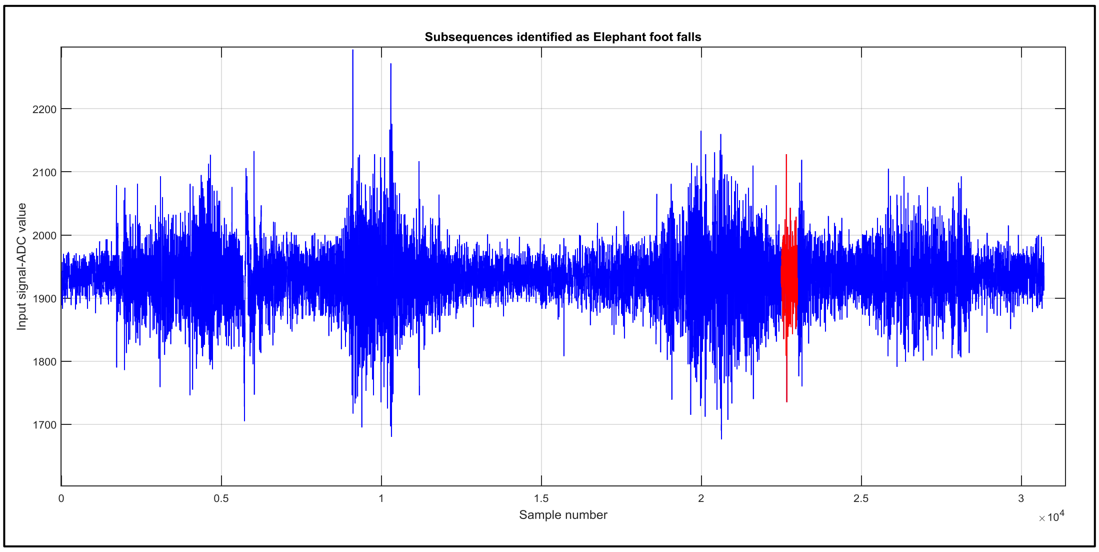


Figure 17: Time domain signal plot of data recording 7 of table 7. False-detected subsequences of elephant footfalls are in red color

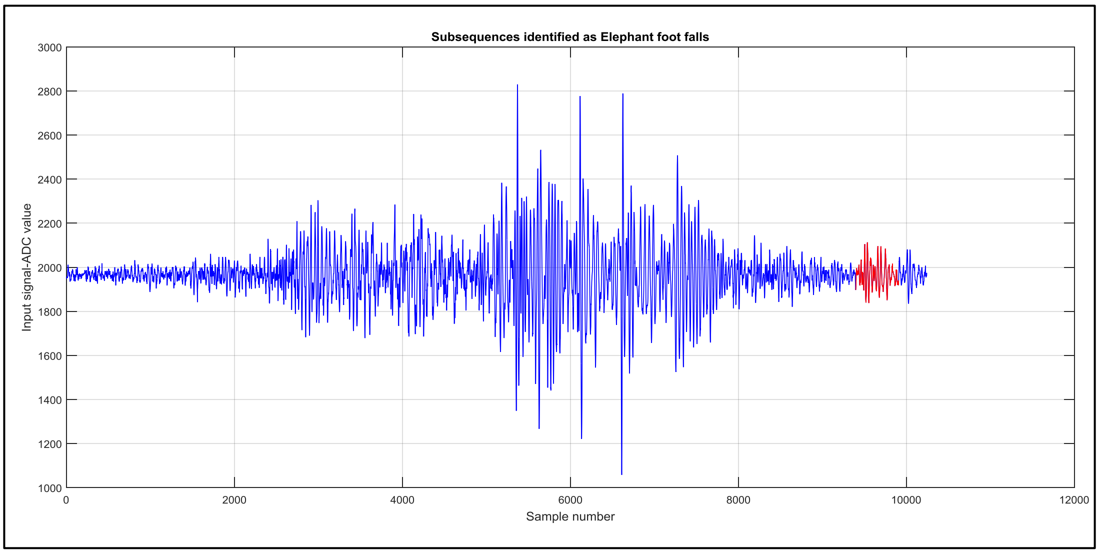


Figure 18: Time domain signal plot of data recording 8 of table 7. False-detected subsequences of elephant footfalls are in red color

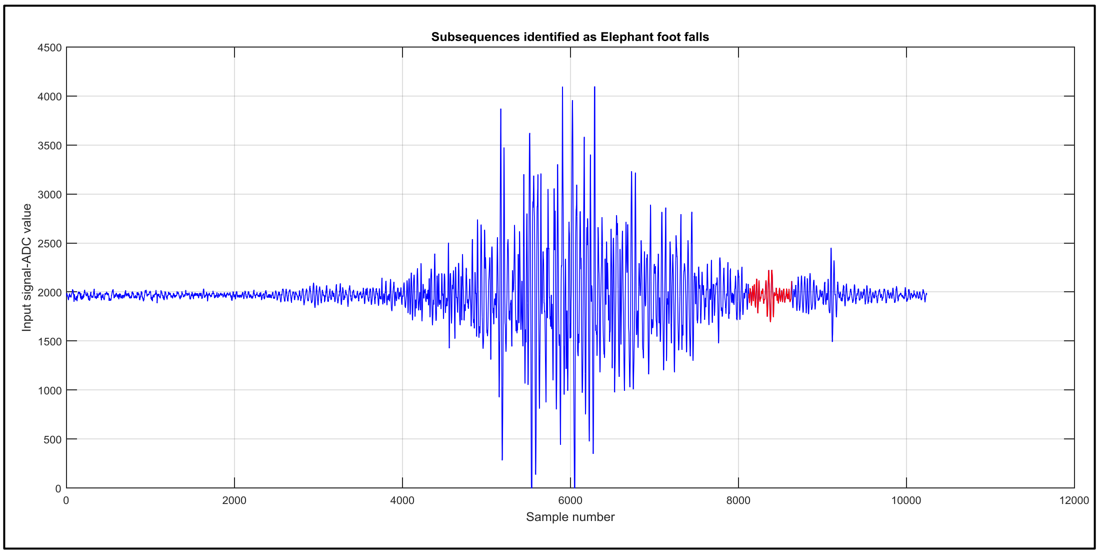


Figure 19: Time domain signal plot of data recording 9 of table 7. False-detected subsequences of elephant footfalls are in red color

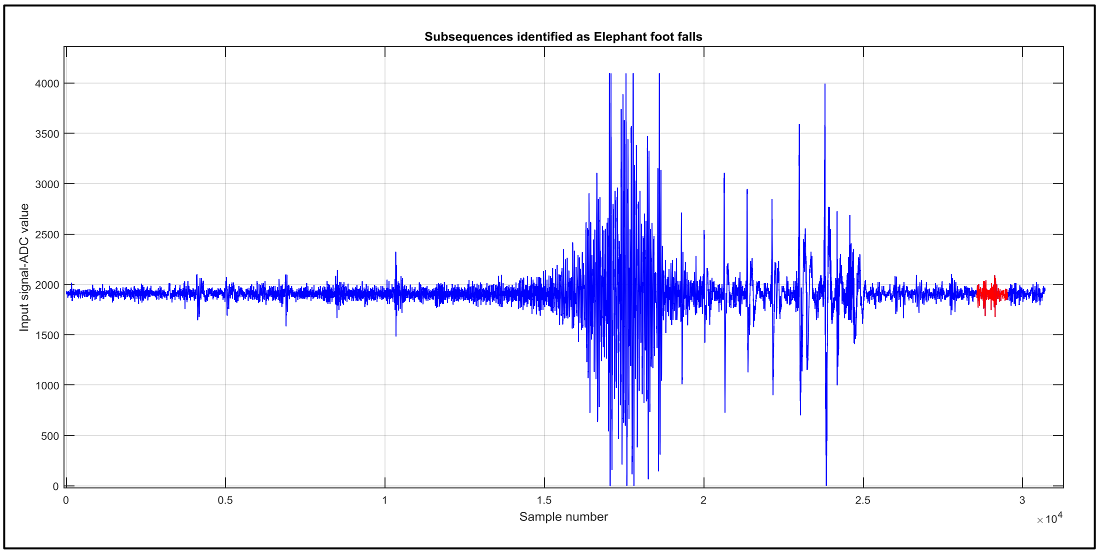


Figure 20: Time domain signal plot of data recording 11 of table 7. False-detected subsequences of elephant footfalls are in red color

CHAPTER 8: CONCLUSION

8.1 Detection of elephants

It has been shown that transient seismic waves generated by individual footfalls of elephants can be distinctly identified. Hence the research could show significant new possibility of elephant detection using seismic waves. A SVM classifier has been trained successfully for classification and detects elephant footfalls.

Further a novel algorithm has been developed based on relative strength of harmonics contents and used successfully for the detection. For an elephant in 40m vicinity this algorithm could detect elephant footfalls in average interval of 6s when there is high seismic noise. This interval reduces to 3s when there noise gets low.

8.2 Query sequence identification in a time series

For time series data, a query sequence identification method has been introduced which doesn't depend on magnitude of data in time or frequency domains. The concept used in algorithm of this research can be further developed such that, dividing complete DFT frequency range in to subsequences and arrange them according to average DFT coefficient. Similarity matching can be done by matching the arranged order of frequency ranges. This can be used as a scale invariant similarity search method for *whole matching* of sequences.

8.3 Portable algorithm

The developed algorithm (section 6.6.3) is possible to implement in a 32bit DSP microcontroller for real-time detection of seismic waves due to elephant footfalls.

8.4 Possibility of further developments

8.4.1 Range of detection

When signal saturates ADC, detection rate gets low. Hence minimum distance to locate geophone should be identified. The observed range of elephant detection is about 40m, but need more site tests to be carried out to establish correlation between distance and detection rate. Thus further measures can be taken to improve the detection range.

8.4.2 Rate of detection

As well the relation between rate of detection and pair of feet (front/rear) of the elephant should be analyzed. Apparently the developed system is sensitive to both pairs, but need more test to identify properties of seismic waves of them separately. Then algorithm can be modified to increase rate of detection.

8.4.3 Effect of soil condition

Also propagation of Rayleigh waves depend on soil condition, therefore further tests should be done in different soil conditions to identify how range of detection is depend on soil condition.

8.4.4 Implementation of algorithm on a microcontroller

Developed algorithm should be implemented on a microcontroller and warning system for remote users should be designed. Also a proper power source should be selected and it should be low cost and require minimize human intervene. The complete system should be tested at sites for ample time duration to define the efficiency and usefulness of the system.

8.4.5 Identify different seismic wave sources

Capabilities of the developed system can be further extend by identifying correct parameter values of the features introduced in this research. Hence system can be incorporated to identify several other seismic wave sources, like other animals, humans, vehicles movements, gun firing. Multiple source identification in a single system is also possible.

8.4.6 Implement with Analog electronics

In this research, signal processing has been done in digital domain. But the exploited key feature in frequency domain can be implemented using band-pass analog filters. Designing hardware circuitry completely in analog mode may lead to several new possibilities and advantages.

REFERENCES

1. “The first island wide national survey of elephants in Sri Lanka 2011”, Department of wildlife Conservation (DWC). http://www.dwc.gov.lk/Aoldsite/index.php/en/downloads.

2. Fernando, Prithiviraj & Pastorini, Jennifer. (2011). “Range-wide status of Asian elephants.” Gajah. 35. 15-20. 10.5167/uzh-59036

3. http://www.dwc.gov.lk/Aoldsite/index.php/en/component/content/category/97-protected-areas

4. Fernando P & Leimgruber P (2011) Asian elephants and dry forests. In: The Ecology and Conservation of Seasonally Dry Forests in Asia. McShea WJ, Davies SJ, Phumpakphan N & Pattanavibool A (eds) Smithsonian Institution Scholarly Press. pp. 151-163

5. Prithiviraj Fernando, Elephants in Sri Lanka: past present and future

6. Charles Santiapillai, Prithiviraj Fernando and Manori Gunawardene, “A strategy for the conservation of the Asian elephants in Sri Lanka”, Journal of the IUCN/SSC Asian Elephant Specialist Group. Number 25: 91–102, 2006

7. Prithiviraj Fernando, Jayantha Jayewardene, Tharaka Prasad, W. Hendavitharana and Jennifer Pastorini, “Current Status of Asian Elephants in Sri Lanka.” – 2011

8. J. Hutchinson, "The locomotor kinematics of Asian and African elephants: changes with speed and size", Journal of Experimental Biology, vol. 209, no. 19, pp. 3812-3827, 2006. Available: 10.1242/jeb.02443.

9. Nakandala, M. S., Namasivayam, S. S., Chandima, D. P., & Udawatta, L. (2014). Detecting wild elephants via WSN for early warning system. 7th

International Conference on Information and Automation for Sustainability. ***(IR detection)***

10. Payne, K. B., Langbauer, W. R., & Thomas, E. M. (1986). “Infrasonic calls of the Asian elephant (Elephas maximus)”. Behavioral Ecology and Sociobiology, 18(4), 297–301. ***(Volume 18, issue 4)***

11. Sayakkara, A. P., Jayasuriya, N., Ranathunga, T., Suduwella, C., Vithanage, N., Keppitiyagama, C., … Voigt, T. (2017). Eloc: Locating Wild Elephants Using Low-Cost Infrasonic Detectors. 2017 13th International Conference on Distributed Computing in Sensor Systems (DCOSS).

12. Sugumar, S. J., & Jayaparvathy, R. (2014). An Improved Real Time Image Detection System for Elephant Intrusion along the Forest Border Areas. The Scientific World Journal, 2014, 1–10.

13. O’Connell-Rodwell, C. E., Arnason, B. T., & Hart, L. A. (2000). Seismic properties of Asian elephant (Elephas maximus) vocalizations and locomotion. The Journal of the Acoustical Society of America, 108(6), 3066–3072.

14. Jason D. Wood, O'Connell-Rodwell, C.E., Simon L. Klemperer, Using Seismic Sensors to Detect Elephants and Other Large Mammals: A Potential Census Technique. Journal of Applied Ecology, Vol. 42, No. 3 (Jun., 2005), pp. 587-594

15. G. Succi, D. Clapp, R. Gampert, and G. Prado, “Footstep detection and tracking,” in Unattended Ground Sensor Technologies and Applications III, vol. 4393 ofProceedings of SPIE, pp. 22–29, April 2001.

16. Succi, G. P., Prado, G., Gampert, R., Pedersen, T. K., & Dhaliwal, H. (2000). Problems in Seismic Detection and Tracking. Unattended Ground Sensor Technologies and Applications II.

17. Houston, K. M., & McGaffigan, D. P. (2003). Spectrum analysis techniques for personnel detection using seismic sensors. Unattended Ground Sensor Technologies and Applications V.

18. Koç, G., & Yegin, K. (2013). Footstep and Vehicle Detection Using Slow and Quick Adaptive Thresholds Algorithm. International Journal of Distributed Sensor Networks, Volume 2013.

19. Faloutsos, C., Ranganathan, M., & Manolopoulos, Y. (1994). Fast subsequence matching in time-series databases. Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data - SIGMOD '94.

20. Agrawal, R., Faloutsos, C., & Swami, A. (1993). Efficient similarity search in sequence databases. Lecture Notes in Computer Science, 69–84

21. Cassisi, C., Montalto, P., Aliotta, M., Cannata, A., & Pulvirenti, A. (2012). Similarity Measures and Dimensionality Reduction Techniques for Time Series Data Mining. Advances in Data Mining Knowledge Discovery and Applications.

22. Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., & Keogh, E. (2008). Querying and mining of time series data. Proceedings of the VLDB Endowment, 1(2), 1542–1552.

23. Keogh, E., Chakrabarti, K., Pazzani, M., & Mehrotra, S. (2001). Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. Knowledge and Information Systems, 3(3), 263–286.

24. Chakrabarti, K., Keogh, E., Pazzani, M., & (2001). Locally adaptive dimensionality reduction for indexing large time series databases. ACM Transaction on Database Systems, Vol 27, No. 2, June 2002, 188-228.

25. Qiuxia Chen, Lei Chen, Xiang Lian, Yunhao Liu, Indexable PLA for efficient similarity Search. International Conference on Very Large Data Bases (VLDB),Vienna, Austria, September 2007.

26. V. Ingle and J. Proakis, *Digital signal processing using MATLAB*. Stanford: Cengage Learning, 2012.

27. S. Kuo, B. Lee and W. Tian, Real-Time Digital Signal Processing: Fundamentals, Implementations and Appl, 3rd ed. John Wiley & Sons, 2013.

28. A. Oppenheim, A. Willsky and S. Nawab, Signals and systems, 2nd ed. Noida-(India): Pearson, 2018.

29. A. Oppenheim and R. Schafer, Discrete-time signal processing, 3rd ed. Noida, (Índia): Pearson, 2017.

30. Texas Instruments Application Report (2013), AN-31 Op Amp Circuit Collection, SNLA140B

31. W. Jung, Op Amp applications handbook. Burlington, MA: Newnes, 2005.

32. B. Carter and R. Mancini, Op amps for everyone, 3rd ed. Burlington, MA: Newnes, 2009.

33. Texas Instruments Application Report. (2007). Noise Analysis in Operational Amplifier Circuits, SLVA043B.

34. P. Stoica and R. Moses, Spectral analysis of signals. Upper Saddle River, NJ: Pearson Education, 2005.

35. Bishop, C. (2006). Pattern recognition and machine learning. New York, NY.: Springer.

36. Rogers, S. and Girolami, M. (2012). A first course in machine learning. Boca Raton: CRC Press.

37. Marsland, S. (2009). Machine Learning: An Algorithmic Perspective. Boca Raton: CRC Press.

38. Hamel, L. (2009). Knowledge discovery with support vector machines. Hoboken, N.J.: John Wiley & Sons.

Appendix A: Design of signal amplifier -filter circuit and ADC for data recording

I. Gain calculation

Consider oscillation of the geophone at its natural frequency 8Hz and 1000th of its maximum excursion 2mm (refer Table 2);

$Total\ coil\ path\ for\ 1s\ duration = 8Hz \text{ x } 2 \text{ x } (2\text{mm}/1000) = 0.032\text{mm}$

Then *average velocity* over complete oscillation is 0.032mm/s. (If sine wave oscillation is considered maximum velocity is 0.05mm/s) For that oscillation generated average voltage;

$V_{G\_8Hz} = 0.032 \text{ mm/s x } (Geophone\ sensitivity)$

$V_{G\_8Hz} = 0.032 \text{ mm/s x } (28.8 \text{ V/m/s}) = 921.6\ \mu\text{V}$

Consider $921.6\ \mu\text{V}$ is given in full-scale of ADC 3.3V, hence required minimum gain

$G_{Amp} = 3.3V \ / \ 921.6\ \mu\text{V} = 3580$

Therefore selected gain is $G_{Amp} = 4000$.

Also it was intended to maintain noise below least significant bit (LSB) sensitivity of the ADC. Noise voltage of the geophone is mainly thermal noise given by;

$$V_{Noise} = \sqrt{4kTBR}$$

Which, $k -$ Boltzmann constant, $T -$Temperature in K, $B -$ Bandwidth in Hz and

$R -$Equivalent resistance of the sensor

Hence thermal noise of geophone at 35℃ is;

$V_{Noise\_G} = \sqrt{4 \text{ x } 1.38064852 \text{ x}10^{(-23)} \text{ x } 308.15 \text{ x } 512\text{Hz x } 375 \text{ Ohm}}$

$V_{Noise\_G} = 57.16\ nV$

To find LSB sensitivity of ADC; consider full scale of ADC is 0~3.3V and resolution of ADC 12-bit

$$Sensitivity_{LSB\_ADC} = \frac{3.3\,V}{(2^{12}-1)} = 805.9\,uV$$

Assuming amplifier gain $G = 4000$, amplifier input signal equivalent for LSB

$$Sensitivity_{LSB\_Input} = \frac{805.9\,uV}{4000} = 201.4\,nV$$

It can be noted $Sensitivity_{LSB\_Input} > V_{Noise\_G}$. Hence it can be adequately assumed that thermal noise of the geophone will not affect ADC values.

II. Circuit configuration

Amplifier and filter circuit was designed with following features [30], [31 pp. 3-95, 173-282];

1. Non-inverting configuration was used to have highest input impedance.
2. Included offset adjustment (RV3) to compensate OP IC offset and make it possible to keep mean of the signal to mid of ADC range 3.3V/2.
3. In order to save power, microcontroller with ADC wakes up only when output of the circuit exceeds adjustable threshold voltage. Hence a comparator is included to generate analog-level-trigger signal.
4. In order to use low bandwidth (less than 0.5MHz) OP amplifier ICs, two stage amplifiers has been used. First stage gain $g_{amp1} = 57$ (Resistor values 56k, 1k) and second stage gain $g_{amp2} = 77.9$ (Resistor values 100k, 1.3k)
5. R-C filters are used as anti-aliasing filter, in between amplifier stage and front end of ADC. This leads to reduce the cost and to use commonly available components. Since circuit bandwidth is 180Hz, initially for all filter stages cutoff frequency was set to 194Hz selecting 8.2kOhm resistors and 0.1uF capacitor.

$$f_c = \frac{1}{2\pi RC} = \frac{1}{2\pi \text{ x } 8200 \text{ x } 0.1 \text{ x } 10^{-6}} = 194 \text{ Hz}$$

III. Selection of Operational amplifier IC

Following parameters of OP amplifier ICs were concerned;

1. Gain Bandwidth product (GBP) of IC $>$ 10 x (Circuit bandwidth 180Hz x gain of amplifier stage)
2. Offset voltage below 500uV; hence gain of 4000 will not saturate output in 3.3V power supply.
3. Operate between 2V-6V.
4. CMRR >95 dB
5. For long battery life, Input current per OP amplifier < 100 uA. Theoretically, this makes 2000mAh battery system can power 4 OP amplifier (equal to 400 uA) circuit more than 200 days. This is except the power consumption of microcontroller which contributes most power demand. However this was the most critical parameter since low noise OP amplifier ICs tends to draw higher current (and expensive).

OP amplifier parametric search in manufactures websites were used to search ICs. There were 6 selected potential ICs ; AD8607, TLV2252, OPA2333, OPA2336, LTC6078, OPA2317. For them noise generation in the circuit was calculated [32, pp. 163-188],[33]. Calculated results are shown in Table 8 (including cost and current consumption as on datasheet).

Table 8**:** Results of calculated output noise of circuit for different ICs.

OP amplifier IC		**AD8607**	**TLV2252**	**OPA2333**	**OPA2336**	**LTC6078**	**OPA2317**
Noise voltage	**uV**	4834.1	1590.6	3680.0	3275.6	1492.0	3680.0
ADC full scale	**V**	3.3	3.3	3.3	3.3	3.3	3.3
ADC bandwidth		4095	4095	4095	4095	4095	4095
Voltage Error of full-scale	**%**	0.15	0.05	0.11	0.10	0.05	0.11
ADC resolution	**uV / LSB**	805.9	805.9	805.9	805.9	805.9	805.9
ADC noise error	**LSB**	6.0	2.0	4.6	4.1	1.9	4.6
SNR	**dB**	56.7	66.3	59.1	60.1	66.9	59.1
Cost per IC	**$**	4.20	4.12	1.80	7.92	4.40	4.26
Current for 4 OP amp	**uA**	200	140	112	128	288	140

Lowest output noise is with LTC6078, but it is expensive. OPA2333 has the lowest price and lowest power consumption. Hence most economical solution is it. Also it has fairly good SNR of 59.1 dB, thus OPA2333 is selected for the amplifier circuit.

IV. Circuit schematic and simulation

Circuit schematic is shown figure 21 and 22. According to the simulation (including OP ICs) the circuit had cutoff frequency lower than expected $f_c = 194$ Hz, hence resistor values in filters were lowered and obtained $f_c = 173$ Hz. Then simulated frequency response is shown in figure 23. Gain of the circuit is 72.8 dB (=4365).

V. Verification of design parameters

The prototype circuit was assembled on a PCB. The design of PCB is shown in figure 24 and 25. A 5Hz sine wave was injected at the circuit input and observed output was 3.18V. Then frequency was increased until cutoff amplitude $2.22V$ ($\cong 3.18 \text{ x } 0.707$) was achieved which occurred at 168Hz. Hence simulated value of $f_c = 173$ Hz has been sufficiently achieved.

VI. ADC and the microcontroller

Output voltage offset of the amplifier-filter circuit was set to mid of ADC range that is about 1.65V. This can be done by adjusted with RV1 potentiometer which is in the 2nd stage of amplifier (OP amplifier U1:B) and before the buffering stage (OP amplifier U2:A). This minimizes amplified signal exceeding ADC voltage range 0~3.3V.

The microcontroller wakes-up and starts ADC only when Analog-level-trigger signal goes high. Analog-level-trigger is a comparator output which continuously compares output voltage of the amplifier-filter circuit with a preset threshold voltage level. This threshold voltage can be adjusted with RV2 potentiometer. In the event seismic waves caused to output voltage higher than threshold level, Analog-level-trigger goes high. After several trial and errors, this threshold voltage was set to 1.95V. This cause to triggers the circuit for human footfalls at about 5m distances. This triggering method alternatively used with IR sensor that makes it easy for sometimes to manually trigger the circuit using an IR remote.

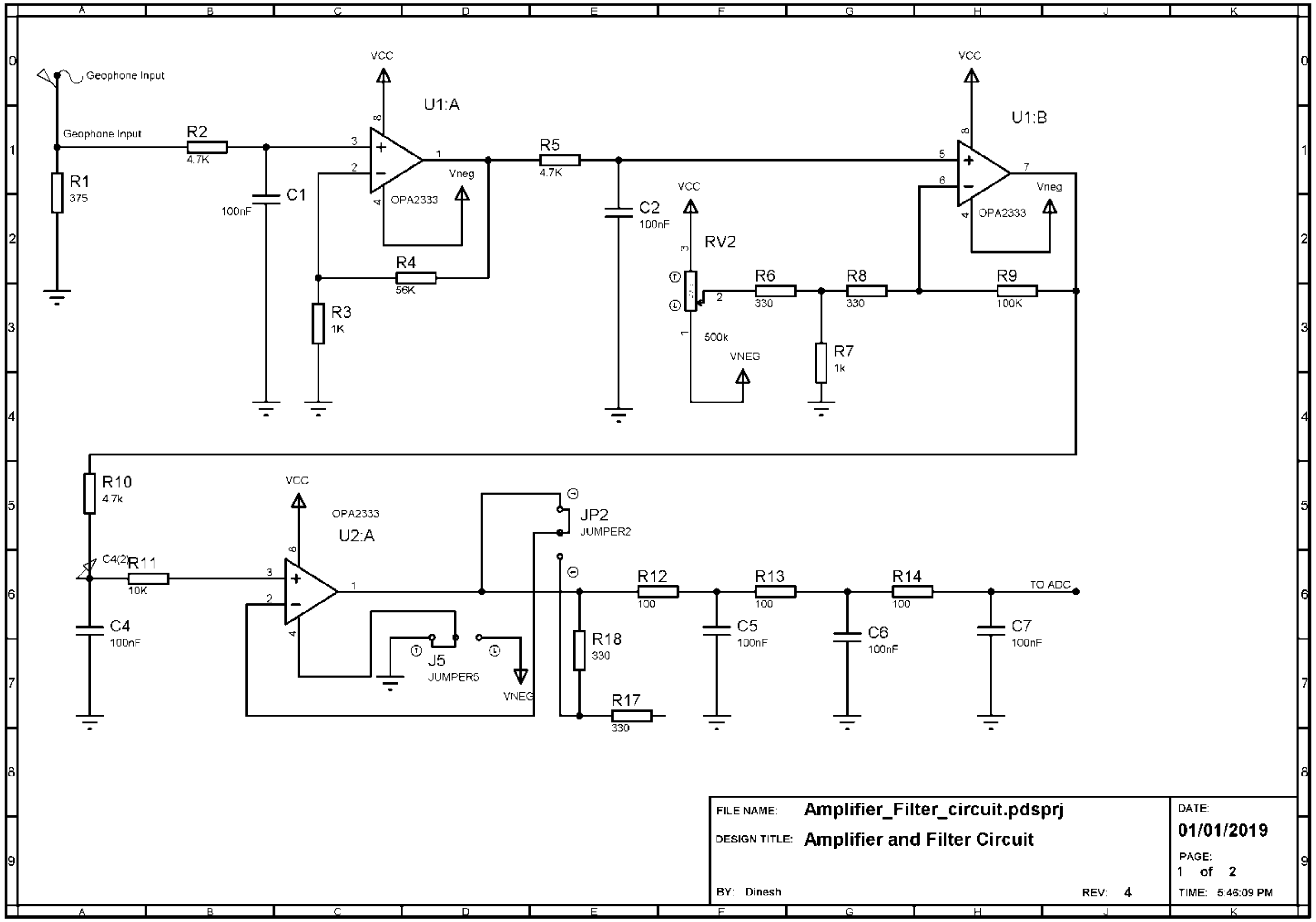


FIGURE 21: Circuit schematic 1/2

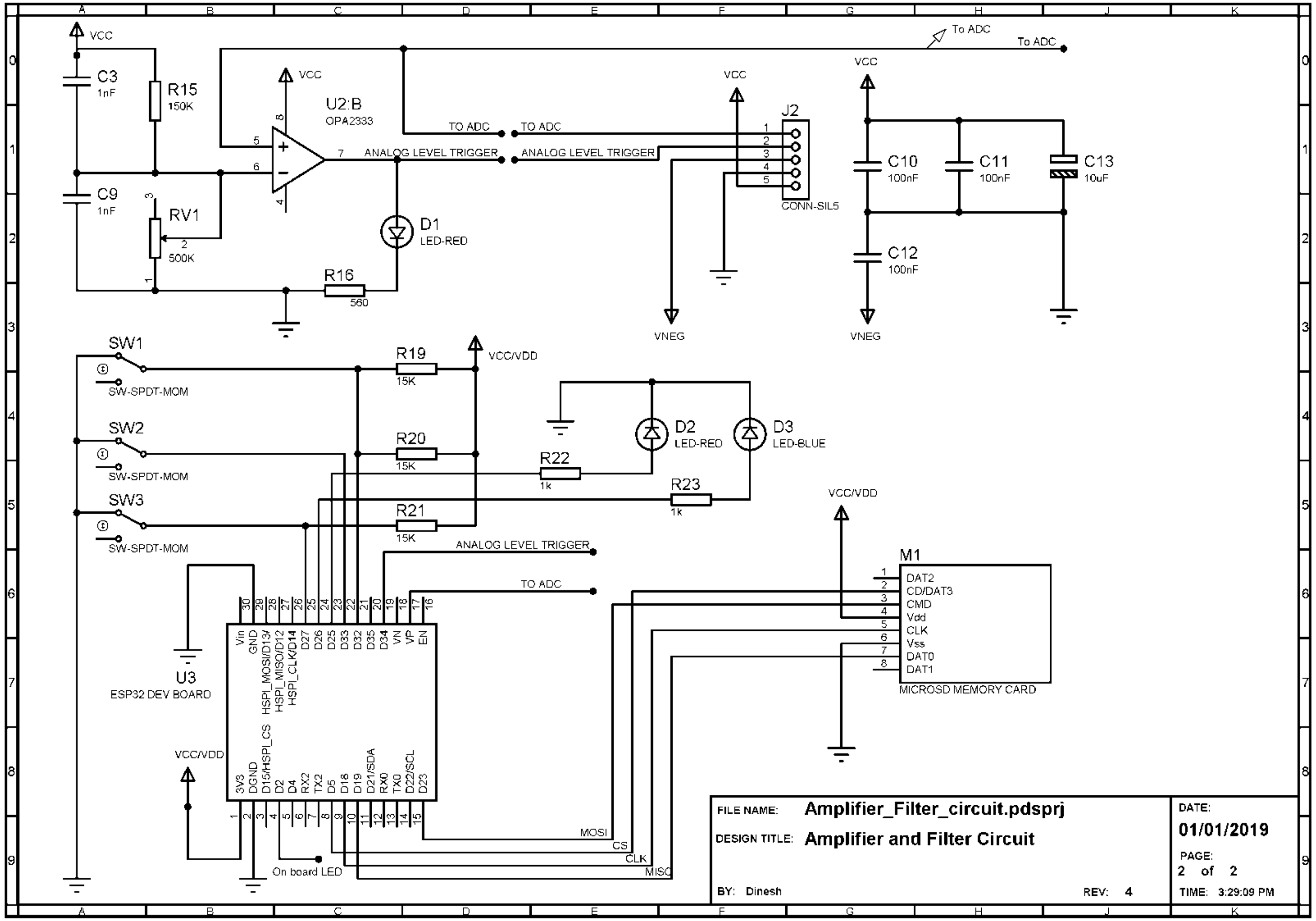


FIGURE 22: Circuit schematic 2/2

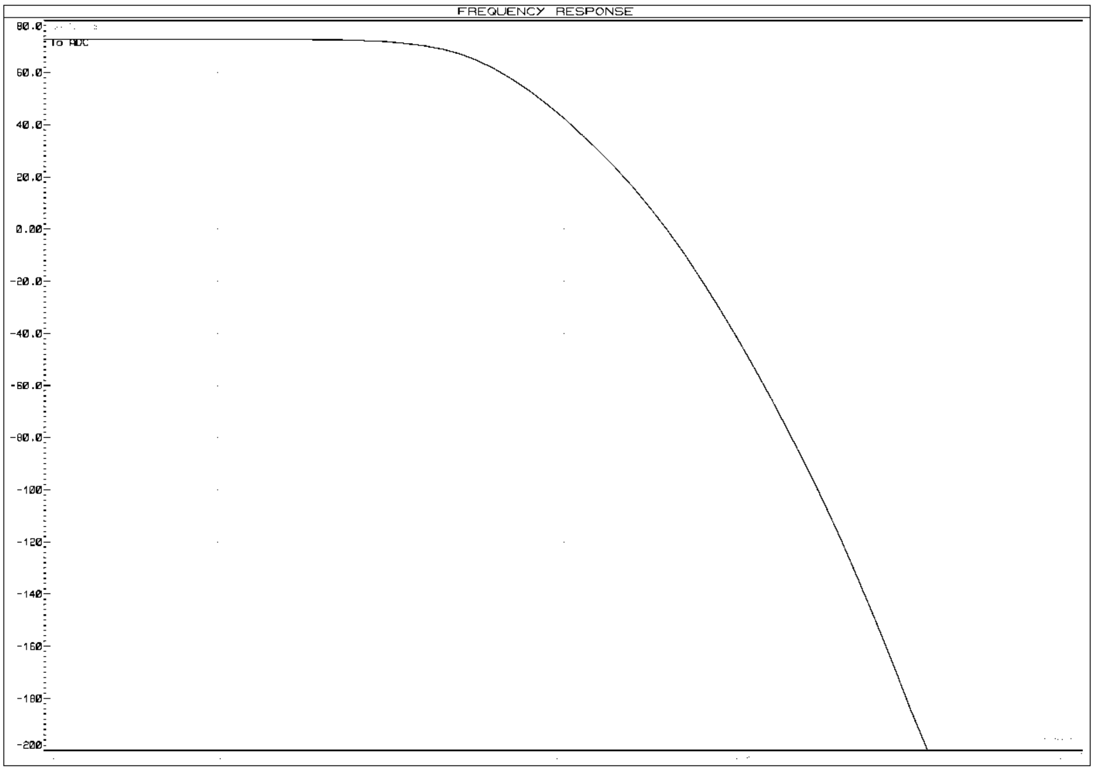


Figure 23: Frequency response of the amplifier-filter circuit

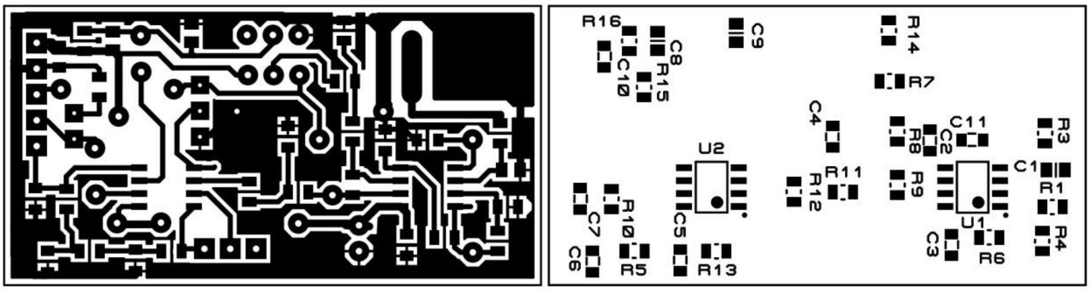


Figure 24: Bottom layer copper (Left) and Silk-screening (Right) of the PCB

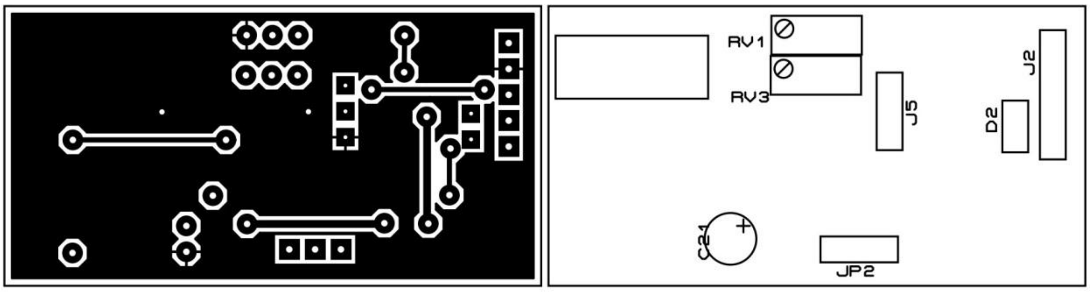


Figure 25: Top layer copper (Left) and Silk-screening (Right) of the PCB

Arduino code has been written to ESP32 such that it wake-up on trigger signal and start sampling analog signal in 1024 samples/sec rate. ADC bandwidth was selected to 12bit. Data is sampled for 30 seconds duration while storing ADC values in RAM. Hence 30720 number of 12bit ADC values were stored in 46080 bytes of RAM. At the end of the duration all data is written to a microSD memory card (2GB total size).

Appendix B: Extracted feature values

Table 9: Extracted feature values

Appendix C: Correlation matrix of features

Table 10: Correlation matrix of features

FEATURES	cutin-count	avg-ratio	dft-maxf	cog	dft-maxf /cog	trig-width	filt-diff	minMax	minMax /filt_diff	minMax /st-div	minMax /trig-width	st-div
cutin-count	1.0000	0.0053	-0.1772	-0.1637	-0.1673	0.1436	-0.2467	-0.3994	-0.6523	-0.7716	-0.4895	-0.2298
avg-ratio	0.0053	1.0000	-0.0184	0.5056	-0.0753	-0.3773	-0.1606	-0.2213	-0.1407	-0.0650	-0.0864	-0.1895
dft-maxf	-0.1772	-0.0184	1.0000	0.4578	0.9952	-0.1683	0.2488	0.3430	0.3492	0.2771	0.4315	0.3030
cog	-0.1637	0.5056	0.4578	1.0000	0.3693	-0.4468	0.1020	0.1500	0.1744	0.3118	0.3274	0.0702
dft-maxf /cog	-0.1673	-0.0753	0.9952	0.3693	1.0000	-0.1288	0.2470	0.3406	0.3458	0.2547	0.4142	0.3076
trig-width	0.1436	-0.3773	-0.1683	-0.4468	-0.1288	1.0000	0.4587	0.4421	0.1570	0.0726	0.1288	0.4761
filt-diff	-0.2467	-0.1606	0.2488	0.1020	0.2470	0.4587	1.0000	0.9501	0.3003	0.3352	0.8762	0.9908
minMax	-0.3994	-0.2213	0.3430	0.1500	0.3406	0.4421	0.9501	1.0000	0.5597	0.5725	0.9338	0.9586
minMax /filt_diff	-0.6523	-0.1407	0.3492	0.1744	0.3458	0.1570	0.3003	0.5597	1.0000	0.9351	0.5653	0.3535
minMax /st-div	-0.7716	-0.0650	0.2771	0.3118	0.2547	0.0726	0.3352	0.5725	0.9351	1.0000	0.6051	0.3432
minMax /trig-width	-0.4895	-0.0864	0.4315	0.3274	0.4142	0.1288	0.8762	0.9338	0.5653	0.6051	1.0000	0.8807
st-div	-0.2298	-0.1895	0.3030	0.0702	0.3076	0.4761	0.9908	0.9586	0.3535	0.3432	0.8807	1.0000

Appendix D: Readings transformed on to orthogonal space by SVD

Table 11: 41 Feature vectors of elephant footfalls transformed to orthogonal space by SVD

Signal Index	PCA 1	PCA 2	PCA 3	PCA 4
1	-32.9568	-0.6622	1.3093	-0.0933
2	-34.3853	-0.3925	1.6660	-0.0574
3	-35.3067	7.9837	3.5700	-0.0839
4	-37.7455	26.1028	1.2648	0.0666
5	-35.5946	0.9368	0.8810	-0.0325
6	-35.9166	3.6673	-0.1989	0.1207
7	-33.7206	-6.4991	0.7645	0.8552
8	-34.0775	0.3065	2.5854	-0.1786
9	-32.7496	-1.3021	2.3202	-0.1151
10	-34.1986	-1.1266	3.7233	0.1259
11	-33.7140	-0.9136	0.4827	-0.1143
12	-31.9174	-2.0582	3.1757	-0.0995
13	-34.5907	-2.4118	-4.3889	-0.1060
14	-33.8946	-1.6870	-1.5019	-0.1394
15	-33.4592	-1.5141	-0.5886	-0.1392
16	-33.6979	-1.8640	-0.5096	-0.0953
17	-33.5089	-1.8300	0.4753	0.0117
18	-34.3927	-1.1998	0.6713	-0.1579
19	-33.6914	-1.2241	2.5629	0.1301
20	-34.9709	-1.6893	-0.1863	0.1265
21	-34.1426	-1.6739	-1.4371	-0.0314
22	-34.9459	3.9564	1.6046	0.1303
23	-34.2933	-0.7751	2.6950	-0.0246
24	-34.8696	-1.8148	-2.2744	0.1311
25	-34.2763	-0.9346	-2.4717	-0.0038
26	-34.7898	-0.2625	1.7647	-0.0225
27	-35.2592	-0.7384	-3.2623	-0.0951
28	-34.2392	-1.3427	-0.3951	-0.1227
29	-35.8939	-1.6506	-5.1217	-0.0274
30	-34.0421	-1.1181	2.6482	0.1072
31	-34.1485	-1.3179	-0.4197	-0.1170
32	-34.5733	-1.1531	-2.3855	-0.0857
33	-34.5187	-1.8642	-3.4000	-0.1235
34	-34.7689	-0.2065	-0.3130	-0.1427
35	-35.9601	-0.0693	-4.1474	-0.1025
36	-33.4274	-1.0032	2.4798	-0.2139
37	-34.4934	-1.1689	-0.3349	0.1047
38	-32.8316	-10.5098	5.8867	-0.0010
39	-34.9004	0.1668	0.7397	0.0627
40	-36.6273	5.1484	-2.1519	0.3223
41	-36.3818	1.3501	-6.1710	0.1812

Appendix E: MATLAB Code

I)

```
function info=GetFeat(sig,maxNoise,Fs,cir_buf_size)

%Fs=1024;
Ts=1/Fs;
detect=0;           %Detection of trigger level exceeds
lenx=length(sig);
%cir_buf_size=512; %>100
buf_indx=cir_buf_size; %or equal to i of previous loop


buff=zeros(1,cir_buf_size);
peak_array=zeros(1,cir_buf_size);
sig_total=0;
st_div=0;
minMax=0;
trig_width=0;

for i=(1:cir_buf_size)%Fill buffer onece to calculate signal mean
    sig_total=sig(i)+sig_total;
    buff(i)=sig(i);
end
sig_mean=sig_total/cir_buf_size;


i=cir_buf_size/8;  %Start from 512/8

while(i<(lenx-2))
        %Search for trigger
        trig_end=0;
        while ( (i<=lenx) && ((trig_end==0) || (i<trig_end)) )
            if ( (sig(i)>(sig_mean+maxNoise)) && (sig(i-
1)>(sig_mean+maxNoise)) && (sig(i-2)>(sig_mean+maxNoise)) &&
(trig_end==0) && ((lenx-1-i)>=(cir_buf_size-(cir_buf_size/8))) )
             trig_start=i;
             %Raise flag of possible foot signal
             trig_end=i+(cir_buf_size-(cir_buf_size/8));   %Keep
last 64 readings

            end

            buf_indx=mod((i-1),cir_buf_size)+1;

            sig_total=sig_total-buff(buf_indx)+sig(i);
            sig_mean=sig_total/cir_buf_size;
            buff(buf_indx)=sig(i);

            i=i+1;
        end

        if (trig_end ~=0)
```

```
            detect=detect+1;
            x=zeros(1,cir_buf_size);        %Straight array for DFT
calculation
            buf_indx=mod((buf_indx+1-1),cir_buf_size)+1;    %
buf_indx+1 position will be x(1)
            for j=1:cir_buf_size
                x(j)=buff(buf_indx);
                buf_indx=mod((buf_indx+1-1),cir_buf_size)+1;
%Advance buf_indx by 1
            end

            xn=(1:cir_buf_size);%Index for array of signal to find
DFT
            %[y]=My_filter(x,FN,M)
            y=STD_filter(x);
                figure(2);
                plot(xn,x);
                hold on
                grid on
                plot(xn,y);
                hold off
            xlabel 'Sample ',ylabel 'Original & filtered signal'
            title('Current triggered section of signal');

            st_div(detect)=std(y);
            minMax(detect)=max(y)-min(y);
            filt_x=8:cir_buf_size;
            filt_diff(detect)=(sqrt(sum((x(filt_x)-y(filt_x-
7)).^2)))/(cir_buf_size-7);

            cutinLevel=minMax(detect)/6+sig_mean;
            cutin_count(detect)=0;
            for cut_indx=2:cir_buf_size
                if (y(cut_indx)>=cutinLevel && y(cut_indx-
1)<=cutinLevel)
                    cutin_count(detect)=cutin_count(detect)+1;
                end
            end


            y_front_trig=0;
            y_mean=mean(y);
            width_trace=1;
            while ((width_trace<lenx) && (y_front_trig==0))
                if (y(width_trace)>(y_mean+maxNoise))
                    y_front_trig=width_trace;
                end
                width_trace=width_trace+1;
            end

            y_back_trig=0;
            width_trace=length(y);
            while ((width_trace>0) && (y_back_trig==0))
                if (y(width_trace)>(y_mean+maxNoise))
                    y_back_trig=width_trace;
                end
                width_trace=width_trace-1;
            end

            trig_width(detect)=(y_back_trig-y_front_trig);

            %function [fk,Xk]=MyDFT(xn,x,Fs)
            [fk,Xk]=MyDFT(xn,y,Fs);

            fk_plot=fk(2:64);
            Xk_plot=floor(20*log(abs(Xk(2:64))));%Neglect DC
                figure(3);
                plot(fk_plot, Xk_plot );
                grid on
                hold off
            xlabel 'Frequency',ylabel 'DFT Coefficients /dB'
            title ('DFT Coefficients for peak detection');

            avg1(detect)=sum(Xk_plot(7:10))/4;
            avg2(detect)=sum(Xk_plot(14:17))/4;
            avg3(detect)=sum(Xk_plot(23:26))/4;

            dft_val(detect,:)=Xk_plot;

            max_dft=Xk_plot(1);
            maxf=1*2;
            for find_maxf=1:63
                if (max_dft < Xk_plot(find_maxf) )
                    max_dft=Xk_plot(find_maxf);
                    maxf=find_maxf*2;
                end
            end
            dft_maxf(detect)= maxf;

            cog_total=sum(Xk_plot(6:24));
            cog_accumu=0;
            cog_index=7;
            while (cog_accumu<(cog_total/2))
                cog_accumu=cog_accumu+Xk_plot(cog_index);
                cog_index=cog_index+1;
            end
                cog(detect)=fk_plot(cog_index)-((cog_accumu-
(cog_total/2))/Xk_plot(cog_index))*(fk_plot(cog_index)-
fk_plot(cog_index-1));

            current_peak=1;
            max_interest_freq=63;
            while ((current_peak~=0) && ((max_interest_freq-
current_peak)>1))
                current_peak=peak_find(
Xk_plot,current_peak,max_interest_freq);% Get first peak
                if (current_peak~=0)
                    peak_array(detect,current_peak)=1;
                end
            end
```

```
        end

        %detect;
        minMax_filtdiff(detect)=minMax(detect)/filt_diff(detect);
        minMax_st_div(detect)=minMax(detect)/st_div(detect);
        minMax_trig_width(detect)=minMax(detect)/trig_width(detect);
        dft_maxf_cog(detect)=dft_maxf(detect)/cog(detect);
        avg_ratio(detect)=(avg2(detect)-avg1(detect))/(avg2(detect)-
avg3(detect));

end

%info=[cutin_count' avg_ratio' dft_maxf' cog' dft_maxf_cog'
trig_width' filt_diff' minMax' minMax_filtdiff' minMax_st_div'
minMax_trig_width' st_div'];
%info=[cutin_count' avg_ratio' cog' dft_maxf_cog'  trig_width'
minMax_st_div' st_div'];
info=[cutin_count' avg_ratio' cog' dft_maxf_cog'];

%may include dft_val or peak_array also

end
```

II)

```
function [sig_index]=DetectEle(inp,maxNoise,Fs,cir_buf_size,foot)
% inp:Input signal
% maxNoise :Trigger level to start processing
% Fs : Sampling frequency
% cir_buf_size : Length of subsequence to be processed in a time.
% Equal to single footfall duration in number of data points.


Ts=1/Fs;
trig_events=0;     %Detection of trigger level exceeds
foot_count=0;      %Count of detected footfalls
len_inp=length(inp);
buff=zeros(1,cir_buf_size);
sig_index=0;


sig_total=0;
c1=0;
c2=0;


%% Plot whole input signal
figure(2);
hold off
plot(inp,'b');
xlabel 'Sample number',ylabel 'Input signal-ADC value'
title('Subsequences identified as Elephant foot falls');

for i=(1:cir_buf_size)% Fill buffer onece to calculate signal mean for first time
    sig_total=inp(i)+sig_total;
```

```
   buff(i)=inp(i);
end
sig_mean=sig_total/cir_buf_size;

buf_indx=cir_buf_size; %equal to i of previous loop
i=cir_buf_size/8;  %Start from cir_buf_size/8 location of inp signal

while(i<(len_inp-2))
      % Search for trigger
      trig_end=0;
      while ( (i<=len_inp) && ((trig_end==0) || (i<trig_end)) )
         if ( (inp(i)>(sig_mean+maxNoise)) && (inp(i-1)>(sig_mean+maxNoise)) && (inp(i-
2)>(sig_mean+maxNoise)) )
          % Three consecutive data points > (sig_mean+maxNoise)
             if (trig_end==0) && ((len_inp-1-i)>=(cir_buf_size-(cir_buf_size/8)))
             % not yet detected threshold exceed (ie. not yet triggered
             % trig_end=0) and have enough remaining data in inp

             % Set end loaction for selected subsequence to be processed, Keep last (cir_buf_size/8)
           readings
             trig_end=i+(cir_buf_size-(cir_buf_size/8));
             end
         end

         buf_indx=mod((i-1),cir_buf_size)+1;% Update circular buffer index

         sig_total=sig_total-buff(buf_indx)+inp(i);
         sig_mean=sig_total/cir_buf_size; %Update signal mean in ciruclar buffer
         buff(buf_indx)=inp(i);%Update buffer
         i=i+1;
      end

      if (trig_end ~=0)%Signal processing triggered, threshold exceed found
         trig_events=trig_events+1;
         x=zeros(1,cir_buf_size);        %Straight array x, for subsequence of inp (for DFT calculation)
         buf_indx=mod((buf_indx+1-1),cir_buf_size)+1;    % buf_indx+1 position will be x(1)
         for j=1:cir_buf_size
            x(j)=buff(buf_indx);
            buf_indx=mod((buf_indx+1-1),cir_buf_size)+1; %Advance buf_indx by 1
         end

         xn=(1:cir_buf_size);%Index for subsequence array
         y=STD_filter(x);%Digital Filter
         minMax_div=max(y)-min(y);
         cutinLevel=minMax_div/6+sig_mean;%Dynamic threshold level
         cutin_count=0;
         for cut_indx=2:cir_buf_size
            if (y(cut_indx)>=cutinLevel && y(cut_indx-1)<=cutinLevel)
               cutin_count=cutin_count+1;
            end
         end

         [fk,Xk]=MyDFT(xn,y,Fs);
         fk_plot=fk(2:64);
         Xk_plot=floor(20*log(abs(Xk(2:64))));%Neglect DC
         %Since Fs=1024 & length of data series is 512, DFT values Xf are in
         %f=2, 4, 6 ....., 512 (steps of 2)
         r1_avg=sum(Xk_plot(7:10))/4;
         r2_avg=sum(Xk_plot(14:17))/4;
         r3_avg=sum(Xk_plot(23:26))/4;

         if ((minMax_div>(maxNoise*4)) )
            c1=c1+1;

            if (cutin_count>4 &&  cutin_count<11)
               c2=c2+1;

                 if ((r2_avg>r3_avg) && (r3_avg>r1_avg)) %Not same as ((avg2(detect)-
avg1(detect))/(avg2(detect)-avg3(detect))) >1 in Getfeat
                  foot_count=foot_count+1;
                  sig_index(foot_count)=trig_end-cir_buf_size;
                  figure(2);
                  hold on
                  plot_range=(trig_end-512): (trig_end-1);
                  plot(plot_range,x, 'r');
                  grid on
                 end
            end
         end

      end


end
%% Outputs
trig_events
c1
c2
foot_count
detection_rate=foot_count/(len_inp*Ts)
avg_footfall_interval=1/detection_rate
end
```

III)

```
function [y]=STD_filter(x)
%x=Signal to be filtered; y=Filtered output
%Hamming window 15h order

lng_x=length(x);
%180Hz (-6dB)
filt_cof=[0.00359174577624323,0.00223234043583294,-
0.0110569789104463,-0.0331538775370290,-0.0115917331304444,
0.105146087106231,0.269962740921109,0.349739350677007,0.269962740921
109,0.105146087106231,-0.0115917331304444,-0.0331538775370290,-
0.0110569789104463,0.00223234043583294,0.00359174577624323];
lng_cof=length(filt_cof);
```

```
   y=zeros(1,lng_x);
   for i=1:lng_x

       temp=0;
       if (i>=lng_cof)
           for j=1:lng_cof
                   temp=temp+filt_cof(j)*x(i-j+1);
           end
       else
       temp=x(i);
       end

       y(i)=floor(temp); %Remove fraction and get only integer value
   end
end
```

IV)

```
function [fk,Xk]=MyDFT(xn,x,Fs)
%Not used FFT, xn=Index for samples in x; x=Signal samples;
%Fs=Sampling frequency for fk frequency calculation
%Xk=kth DFT coefficient
%fk=kth frequency in Hz

if (length(x) ~=length(xn))
    error('Length input signal x is not equal to n')
end

N=length(x);
n=[0:1:(N-1)];
k=[0:1:(N-1)];

WN=exp(-1j*2*pi/N);
nk=n'*k;
WNnk=WN.^nk;

%If x is column vector, transform it to row vector; to be compatible with
%belowmatrix multiplication
[~,colz]=size(x);
if colz==1
  x=x';
end

Xk=x*WNnk;
Ts=(1/Fs);  %Time domain sampling interval
fk=(1/(N*Ts)).*k;%Frequency span

end
```

V)

```
%load('4_Features_on_SVM.mat');
%rndinx=randperm(41);
%t40info_4_train30=t40info_4(rndinx(1:30),:,:,:);
%datasvm=[t40info_4_train30;Feat_cattles2dog;Feat_girlwalk_1;Feat_hu
manwalk_1;Feat_mbike_1;Feat_tractor2;Feat_Two3W_BigLorry;Feat_TwoMbi
ke_3W];
%theclass=ones(345,1); %set all are +Ve data points
%theclass(31:345)=-1; %-Ve data points
%t40info_4_test11=t40info_4(rndinx(31:41),:,:,:);

figure;
%Selecte features, column numbers in datasvm
f1=1;
f2=2;

data=[datasvm(:,[f1]),datasvm(:,[f2])];

scatter(data(1:30,f1),data(1:30,f2),'ro');%+1: Elephant foot falls
hold on;
grid on;
scatter(data(31:345,f1),data(31:345,f2),'go');%-1: Othere sources


% Train the classifier
SVMModel =
fitcsvm(data,theclass,'KernelFunction','rbf','ClassNames',[-1 1]);

% Predict scores over the grid
d = 0.1;
[x1Grid,x2Grid] = meshgrid(min(data(:,1)):d:max(data(:,1)),
min(data(:,2)):d:max(data(:,2)));
xGrid = [x1Grid(:),x2Grid(:)];
[~,scores] = predict(SVMModel,xGrid);

predict(SVMModel,[t40info_4_test11(:,f1), t40info_4_test11(:,f2)])

% Plot the data and the decision boundary
figure;
h(1:2) = gscatter(data(:,1),data(:,2),theclass,'rg','+*');
hold on
h(3) = plot(data(SVMModel.IsSupportVector,1),...
    data(SVMModel.IsSupportVector,2),'ko');
contour(x1Grid,x2Grid,reshape(scores(:,2),size(x1Grid)),[0 0],'k');
legend(h,{'-1','+1','Support Vectors'},'Location','Southeast');
grid on
hold off
```