ACOUSTIC SURFACE PERCEPTION THROUGH THE GROUND INTERACTION OF COMPLIANT LEGS OF A HEXAPOD ROBOT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MİNE CÜNEYİTOĞLU ÖZKUL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

JANUARY 2012

Approval of the thesis:

ACOUSTIC SURFACE PERCEPTION THROUGH THE GROUND INTERACTION OF COMPLIANT LEGS OF A HEXAPOD ROBOT

submitted by **MİNE CÜNEYİTOĞLU ÖZKUL** in partial fulfillment of the requirements for the degree of **Master of Science in Mechanical Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Süha Oral
Head of Department, **Mechanical Engineering**

Asst. Prof. Dr. Yiğit Yazıcıoğlu
Supervisor, **Mechanical Engineering Department**

Asst. Prof. Dr. Afşar Saranlı
Co-supervisor, **Electrical and Electronics Engineering Department**

Examining Committee Members:

Asst. Prof. Dr. A. Buğra Koku
Mechanical Engineering Dept., METU

Asst. Prof. Dr. Yiğit Yazıcıoğlu
Mechanical Engineering Dept., METU

Asst. Prof. Dr. Afşar Saranlı
Electrical and Electronics Engineering Dept., METU

Asst. Prof. Dr. Gökhan O. Özgen
Mechanical Engineering Dept., METU

Prof. Dr. Aydın Alatan
Electrical and Electronics Engineering Dept., METU

Date: **25.01.2012**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: MİNE CÜNEYİTOĞLU ÖZKUL

Signature :

ABSTRACT

ACOUSTIC SURFACE PERCEPTION THROUGH THE GROUND INTERACTION OF COMPLIANT LEGS OF A HEXAPOD ROBOT

Özkul, Mine Cüneyitoğlu
M.S., Department of Mechanical Engineering
Supervisor : Asst. Prof. Dr. Yiğit Yazıcıoğlu
Co-Supervisor : Asst. Prof. Dr. Afşar Saranlı

January 2012, 83 pages

A dynamically dexterous legged robot platform generates specific acoustic signals during the interaction with the ground and the environment. These acoustic signals are expected to contain rich information that is related to the interaction surface as a function of the position of the legs and the overall contact process mixed with the actuator sounds that initiate the movement. As the robot platform walks or runs in any environment, this convolved acoustic signal created can be processed and analyzed in real time operation and the interaction surface can be identified. Such a utilization of acoustic data can be possible for various indoor and outdoor surfaces and this can be useful in adjusting gait parameters that play an essential role in dynamic dexterity. In this work, surface type identification is achieved using several popular signal processing and pattern classification methods not on the robot platform but off-line. The performances of the selected features and algorithms are evaluated for the collected data sets and these outputs are compared with the expectations. Depending on the off-line training and experiment results, the applicability of the study to an embedded robot platform as a future application is found quite feasible and the surface type as an input to the robot sensing is expected to improve the mobility of the robot in both indoor and outdoor environment.

Keywords: Legged Robotics, Acoustic Sensing, Classification, Surface Type, Identification

ÖZ

ALTI BACAKLI BİR ROBOTUN ESNEK BACAKALARININ YÜZEY ETKİLEŞİMİNİ KULLANARAK AKUSTIK YÜZEY ALGILAMASI

Özkul, Mine Cüneyitoğlu
Yüksek Lisans, Makina Mühendisliği Bölümü
Tez Yöneticisi : Asst. Prof. Dr. Yiğit Yazıcıoğlu
Ortak Tez Yöneticisi : Asst. Prof. Dr. Afşar Saranlı

Ocak 2012, 83 sayfa

Dinamik çevikliğe sahip bacaklı bir robot platformu hareket halindeyken çevresiyle etkileşim halinde olduğu için akustik sinyaller yayar. Bu akustik sinyallerin içeriğinde, hareketi sağlayan eyleyicilerin seslerine karışmış olarak, bacakların pozisyonuna ve etkileşim yüzeyi ile olan etkileşim sürecinin tamamına dair bilgiler bulunması beklenmektedir. Bu karışık sinyali gerçek zamanlı olarak işlemenin ve sinaylin içerisinden etkileşim yüzeyinin türüne dair bilgiyi elde etmenin mümkün olduğu düşünülmektedir. Bu tür bir süreçten elde edilecek yüzey tipi verisi, bacaklı bir robot platformumun yürüyüş parametrelerini belirlemesine katkıda bulunarak çeşitli iç ve dış ortamlardaki çeviklik performansını arttıracağı için önemlidir. Bu çalışmada, robot üzerinde gerçek zamanlı işleme yerine veriler ayrı bir kaynakta toplanmıştır ve bu ayrı ortamda çeşitli sinyal işleme ve örüntü tanımlama teknikleri uygulanarak incelenmiştir. Bu inceleme sonucu çeşitli işlem parametrelerinin ve örüntü tanıma algoritmalarının karar sürecine olan katkıları ve performansları analiz edilerek deneyler öncesi beklentiler ile kıyaslanmıştır. Sonuçlar değerlendirildiğinde, bu tür bir analiz ile önemli miktarda başarılı sonuçlar elde edildiği görülmüştür ve bu tür bir uygulama robot platformu üzerinde gömülü olarak yapıldığı taktirde robotun yüzey tipini tayin etmesini sağlayabilir. Bu sayede robotun içinde bulunduğu ortamı algılayarak iç ve dış ortamda hareket kabiliyetinin artması söz

konusu olabilecektir.

Anahtar Kelimeler: Bacaklı Robotlar, Akustik Algılama, Sınıflandırma, Yüzey Tipi, Tanımlama

Dedicated to my beloved one, Alp Eren Özkul...

ACKNOWLEDGMENTS

I would like to thank my thesis supervisor Dr. Yiğit Yazıcıoğlu and co-supervisor Dr. Afşar Saranlı for their efforts throughout the last three years of my graduate study. With their open minded perspective and innovative minds, I had a very fruitful graduate study period which I believe is of utmost importance to an engineering student. With their guidance, I could utilize creative solutions to the problems that I faced and I could still stay focused on the direction that I want to go.

SensoRHex project is a great opportunity for graduate level studies. I would like to thank TÜBITAK for funding the programs 106E089 and 110E120 as 1001 projects and granting many graduate students such an opportunity. I would like to express my deepest appreciations to Dr. Afşar Saranlı, Dr. Uluç Saranlı, Dr. Yiğit Yazıcıoğlu and Dr. Kemal Leblebicioğlu as members of the SensoRHex project and their support and countless invaluable advices. I would like to thank METU Department of Mechanical Engineering, METU Department of Electrical Engineering, Bilkent University Department of Computer Engineering and Bilkent University Department of Electrical Engineering for constituting such an interdisciplinary work environment. I would like to thank all of the students in RoLab and Bilkent Dexterous Robotics and Locomotion (BDRL) for developing the SensoRHex platform and sharing their academic and practical experiences anytime.

Without the help of the graduate students in Laboratory of Robotics & Autonomous Systems (RoLab) this thesis would not exist at all. I would like to express my sincere thanks to Emre Akgül, Emre Ege, Gökhan Gültekin, Mert Ankaralı and Ferit Uzer for their support during the experiments, through the overall study and for anything I cannot name of but most important of all, their invaluable, warm friendship.

I would like to thank my beloved husband, Alp Eren Özkul for his encouragement, patience and endless support. He never spared any of these essential elements in a graduate study. I would like to thank my parents Emine and Mehmet Gani Cüneyitoğlu for their support and especially thank my brother Cüneyt Cüneyitoğlu for introducing me to scientific thinking

when I was a little kid and then to this brilliant occupation called ”engineering” and his cute family for their warmness and support. I would like to thank Alp’s parents for supporting me through all of this process. I would like to thank my cousin Selin Demir and my friend Yeliz Balcı for our fun talks about my graduate studies. Finally, I would like to express a deep gratitude to all of my close friends but especially to Buse Gönül, for the two and a half years that we shared as flat mates before my marriage. She always listened to my endless murmurs, endured my ”engineering talk” during my studies and we managed to turn any kind of trouble to fun all the time. Without her friendship, warmness and sincerity, I possibly could not find the will to carry on.

TABLE OF CONTENTS

ABSTRACT iv

ÖZ vi

ACKNOWLEDGMENTS ix

TABLE OF CONTENTS xi

LIST OF TABLES xiv

LIST OF FIGURES xvi

LIST OF SYMBOLS xix

CHAPTERS

1 INTRODUCTION 1

1.1 ROBOTIC ACOUSTIC PERCEPTION 2

1.1.1 ANALYSIS OF NON-SPEECH SIGNALS 3

1.1.2 EARLIER STUDIES ON INTERACTION AND IMPACT SOUNDS 3

1.1.3 ROBOTIC SURFACE AND MATERIAL IDENTIFICATION 4

1.2 EXPERIMENTS RELATED TO THE SUBJECT 6

1.3 MOTIVATION AND OBJECTIVES 6

2 EXPERIMENTAL SETUP AND DATA PREPARATION 8

2.1 ROBOT PLATFORM - SENSORHEX 8

2.1.1 MECHANICS OF THE ROBOT 8

2.1.2 RECORDING EQUIPMENT 10

2.1.3 MICROPHONE 11

2.2 EXPERIMENT SURFACES 14

2.3 EXPERIMENTAL SCHEME 15

2.4 ANALYZING THE SIGNAL 19

2.4.1 STATISTICAL PROPERTIES OF THE DATA 20
2.4.2 SPECTRAL EXAMINATION OF THE DATA 20
2.4.3 SPECTROGRAMS OF THE DATA 24
2.5 PATTERN SELECTION FOR CLASSIFICATION 26
2.5.1 SEGMENTATION APPROACH 26
2.5.2 HOLISTIC APPROACH 28
2.5.3 MOTOR NOISE EXTRACTION APPROACH 28
3 FEATURE EXTRACTION 30
3.1 APPLICATION SPECIFIC SELECTION 30
3.2 THE PROBLEM OF HIGH DIMENSIONALITY 31
3.3 SELECTION OF RELEVANT FEATURES 33
3.4 SIGNAL PROCESSING 34
3.4.1 FREQUENCY DOMAIN 34
3.4.2 TRAPEZOIDAL ENERGY BANDS 35
3.4.3 ZERO CROSSING RATE 37
3.4.4 DERIVATIVE OF THE SPECTRUM 38
4 CLASSIFIERS USED IN THIS WORK 39
4.1 ALGORITHMS IMPLEMENTED 39
4.1.1 VECTOR QUANTIZATION 40
4.1.1.1 CODEBOOK VECTORS 40
4.1.1.2 VQ AS A SUPERVISED LEARNING METHOD 42
4.1.2 NAIVE BAYES ALGORITHM 42
4.1.3 DECISION TREE ALGORITHMS IN GENERAL 43
4.1.3.1 MULTIVARIATE TREES 43
4.1.3.2 THE LOGITBOOST ALGORITHM 43
4.1.3.3 FUNCTIONAL TREES 45
4.1.4 SIMPLE LOGISTIC 46
4.2 CROSS VALIDATION 47
4.3 THE EXECUTION PROCEDURE 47
4.3.1 THE PROGRAM FLOW 47

4.3.2 THE EXPERIMENT PLANNING 49
4.4 ANALYSIS OF CLASSIFICATION PERFORMANCE 50
4.4.1 CONFUSION MATRICES 50
4.4.2 OVERALL SUCCESS RATE 51
4.4.3 SUCCESS RATE PER CLASS 51
4.5 CONSIDERATION OF COMPUTATION TIME 52
5 EXPERIMENTAL RESULTS AND DISCUSSION 53
5.1 EARLIER VECTOR QUANTIZER WORK ON PRELIMINARY DATA SET 53
5.2 VQ IMPLEMENTATION TO FOUR CLASSES OF MAIN DATA SET 55
5.3 WEKA IMPLEMENTATION ON THE MAIN DATA SET 59
5.3.1 TIME COMPLEXITY VS PERFORMANCE 60
5.3.2 NAIVE BAYES VS FUNCTIONAL TREE ALGORITHM 64
5.4 ANALYZING THE EFFECT OF THE SELECTED FEATURES . . 65
5.4.1 THE EFFECT OF NUMBER OF AVERAGE SPECTRUM FEATURES 65
5.4.2 THE EFFECT OF ZERO CROSSING RATE FEATURE . 67
5.4.3 THE EFFECT OF DERIVATIVE SPECTRUM FEATURE 67
5.5 ANALYZING THE EFFECT OF TOKEN SIZE 67
5.6 ANALYSIS ON CLASSES 70
5.7 THE EFFECTS OF SPECTRAL SUBTRACTION 71
6 CONCLUSION 76
6.1 APPLICABILITY OF THE WORK 77
6.2 FUTURE DIRECTIONS 79
REFERENCES 80
APPENDICES
A SONY ECM-DS70P MICROPHONE SPECIFICATIONS 83

LIST OF TABLES

TABLES

Table 2.1 Properties of the Recordings 11

Table 2.2 Specifications of SONY ECM-DS70P Stereo Microphone - See Appendix - A for details 12

Table 2.3 The Experiment Surfaces for the Preliminary Analyses 15

Table 2.4 The Experiment Surfaces for the Main Set of Experiments 15

Table 2.5 The Recording Properties for the Preliminary Analyses 16

Table 2.6 The Recording Properties for the Main Set of Experiments 16

Table 2.7 Statistical Properties of the Data - The Preliminary Experiment Set 20

Table 2.8 The Spectrogram Properties of the Images 24

Table 2.9 Signal Lengths and Token Sizes for Each Speed - The Main Experiment Set - Holistic Approach 28

Table 3.1 Parameters in Frequnecy Domain Transfrom with FFT 35

Table 4.1 Parameters of FT Algorithm in WEKA 45

Table 4.2 Experiment Sets Table - In both of these experiment sets, Hanning window is preferred 49

Table 4.3 Experiment Numbers Table- In all of these experiments, the main data set for recordings is used. The classes for $C = 6$ cases are (C, L, R, H, A, G) and in other cases, the classes added or removed are denoted with + and - signs. 49

Table 4.4 A Sample Confusion Matrix for $E_n = 3$, $B = 8$ and $P_c = 3$ 50

Table 4.5 A Sample Confusion Matrix for Experiment, $E_n = 8$, $B = 8$ and $P = 1$ . . . 51

Table 5.1 Some outstanding results of varying number of average spectrum features - B, for fixed $P_c = 2$ 54

Table 5.2 Some outstanding results of varying number of average spectrum features - B, for fixed $P_c = 3$ 54

Table 5.3 A Sample Confusion Matrix for Experiment $E_n = 13$, $B = 10$ 71

LIST OF FIGURES

FIGURES

Figure 2.1 SensoRHex Robot Platform 9

Figure 2.2 SensoRHex is a robot platform that can walk on irregular terrain which is a useful property especially for outdoors 9

Figure 2.3 Alternating tripod gait 10

Figure 2.4 SONY ECM-DS70P Microphone used in experiments - Adapted From [27] 12

Figure 2.5 The current microphone placement within the body of the robot is illustrated. However, it should be considered that the microphone is actually under the robot body on its other side, closer to the ground 13

Figure 2.6 The Indoor Experiment Surfaces - C,L,R,H,S respectively 13

Figure 2.7 The Outdoor Experiment Surfaces - A,O,G respectively 13

Figure 2.8 C surface is actually on L surface 14

Figure 2.9 The corresponding physical speed for the values of parameter P of SensoRHEX platform is measured in [28]. 17

Figure 2.10 The stereo record from the Carpet Surface as .wav file loaded in Audacity. 18

Figure 2.11 Comparison of approximately 1 s records of C and L. In here, the signals start with motor sound peak and end with ground leg interaction peak and the diminishing wave of the surface impact. 18

Figure 2.12 Hanning Window - D=1024 Samples 22

Figure 2.13 Ensemble Average Power Spectrum Estimate of the Record on Carpet, Linoleum and Stone. The ensemble average plots may not be very informative since the signal is non-stationary. 23

Figure 2.14 Spectrogram of Two Interactions with a Carpet (C) Surface. 25

Figure 2.15 Spectrogram of Two Interactions with a Linoleum (L) Surface. 25

Figure 2.16 Spectrogram of Two Interactions with a Stone (S) Surface. 26

Figure 2.17 Ensemble Average Power Spectrum Estimate of the 20 Segmented Motor Sounds 27

Figure 3.1 A two class problem. The projection to neither of the feature axes yields a successful classification. 32

Figure 3.2 Linear scale trapezoidal energy filters that average spectral energy on predetermined number of overlapping frequency bands. 36

Figure 3.3 The final feature vector when all of the mentioned features are included . . 37

Figure 4.1 A cluster of data expressed with three features and three codebook vectors are calculated and shown in the figure as C1, C2 and C3 in three dimensions. . . . 41

Figure 4.2 The k-means clustering algorithm - Adapted from [19] 41

Figure 4.3 The univariate tree generated by Simple CART algorithm for P=1, $E_b = 8$. This simpler type of classification does not yield satisfactory results for out problem. 44

Figure 4.4 Logitboost Algorithm applied to a J class problem - Adapted from [37] . . 45

Figure 4.5 Pseudo Code of the GrowTree Function of the Functional Tree Algorithm for inputs $DataSet, Constructor$- Adapted from [38] 46

Figure 4.6 The flow of the overall procedure 48

Figure 5.1 The performance of VQ for different speeds with features r_{zc} and a_{sds} included 56

Figure 5.2 The performance of VQ for different speeds without features r_{zc} and a_{sds} . 57

Figure 5.3 Observing the performance change with different values of P_c for $P = 5$ . . 58

Figure 5.4 Seeking the best performance with tuning the values of P_c for $P = 8$ 59

Figure 5.5 The performances of the applied algorithms in WEKA 61

Figure 5.6 The Performances of Classification for Different Speeds, $B = 10$, $E_n = 8$ (Time axis is in log scale) 62

Figure 5.7 The Performances of Naive Bayes and Functional Tree Classifiers for $P = 1, P = 5, P = 8$ ($E_n = 8$) 63

Figure 5.8 The Effect of Features r_{zc} and ΔE_b on Classification Performance (Classifier FT) 66

Figure 5.9 The Performances of Naive Bayes and Functional Tree Classifiers for a shorter token size, for $P = 1, P = 5, P = 8$ ($E_n = 9$) 68

Figure 5.10 The Performance Comparison for Different Token Sizes M, for Average of Performances of $P = 1$, $P = 5$, $P = 8$ 69

Figure 5.11 The Performance Change With 7 Classes for Classifiers NB and FT ($E_n = 12, E_n = 8$ and $E_n = 13$) 72

Figure 5.12 The effects of spectral subtraction for $P = 1$ 73

Figure 5.13 The effects of spectral subtraction for $P = 5$ 74

Figure 5.14 The effects of spectral subtraction for $P = 8$. The most successful results of this operation is observed for this speed. 75

Figure A.1 SONY ECM-DS70P Microphone Specifications 84

LIST OF SYMBOLS

A – Small Concrete Tile Surface

A_p – Confusion Matrix for Speed P

B_e – Effective Bandwidth

C – Carpet Surface

E_b – Energy sum of the bth Base Feature

E_n – Experiment Number

E_s – Experiment Set

G – Grass Surface

H – Hardwood Surface

L – Linoleum Surface

L_z – Number of Zeros to Add

O – Concrete Tile Surface

P – Locomotion Speed

$Q[k]$ – Power Spectrum Estimate of the Motor Noise

R – Marble Surface

S – Stone Surface

$S[k]$ – Power Spectrum Estimate of the Signal

S_{A_p} – Success Rate of Confusion Matrix A_p

T – Record Length

$X[k]$ – Fourier Transform of the Signal

Y – Posterior Probability

$Z[k]$ – Power Spectrum Estimate of the Motor Noise Subtracted Signal

Δ – Time Interval

Φ – Surface Model

α – Motor Spectrum Coefficient

β – Parameter Vector

γ – Power in Spectral Subtraction

λ – Frequency Variable

μ – Codebook Vector

σ_m – Standard Deviation of the mth token

a_{sds} – Sum of Delta Spectrum for a Token

$b = 1, 2, \ldots, B$ – Number of Base Features

b_{ij} – The Element of a Confusion Matrix at Index i,j

$d = 1, 2, \ldots, D$ – Index of the shifted time signal

$f = 1, 2, \ldots, F$ – Index of Feature Matrix

f_{mj} – Regression Function

h_{rj} – Weight of jth class

$j = 1, 2, \ldots, J$ – Number of Classes

k – k^{th} FFT vector component

$m = 1, 2, \ldots, M$ – Number of Tokens

m_m – Mean value of the mth token

$n = 1, 2, \ldots, N$ – Sample Length Per Token

$p = 1, 2, \ldots, P_c$ – The Codebook Vector index

$q = 1, 2, \ldots, Q_i$ – Iteration Index

$r = 1, 2, \ldots, R_i$ – Indicator Variable

$r = 1, 2, \ldots, R_t$ – Sample Points in Time Domain

r_{zc} – Zero Crossing Rate of a Token

v – Input Feature Vector

v_n – New Attribute Set of Input Feature Vector

$w(n)$ – Hanning Window

$w_r(n)$ – Rectangular Window

y – Likelihood

y^*_{ij} – Observed Class Membership Probability

z – Working Response for LogitBoost Algorithm

CHAPTER 1

INTRODUCTION

Sensing and perception of the environment is an indispensable ability for all living creatures. Acoustic sensing of the environment is essential for survival since it has a great role in getting alert from the surrounding dangers even if the threat is out of line of sight. Even the fish, the simplest animal, can hear up to several kHz and make use of this ability as a tool for survival [1]. When this consequential ability is examined more in detail, it will be realized that a huge amount of data related to the composition of the environment comes from the sensory organs related to acoustic perception. Another vital outcome of acoustic perception for more complex living organisms is communication. One of the most well-known example is maybe the under water communication of whales. Some species of whales can hear and decompose from very low frequency sounds like 10 Hz, up to several hundred kHz and moreover, they can actively communicate to other whales even if there is kilometers of distance between them [2].

Putting the sea world and world of predators and preys in the jungle aside, hearing and acoustic perception has been an important part of human history as well, with music and oral tradition. Undoubtedly, in present day's modern life, speech plays the utmost important role in communication. According to survey done with British people, a female person speaks 8805 words per day and a male person speaks 6073 words on average [3]. In addition to communication, people get emotional stimulus via music which are all gifts of the psychoacoustic ability.

Moving from the realm of living creatures to the world of robotics, biomimetics, a relatively new and an interdisciplinary area that studies on the designs in the nature and search the applicability of those designs in engineering problems [4] is becoming more common. In robotics

world, where biomimetic approaches are appreciated due to their functionality in solving various challenging problems, sensing can be considered as an essential part of robotics research. Among the literature survey done through all of this thesis process, there are not many examples of biomimetic sensing found compared to robotic applications that make use of advanced sensing. In all of the industries such as automotive, defense, textile, shortly, in any mass production line, such high technology sensing applications are plenty since industrial robotics cannot be thought of without sensors. Robotics world benefit a lot from advanced sensing that make use of some sophisticated signal processing operations and pattern classification algorithms.

Among sensing types like visual sensing, acoustic sensing with sonars, infrared or other types of proximity sensing, tactile sensing, inertial sensing and so on, acoustic sensing within the audible range has not been quite popular except for the cases with speech processing. Speech processing is undoubtedly a very important element in human robot interaction area. However, the examples given the previous paragraphs are outstanding examples of acoustic sensing in nature. With making use of a biomimetic approach, robots can sense and adapt the environment like those living creatures that make use of their acoustic perception ability of non-speech signals. In the following section, a compilation of various robotic acoustic perception studies that are extracted from the literature are presented.

1.1 ROBOTIC ACOUSTIC PERCEPTION

After a broad literature survey, it is observed that the number and the success of studies on speech perception is enormous [5]. Among acoustic sensing and perception within audible range, sound source localization and classification is an important branch of study [6]. Among the field of environment identification via acoustic perception, current studies are mostly done in the laboratory settings [7], [8] and are concentrated on human generated sounds.

In the fields other than robotics, there are numerous applications of acoustic sensing like medical diagnosis in biomedical applications [9] or in machine health monitoring in industry [10]. In the future, works in robotic acoustic perception can in fact make use of the methods used in these fields as well. Signal conditioning and processing is another essential part of sensing problems, that in the end help, the robot extract the features related to the ambient.

Finally, sophisticated pattern classification algorithms come in handy while making sense of the environment data collected and extracted.

1.1.1 ANALYSIS OF NON-SPEECH SIGNALS

From literature to a versatile number of applications in industry, in music, in medicine and so on, the acoustic data is often examined with some very common features like frequency components, pitch, intensity, that are of time domain and frequency domain. Power spectrum estimate of a sound signal that is to be explained in the following chapters is a powerful method to determine the frequency composition of it. One of the most common tools in acoustic signal processing is the discrete Fourier transform (DFT) [11], [5]. With transformation to frequency domain with applying the so-called short time Fourier transform, the power spectrum estimate of a signal for any time interval can be estimated and frequency content of the signal can be revealed effectively. In addition to its use in speech processing, power spectrum can be used in analyzing non - speech signals effectively as well.

There are transforms other than Fourier Transform that work well on acoustic signals like Wavelet transform [12] or advanced filtering methods [11] that a signal can be represented or further analyses on frequency spectrum called cepstral analysis. However, in the scope of this study, only zero crossing rate analysis and Fourier transform is applied with its most common form Fast Fourier Transform algorithm and its performance is found satisfactory and no more other techniques are applied.

1.1.2 EARLIER STUDIES ON INTERACTION AND IMPACT SOUNDS

In a PhD thesis presented in the field of Cognitive Science [13], acoustic models and theories that lie behind daily sounds are examined. In this study, contact sounds of objects are studied in particular and these sounds are classified into three basic categories like material type, interaction type and configuration. Since that is a thesis on cognitive science, later on, the perceptual and psychological effects of these sounds on humans have been examined. However, for analyzing these interaction sounds, an experimental session is conducted. with nineteen undergraduate students that volunteered. These students are expected to guess the length and the material of wooden and metal struck bars, only listening the interaction sounds of them

with several surfaces, with eyes blindfolded. The guesses have been especially successful for some definite interaction sounds of wooden blocks. At the same time, these interaction sound signals are recored and Time x Frequency x Amplitude plots are formed. The statements in the results section cannot be said to be not quite plenary in terms of acoustics, since this work is in a completely different field than the field of engineering. However, this work of Gaver has been an inspiring study to many forthcoming acoustic experiments and this study is further referenced in many other academic papers and theses.

As the digital computers advanced, more signal processing capabilities have been available on acoustic signals. In [14] and [15], detailed studies on impact sounds with several surfaces of concrete, ceramic, zinc bricks, tiles and ingots has been conducted. In these studies, Durst and Krotkov have successfully segmented out the spikes that are results of impact, from the power spectrum of the signal. They have analyzed the spectral leakage of the Fourier Transform operation as well. The most pronounced side of these studies of Durst an Krotkov for our study is that with a minimum distance decision map classifier they have been able to classify the impact sounds quite efficiently based on their material type.

Later on, Krotkov have conducted collaborative work with Carnegie Mellon University Department of Psychology. In [16] the shape invariant properties of materials are sought that are revealed from their spectral contents. In the research of Klatzky et. al. conducted with human subjects, it is stated that frequency has less contribution compared to the decay rate of the signal, in identification of material type [17].

In [18], a master's thesis is presented where a valid mathematical representation for a contact sound is tried to be formed. In this study, acoustic data is created by interaction of the object with a robot arm in a test station. Vibrations of the object is recorded by a special high speed camera. With such a setup, very pure acoustic data could be recorded; signal to noise ratio measure is kept as high as possible. A theoretical model for impact sounds is formed with the Discrete Fourier Transform of the record, frequency modes and damping parameters.

1.1.3 ROBOTIC SURFACE AND MATERIAL IDENTIFICATION

Like any other sensing and classification problem, robotic acoustic perception can be stated to have three main stages: *Signal Preprocessing*, *Feature Extraction* and *Pattern Classification*

[19]. As new technologies in computer science have emerged, first feature extraction and then pattern recognition steps have become very rich in methods and approaches. The work that include all these steps are examined as a part of literature survey.

Although [18] or other studies mentioned in the previous section can be considered as robotic perception of material or interaction type, direct applications of non-speech acoustic signal analysis and complete acoustic perception in robotics field can be considered to start with [20]. In Fitzpatrick's study, there is detailed analysis of robotic acoustic perception on several humanoid or human interacting robot platforms. In [21], there is a robot named Obrero, which can grasp objects with its force-feedback sensitive hand. Obrero is capable of recording sounds, as well as sensing the forces on its hand. Interactions with several objects like grasping and tapping is done by the robotic hand and the relation between force sensors and sound spectrograms have been analyzed.

In [22], recordings with a wheeled robot platform from various different ambiance are made and the robot is trained to find its current placement. This way an acoustic scene analysis is made. In this study, there are features from sound time signal and spectrum are extracted and used and classifiers are applied on these features. Moreover a holistic approach towards ambiance is considered and the records include very natural daily life acoustic data, this is why this study is selected for further examination. The papers [23] and [24], more complicated techniques in feature extraction called Self Organizing Maps (SOM) are used on recordings of a robot hand interacting with various different daily life objects. In these experiments, in signal processing and pattern classification, there is apparently a built in environment used, rather than manual implementation of feature extraction by MATLAB® R2008a environment (by Mathworks Inc. Natick, MA) or other coding scheme. It may be due to the usage of a built in environment that the feature extraction part of their study is not quite explanatory. They state that they reduce spectrograms in 33 dimensional column vectors and feed the SOM with these segments of spectrogram features. Usage of 33 D feature vectors still seems too much and consequently much more computational complexity, since the increase in features directly translates into considerable demand for higher computational power. This study presents very successful results on recognition of interaction and object type by acoustic means.

1.2 EXPERIMENTS RELATED TO THE SUBJECT

There are studies related to the transmission of impact noise to lower floors in apartments. One such research is conducted by National Research Council of Canada, in 1999 [25]. The conclusions derived from the results of this study is that; there are different acoustic characteristics of different floor structures. In this research, a device called tapping machine is used to produce impact sounds with five steel-faced hammers that strike the test floor. In this experiment, ASTM method E492 is used which is a standard for acoustic testing. The sound intensity levels in the lower rooms are measured in a very controlled environment [25].

In fact, the field of perceptual acoustic sensing is extensive with a large number of prospective applications exemplified in living creatures and some of these can also be considered for autonomous robots within a variety of task domains. Sensing the presence and direction of danger, sensing human and animal presence, sensing the occurrence of sudden events as well as failing mechanical functions of a robotic system can be listed as a few of these potential applications. Despite the challenges involved, sensing the direction of the these acoustic events is also a possibility with stereo or multi channel acoustic processing.

1.3 MOTIVATION AND OBJECTIVES

Considering the robotics literature, there is no particular work done found in the field of legged robots and their acoustic emission that is the result of their interaction with the environment. It is observed that, dynamically dexterous robots such as the RHex platform [26] make distinct interaction sounds with the ground. These sounds are in fact signals that are believed to include important information that is a composition of mechanical properties of the robot and the ground that is interacted with. This thesis work aims to analyze these signals and propose an utility of perception of environment to mobile robotics in general and add an effective environment sensing tool to legged robotics in a particular manner.

The three main stages in a robotic acoustic perception problem listed in subsection 1.1.3, are considered to be the three main discussions in this study and in fact form a fundamental outline of all of this thesis work. In chapter 2, approaches related to preprocessing is mentioned. In chapter 3, the full definition and explanation of the concept of feature extraction is presented

and feature selection problem is examined in detail. In chapter 4, the final stage, pattern classification is explained. This stage is in fact the actual aim to be achieved and signal preprocessing and feature extraction can be considered as prerequisites of this final stage. In other words, with the help of sophisticated pattern classification algorithms, the aim of surface perception can be achieved.

CHAPTER 2

EXPERIMENTAL SETUP AND DATA PREPARATION

In this part, the experimental platform that is the source of inspiration of this study is explained in detail. Later on, the preliminary analyses on a small data set is given. The theory behind frequency and time domain analyses that are used in this work are explained. In the final parts, the preprocessing and approaches are mentioned.

2.1 ROBOT PLATFORM - SENSORHEX

All of the experimental data of this thesis study is collected from SensoRHex Robot Platform shown in figures 2.1 and 2.2. SensoRHEX is a variation of the RHex platform with dexterous six half circular legs and it has high performance on irregular terrain due to its inherent dynamic stability [26]. In one of the modes of the motion, the robot walks with alternating tripod gait (shown in figure 2.3) and can turn around and walk backwards as well. Using DC servo motors with gearbox connected to its hips, the robot can rotate its half circular compliant legs in precisely. In the following sections, the details of the robot mechanics are mentioned. In this study, the experiment surfaces are important because they are intended to be automatically recognized by the robot. For this reason, the experiment surfaces are mentioned in the following sections and the experimental scheme is explained in detail.

2.1.1 MECHANICS OF THE ROBOT

The interaction of this robot with the ground creates noticeable acoustic signals which are in fact a mixture of ”footsteps” of the robot and motor sounds coming from the robot’s hip nodes in each actuation cycle. The ground interaction is not quite distinctive from other

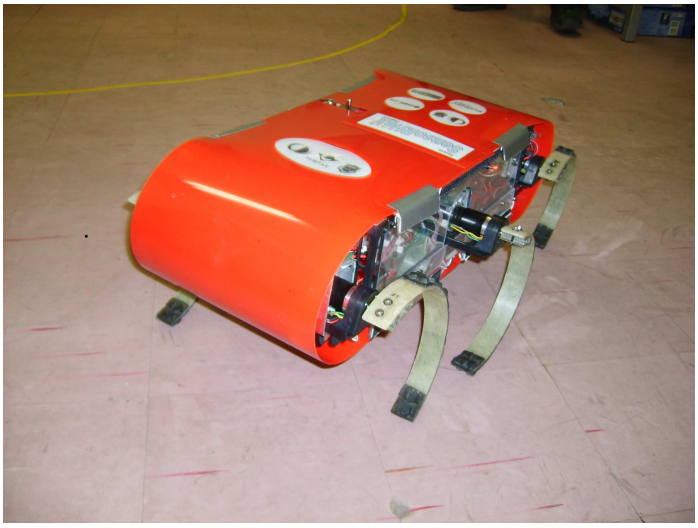


Figure 2.1: SensoRHex Robot Platform

Figure 2.2: SensoRHex is a robot platform that can walk on irregular terrain which is a useful property especially for outdoors

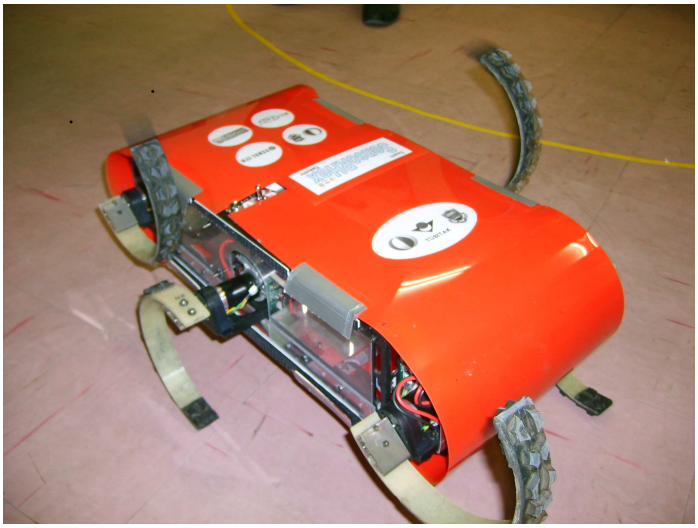


Figure 2.3: Alternating tripod gait

noises coming from the robot body; as a result of the legs being circular, there is no certain time where the interaction starts and stops. Moreover, more irregular terrain like outdoor soil covered with grass, the axial and the lateral planes of the robot frame are inclined. In such a case, it is not possible to tell the exact time when a selected leg will start an interaction with the ground.

2.1.2 RECORDING EQUIPMENT

As an initial trial, an external notebook PC is used in recordings due to the need for a sampling rate of 44100 Hz of the sound signal. This is a common standard in audio with an assumption that the average human hearing is maximum 22 kHz and by doubling that frequency as the *Nyquist sampling theorem* suggests the sampling rate should not be less than 44 kHz so that the signal can be sampled without any aliasing effect [11].

The notebook PC has an audio device which is specified as Intel Corporation 82801H (ICH8 Family) HD Audio Controller. This is only used in the very first set of experiments where only the feasibility of this work was in question. With PC, for a better control of the stereo channels, Ubuntu 9.04 operating system is used. A microphone boost level of 7/10 is fixed during the recordings. However, after checking the data, it is seen that this level is rather high for such a fluctuating sound level record and some of the impact peaks are clipped. This may result in problems in spectral analyses since Fourier Transform is based on continuity of the

Table 2.1: Properties of the Recordings

Sampling Rate:	44100 Hz
Sampling Bits:	16
Sampling Format:	.wav
Mean (Right Channel):	0.1658
Std Dev (Left Channel):	0.2738
Std Dev (Right Channel):	0.2790

signals. A peak with clipped top results in a straight line at the peak and this straight line has a different sine and cosine components which mean false frequency components that can be considered as bias errors.

A linear PCM Recorder - Olympus LS-11 is used in recording of the main experiment set (explained in the following sections). The PCM recorder is fixed on the robot and an extension cord is plugged to the stereo microphone underneath. The extension cord length is quite small (200 mm approximately) and the original gold plated cord of the microphone is used for this purpose. Windscreens of the PCM recorder are fixed and passive damping elements similar to the ones in the microphone are placed between the recorder and the robot body. The sensitivity level of the microphone is set to *high*. This setting is preferred for outdoors, conferences and places where there is ambient noise. Then the level meter is set in the vicinity of 6 as suggested in the manual of the LS-11 recorder. There is a peak warning light that lights when the sensitivity is set too high and the sound levels reach peak. After the recording parameters are set this light is occasionally observed to warn or not warned at all.

2.1.3 MICROPHONE

A stereo microphone, SONY ECM-DS70P (In Figure 2.4) is used in recordings. This microphone is mostly preferred because of being a relatively small size and affordable stereo microphone. It is thought that the saggitally symmetric design of the robot frame and body requires a stereo microphone with symmetrical sides. The microphone is mounted on the back of the robot's aluminum crash frame for the collection of the preliminary data set, on top with rubber pads for passive vibration isolation. The technical specifications of this microphone can be found in table 2.2. For additional information, Appendix-A is available. The placement of the microphone over the robot body surely has different effects. For the main

Figure 2.4: SONY ECM-DS70P Microphone used in experiments - Adapted From [27]

Table 2.2: Specifications of SONY ECM-DS70P Stereo Microphone - See Appendix - A for details

Sensitivity	38 dBV/Pascal
Response Bandwidth	100-15000 Hz
Noise Level	34 dB
Max Sound Pressure	110 dB

set of experiments, the microphone is placed in the middle of the robot frame, on the side that is close to the ground.

For the collection of preliminary data set, the notebook is connected to the microphone on the robot via an extension jack. This jack may also be a source for random errors. The microphone's jack is gold plated; however, the extension cord is a standard steel jack. Acoustic signals are captured over multiple locomotion runs with a spectral range of 100 - 15000 Hz [27]. The placement of the microphone is observed to have notable effect on the performance but this has not yet been carefully characterized. Therefore, we present our preliminary results with a fixed microphone position in the middle of the robot, as illustrated in Fig. 2.5. The processing of the acoustic signals are currently conducted off-board in MATLAB® R2008a environment (by Mathworks Inc. Natick, MA) but the robot platform has the necessary computational capability for further embedded applications.

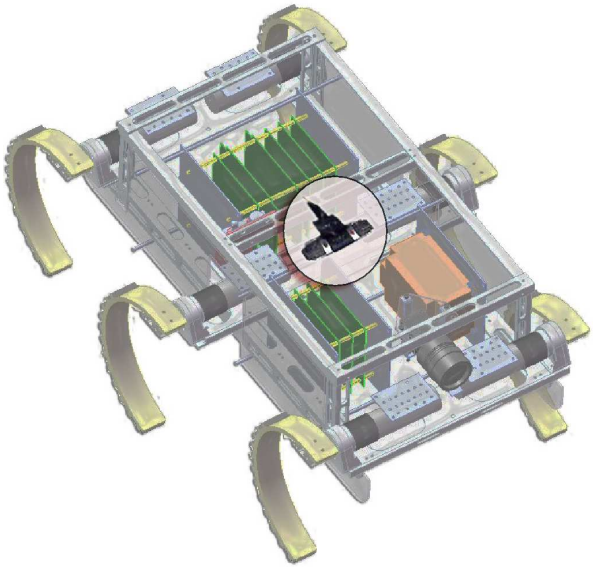


Figure 2.5: The current microphone placement within the body of the robot is illustrated. However, it should be considered that the microphone is actually under the robot body on its other side, closer to the ground

Figure 2.6: The Indoor Experiment Surfaces - C,L,R,H,S respectively

Figure 2.7: The Outdoor Experiment Surfaces - A,O,G respectively

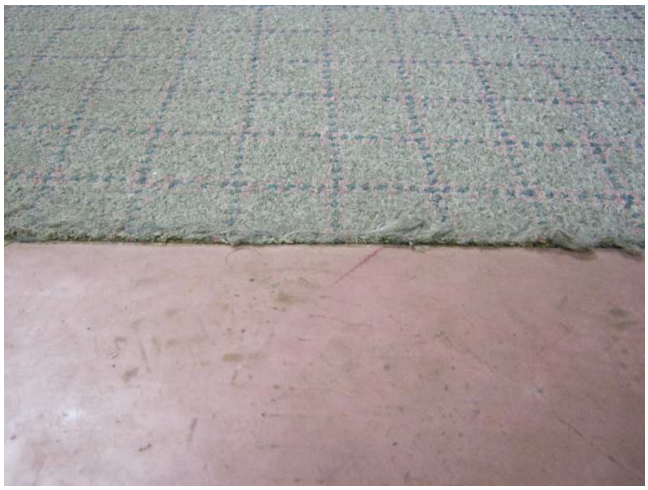


Figure 2.8: C surface is actually on L surface

2.2 EXPERIMENT SURFACES

In the beginning of the study, the data from three different surfaces is examined to check out the feasibility of acoustic surface perception. When these initial experiments have yielded successful results, new surfaces are added. The first three surfaces were carpet, linoleum and stone corridor which are denoted as C,L and S. This experiment set is referred to as *the preliminary experiment set*.

Later on, the experiments on stone corridor could not be repeated, because a large fan with electric motor which operates 24 h. is assembled to that particular corridor. Since this fan is a loud sound source, the acoustic data recorded at that environment is expected to be biased; it would certainly include that motor's operation frequencies. We expect that due to this sound, classification of that specific class would be much easier and this is certainly not acceptable for our case because this ease would not be related to surface but some other element.

This is in fact an important concern that such specific noise can accidentally mix with the useful surface acoustic data in all of the experiments. In order to prevent such a situation, the experiment ambiances are checked carefully for such sound sources. Moreover, spectrograms of each record are visually checked for existence of any harmonic data except the motor sound harmonics that is seen easily on each of these graphs. The detailed analyses are mentioned in the following sections.

Table 2.3: The Experiment Surfaces for the Preliminary Analyses

Surface Name	Notation	Explanation	Location
Carpet	C	Flexible Cover on L	RoLAB METU EE,room DB 23
Linoleum	L	Laboratory Floor	RoLAB METU EE,room DB 23
Stone Floor Tiles	S	Large Tiles	METU EE D Block Corridor

Table 2.4: The Experiment Surfaces for the Main Set of Experiments

Surface Name	Notation	Explanation	Location
Carpet	C	Flexible Cover on L	RoLAB METU
Linoleum	L	Laboratory Floor	RoLAB METU
Marble	R	Large Tiles	Control Lab, Bilkent Uni.,
Hardwood	H	Large Tiles	Classroom, Bilkent Uni.
Concrete-1	O	Large Concrete Tiles	Bilkent Uni. Outdoors
Concrete-2	A	Small Concrete Tiles	METU Outdoors
Grass	G	Random Nature	METU Outdoors

After the feasibility of acoustic surface perception with our setup is confirmed, a larger set of experiments is conducted. In this second set of experiments, seven different surfaces are used in total. Two of these surfaces are carpet and linoleum again and the rest five of them are marble tile, hardwood, outdoor concrete tile-with autumn leaves, outdoor concrete tile-straight and grass which are denoted as R, H, A, O, G respectively. The detailed properties are given in tables 2.3 and 2.4 and the photographs of the surfaces are given in figures 2.6 and 2.7.

2.3 EXPERIMENTAL SCHEME

As stated in the 2.2, the preliminary experiment set of three surfaces is used for the first analyses and their important statistical properties are given. The general properties of this data set is given in 2.5. In all of the following experiments, which can be referred as *the main experiment set*, the robot walks in a fixed linear trajectory with constant speed and under operator control. During the experiments, the robot is turned around by the operator when needed. However, in the final records that are to be tokenized, there is no turning sound involved.

Table 2.5: The Recording Properties for the Preliminary Analyses

Surface Notation	Recording Number	Speed	Record Length
C	1	1	1.12
L	1	1	1.24
S	1	1	1.52

Table 2.6: The Recording Properties for the Main Set of Experiments

Surface Notation	Recording Number	Speed	Record Length
C	3	1	2.57
L	1	1	1.52
L	3	1	4.14
R	2	1	1.37
H	2	1	2.13
A	3	1	3.07
G	3	1	3.22
C	3	5	4.02
L	3	5	2.58
R	2	5	1.45
H	2	5	2.24
O	2	5	3.04
A	3	5	2.14
G	3	5	3.53
C	3	5	3.36
L	3	5	3.09
R	2	5	1.09
H	2	5	2.23
A	3	5	2.20
G	3	5	3.40

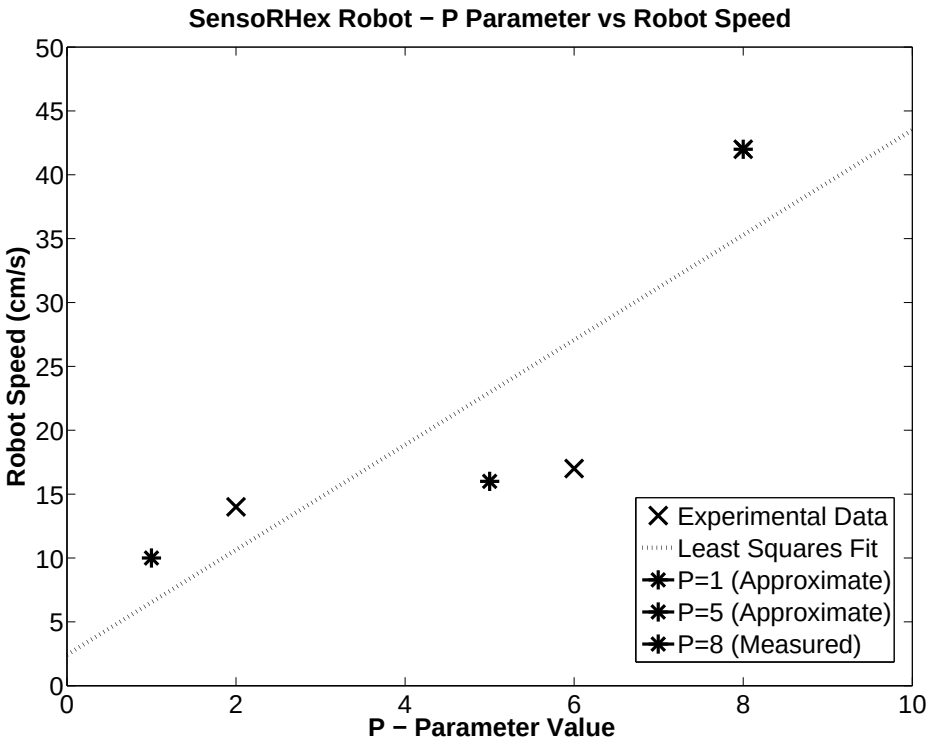


Figure 2.9: The corresponding physical speed for the values of parameter P of SensoRHEX platform is measured in [28].

The experiments are conducted for each surface for three different speeds. For a hexapod robot platform with compliant legs like SensoRHex, it is not possible to measure the exact speed of the body frame since the focus of the gait is on moving on any terrain rather than the precision of the position. For this reason, a low speed set, a mid speed set and a high speed set of data is recorded. In the robot platform control, there is a parameter in gait control which is directly related to the robot platform speed. This parameter is denoted as P in this work. The approximate corresponding physical speed for different values of P is shown in figure 2.9. These values are taken from the experimental work on the actual platform [28]. In the figure 2.9, there is a least squares fit is plotted as well. The recordings are taken for $P = 1$ for low speed, $P = 5$ for mid speed and $P = 8$ for high speed. With the aid of the measured values, the approximate corresponding physical values of the recorded speeds $P = 1$ and $P = 5$ are marked in figure 2.9 and there is already experimental value measured for $P = 8$ [28].

For each recording speed, approximately 2 minutes of recording is taken. In such an experiment, 200 steps are recorded for each surface but obviously, this depends on speed. Although

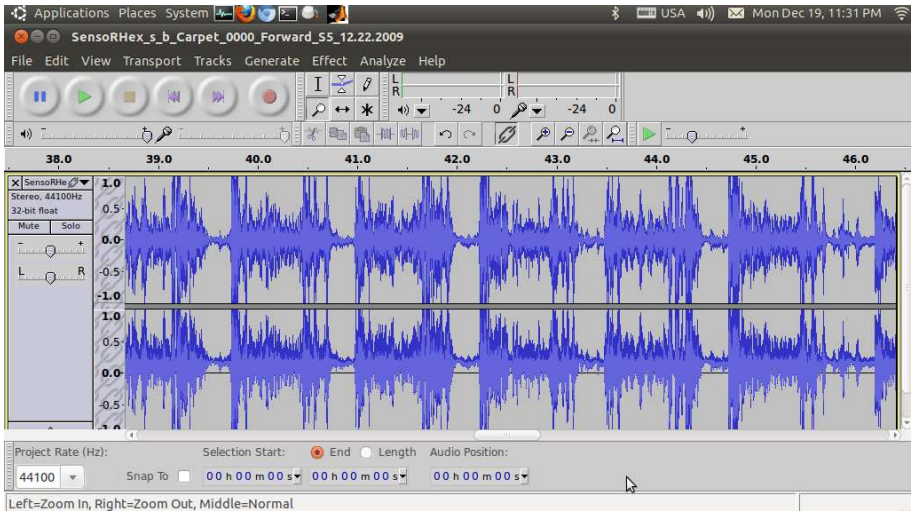


Figure 2.10: The stereo record from the Carpet Surface as .wav file loaded in Audacity.

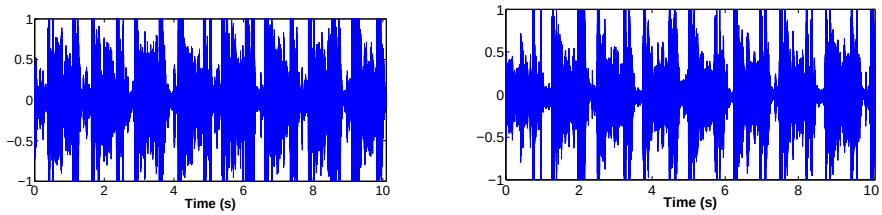


Figure 2.11: Comparison of approximately 1 s records of C and L. In here, the signals start with motor sound peak and end with ground leg interaction peak and the diminishing wave of the surface impact.

the record lengths change for different surfaces, shortest length record is considered and the rest of the data is reduced to this length for other surfaces. The signals are filtered out from the turning sounds and irrelevant sound data between the forward walking sessions. All of this work is done in Audacity 1.3.12-beta environment which is a free sound processing tool compatible with both Ubuntu 9.04 and Ubuntu 10.10. In Audacity, it is possible to analyze the signal in 15 digit floating point numbers varying between -1 and 1 [29]. The illustrative figure of a waveform is given in figure 2.10. In comparative figure 2.11, the two samples from two different surfaces do not seem to differ very much in time domain and it is quite hard to analyze and notice the outstanding specific features out of this data.

2.4 ANALYZING THE SIGNAL

As it is stated in chapter 2, the recorded audio signal is cut out of its sections that contain turning and backwards walking sounds. The analyzed signals include only the straight walking sounds at constant speeds. When these recorded signals are examined, there are motor start and ground impact regions that create peaks with each footstep of the robot 2.10. In alternating tripod gait, the robot's three legs are expected to touch the ground at one time and although in most of the recordings this is observed to be so, there are some footsteps sounds that one leg touches earlier than the other two. In these cases, the signal has two consecutive peaks. There is a rolling region where the robot's half circular legs roll to complete one tour.

As classification units, namely *patterns*, to be classified, small portions of the recorded signals are defined as *tokens*. Each of these tokens are small units that serve the purpose of surface identification and the robot is expected to decide for the surface category of each token. The signals recorded for each surface are stated to be cut into the length of the shortest one, therefore, this way a standard size and number of tokens for each surface could be obtained. The selection scheme of these tokens are explained in the following section, 2.5.

The examination of the recorded time domain signals could not be detected to show any difference from surface to surface with naked eye. When these signals are listened by bare ears, there is also no possibility of differentiating one surface from other. Although analysis of time domain data includes important features, such as *Zero Crossing Rate*, the distinctive analysis mostly depends on frequency domain analysis.

There are other very important time domain acoustic identification features widely used in sound processing applications such as reverberation time which is a property of spaces that is related to echoes [30]. However, only zero crossing rate is assumed to be relevant to our case since there is only sounds related to ground surface interaction are considered to be analyzed.

The Zero crossing rate has a non-trivial relationship with frequency content of a signal but still is computed in the time-domain. It is expressed as

$$r_{zc} = \frac{1}{2} \sum_{r=0}^{N(M-1)} |sgn[x(r)] - sgn[x(r-1)]| w(m-r) \tag{2.1}$$

where $sgn[.]$ is the standard sign function, $w(n)$ is a rectangular window and $Z(m)$ is computed

Table 2.7: Statistical Properties of the Data - The Preliminary Experiment Set

	Carpet	Linoleum	Stone
Data Size (Stereo Record):	3718682	4957716	3204345
Mean (Left Channel):	-0.0218	0.1491	0.1702
Mean (Right Channel):	0.1658	0.1658	-0.0211
Std Dev (Left Channel):	0.2738	0.4785	0.3576
Std Dev (Right Channel):	0.2790	0.4836	0.3490

as an average for each token $m = 1, 2, \cdots, M$.

2.4.1 STATISTICAL PROPERTIES OF THE DATA

For a preliminary analysis, the recorded tracks are loaded into MATLAB® environment. As a worldwide used powerful tool, MATLAB® offers some utilities that allow users to implement basic but very important statistical operations. Table 2.7 values are calculated by the MATLAB® built in functions: `mean()`, `std()` and `length()`. The maximum amplitude for all signals is 1 and the minimum is -1 since the signals are normalized in time domain.

2.4.2 SPECTRAL EXAMINATION OF THE DATA

Fourier Transform has been an important tool in sound processing area that enables to transform a time domain function to frequency domain so that the frequency components and harmonics in signals can be analyzed. Fourier transform applied to a finite length discretely sampled data is called *Discrete Fourier Transform (DFT)* and the frequency content of the data for a a finite length signal can be revealed very clearly with implementing the *Fast Fourier Transform (FFT)* algorithm [11] which is a computationally effective implementation of DFT. The transform for a discrete signal with finite length is given as

$$X[k] = \sum_{n=0}^{N-1} x[n]w[n]e^{-j(2\pi/N)kn}; k = 0, 1, ..., (N-1), \tag{2.2}$$

.

Here, $x[n]$ stands for discrete time domain signal with N samples and $X[k]$ stands for the transformed signal, which is complex and is not suitable for direct use. Since this analysis is

conducted with finite length signals, there are expected to be preceding and receding frames for a broad analysis and a single analysis would not represent the overall phenomenon since there are changes that are time dependent. To analyze such a signal, *time-dependent Fourier transform* (also named as *short-time Fourier Transform*) is used. Time-dependent Fourier transform is represented by

$$X[n, \lambda) = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m}, \tag{2.3}$$

.

where λ is a frequency variable and $w[n]$ is a windowed sequence [11]. As seen with the expression, the resulting $X[n, \lambda)$ is a function of time and frequency. The Hanning window is expressed as

$$w[n] = \begin{cases} 0.5 - 0.5cos(2\pi n/D), 0 \leq n \leq D, \\ 0, otherwise \end{cases} \tag{2.4}$$

.

and this window is preferred because it starts and ends with zero value for the given interval $[0 - D]$. For each consecutive sample, the values at the beginning and the end of the signal are expected to be non-zero and this affects the assumptions of Fourier Transform [11]. To reduce this effect, Hanning window is preferred. A 1024 sample hanning window ($D = 1024$) can be seen in figure 2.12.

For the purpose of analyzing the signals in both time and frequency domains, a larger signal like a token is divided into regions called *windows* and their spectrum are estimated separately. To reduce the effects of truncation of the signal from these surrounding frames, a tapered multiplier window $w[n]$ is usually introduced while the DFT is computed [11], [5]. The Hanning window is preferred since For a more smooth estimate, these windows are preferred to be overlapping up to a given percent.

Expression 2.3, that is referred to as *Short Time Fourier Transform (STFT)* in literature as well where with calculation, $X[n]$ becomes N-point DFT of a finite length signal. The power spectrum of the signal can be estimated from this complex expression. It is given as

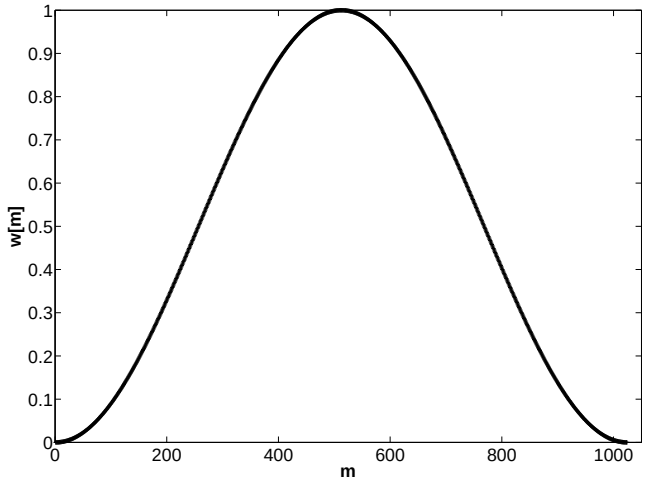


Figure 2.12: Hanning Window - D=1024 Samples

$$S[k] = |X[k]|^2 = X[k]X^*[k]. \tag{2.5}$$

where $S[k]$ of the time domain signal which is in the end a real sequence [11]. With combination of the surrounding frames, an average power spectrum estimate for a larger signal can be estimated. For implementation of FFT algorithm, the N value should be selected from the powers of two [12]. For estimating the power spectrum for all length of signals, zero padding up to the next power of two is implemented to the final frame to complete the analysis properly. The power spectrum estimate is directly related to the intensity of the sound energy [12], [30]. The ensemble averaged power spectrum estimates with 20 samples for the preliminary data set is given in figures 2.13. These spectrum plots seem to be very similar to each other except at the lower frequency values. However, the first samples always seem to be very different may correspond to frequency value that is below 100 Hz and therefore should not be considered as a feature since the microphone data is not valid below 100 Hz. Since the signal is not stationary, the ensemble averaged plots may not contain distinctive properties. The changes in sound energy with respect to time can be seen more in detail when the smaller time frames are analyzed.

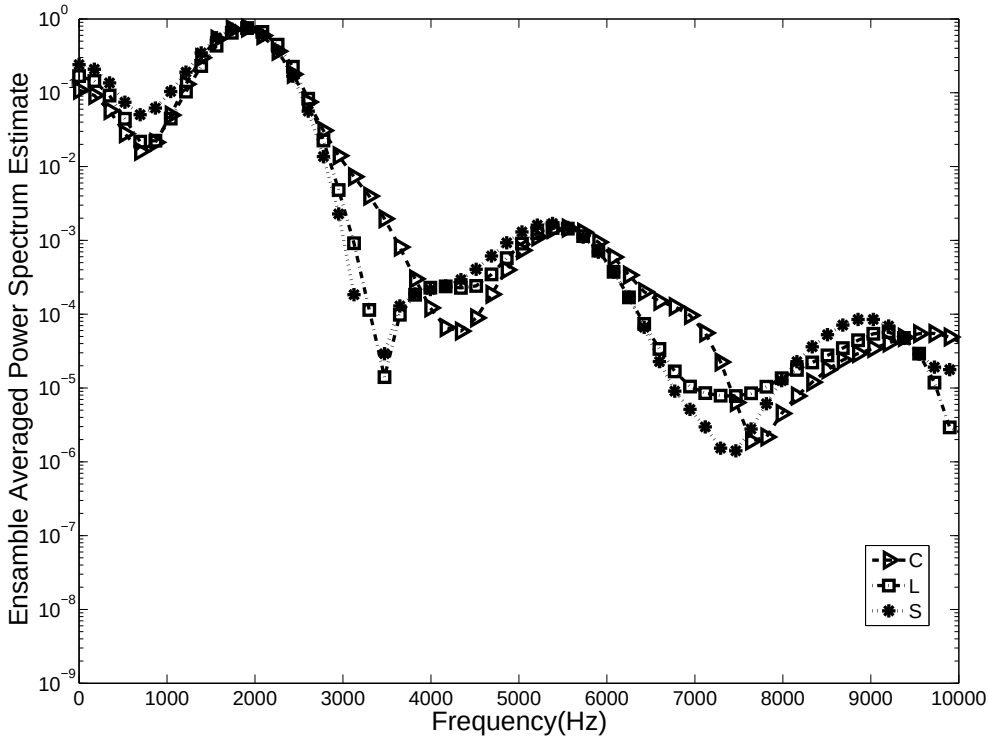


Figure 2.13: Ensemble Average Power Spectrum Estimate of the Record on Carpet, Linoleum and Stone. The ensemble average plots may not be very informative since the signal is non-stationary.

Table 2.8: The Spectrogram Properties of the Images

FFT Size:	1024
Resolution:	1024
Window Overlap:	50%
Window:	Hanning
dB scale:	0 (Dark Blue) - 150 (Red)
Computer Environment:	Matlab Signal Processing Toolbox (Spectrogram() Command)

2.4.3 SPECTROGRAMS OF THE DATA

For a better intuition and understanding of the problem, the recorded signals are first listened by ear and then their spectrum changes with time are observed with the aid of spectrogram tool. This has been a very important preliminary analysis in the study. Because, the average power spectrum estimates formed with divided frames of the larger signal is in fact collapsed in time axis and does not provide any time information anymore. In practice, this results in mixed spectrum of and footstep sound regions which in fact does not yield very clear results. In other words, motor and footstep sounds are desired to be examined as distinct power spectrum estimates and spectrogram tool visualizes these power spectrum frames very well. The spectrogram image can be estimated with the formula given in equation 2.3.

In figures, 2.14, 2.15 and 2.16, the spectrogram images of the acoustic signals are presented and their properties related to representation is given in table 2.8. In these spectrograms, there are small rectangular regions observed. When the regions are close to red, this indicates there is more energy in the corresponding frequency band and when the color goes to blue, this indicates there is lower energy. In all of three spectrogram images given, there are two interaction sounds with the ground, however there are four energy dense regions observed (see red regions). The first and the third peaks result from the interaction sounds whereas the second and the third regions are a result of motor sounds. The motor sound was shown on the recorded signals to create high peaks in time domain and this is shown once again that the sound energy density is quite high in the times when the legs rise from the ground. There are also rolling regions of the three different surfaces of the preliminary experiment set as shown in these figures. The spectrogram images give important clues about the sound energy difference due to the impact with the different surfaces. Unlike the ensemble averages, in spectrograms, the changes with respect to time can be observed.

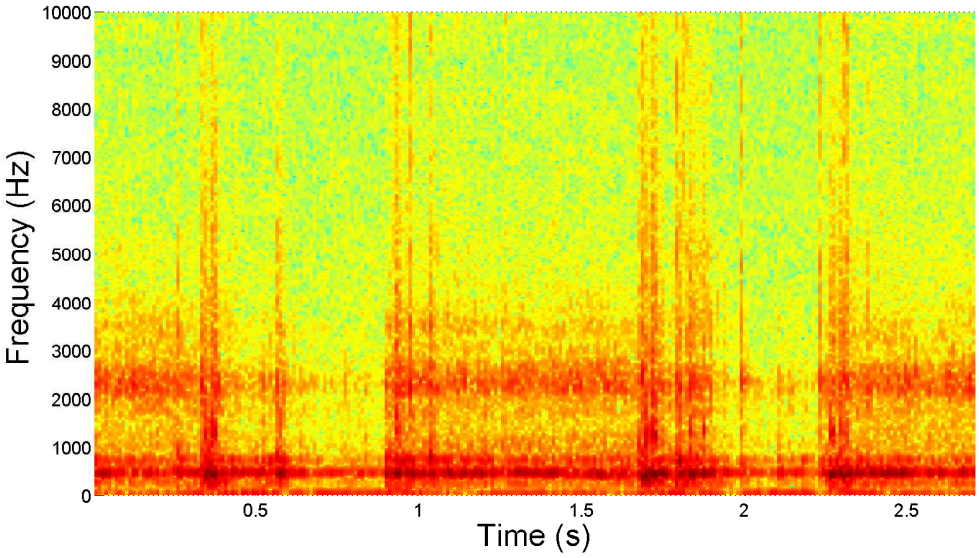


Figure 2.14: Spectrogram of Two Interactions with a Carpet (C) Surface.

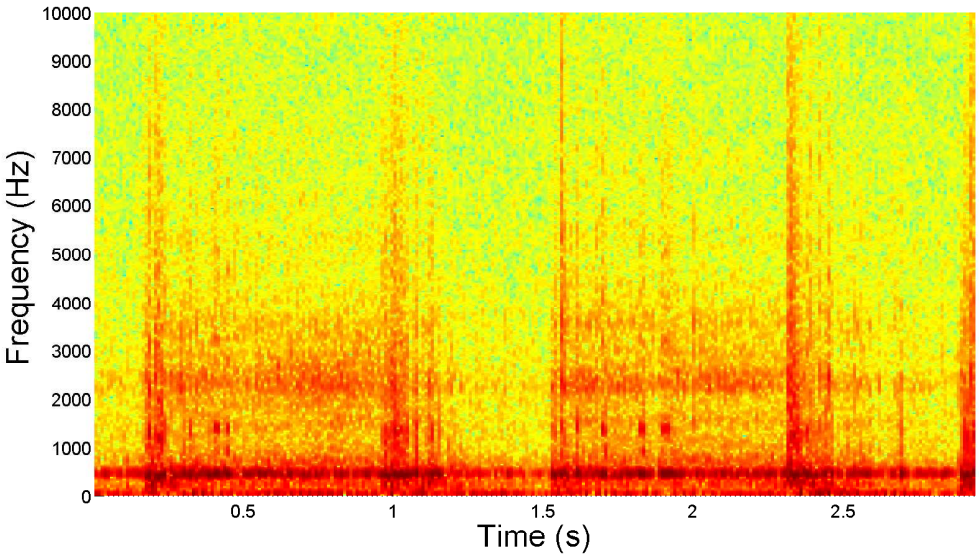


Figure 2.15: Spectrogram of Two Interactions with a Linoleum (L) Surface.

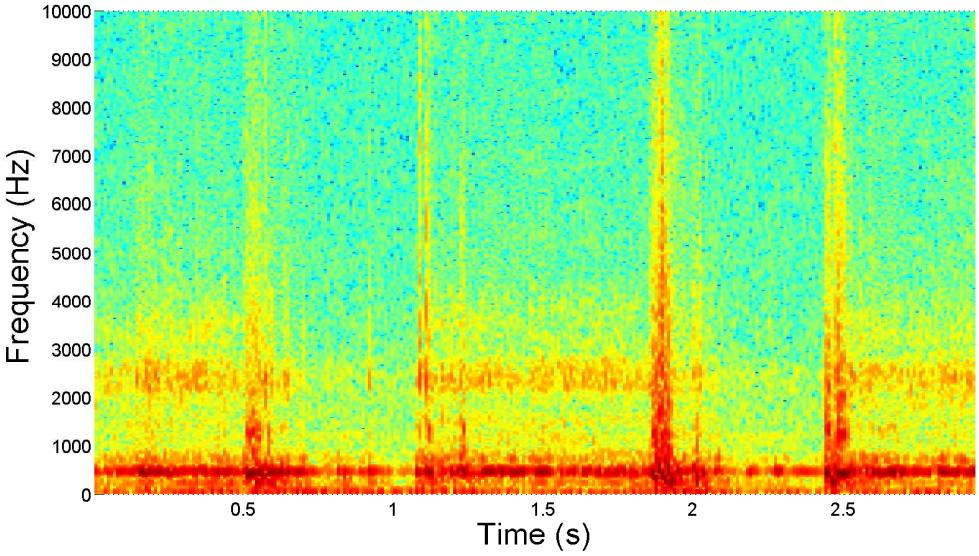


Figure 2.16: Spectrogram of Two Interactions with a Stone (S) Surface.

2.5 PATTERN SELECTION FOR CLASSIFICATION

After some general purpose inspective analyses are conducted on the preliminary experiment set, some time is spent on the utilization of the recorded signals. The units of classification are often referred to as *patterns* in the pattern classification literature [19]. The properties of patterns depend on user selection. Like all of the other works on pattern classification, the patterns should be defined first, before passing on to other procedures. Patterns should include definitive properties and in the end of classification, they are expected to be labeled with a certain class. In the following subsections, all of the approaches to obtain relevant patterns from the sound signals are explained.

2.5.1 SEGMENTATION APPROACH

After examining the composition of the spectrograms and the formation of average power spectrum representations, it is decided to extract the motor sounds from the overall signal. In pattern classification procedure, such extraction operations are called *segmentation*, [19]. With segmentation it is aimed to have an easier feature extraction process since the signal is extracted out of the elements that are thought not to be relevant to classification. The segmentation is intended to be implemented on pure motor sound regions.

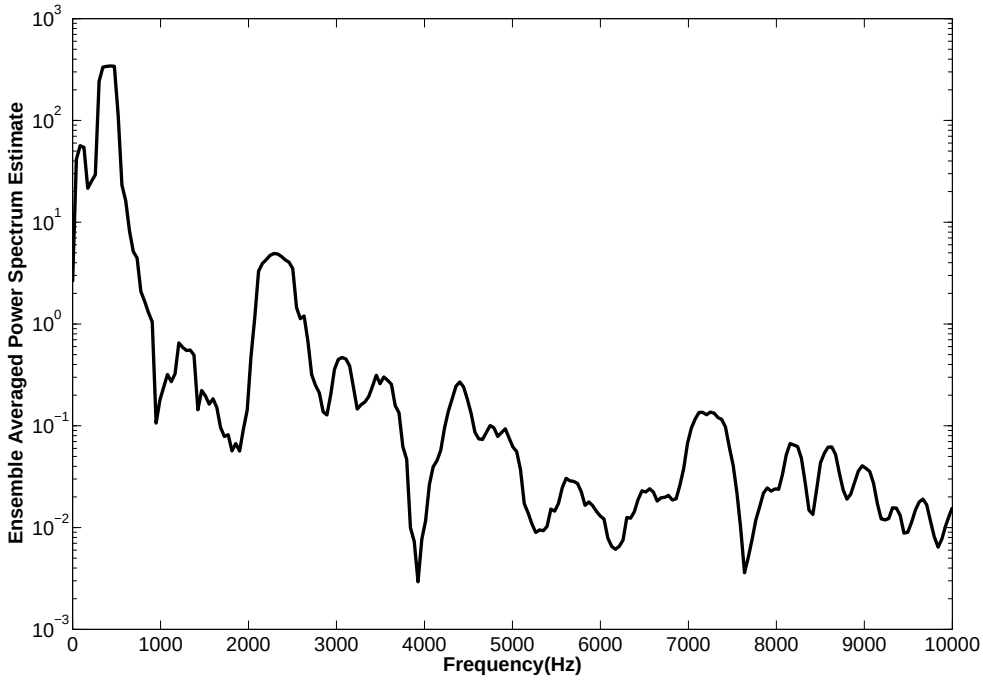


Figure 2.17: Ensemble Average Power Spectrum Estimate of the 20 Segmented Motor Sounds

Various methods in both time domain and frequency domain data like looking for threshold peaks, averaged energy values had shown that the ever present motor noise is hard to segment from the ground interaction noise due to time-domain and spectral similarities. A more powerful and commonly used method called *Spectral Subtraction* [31], is tried later on and its explanation can be found in subsection 2.5.3.

The results of these segmentation approach have not been adequate enough to claim considerable success. Therefore, a more holistic approach is emphasized in the following studies and pre-filtering or segmentation to evaluate the resulting classification performance is not preferred under these conditions. An audio file with pure motor sounds is formed with the successful automated segments of the signal. 20 samples are collected in total from three different surfaces of the preliminary data set. The average spectrum estimate of this merged motor audio file can be seen in figure 2.17. As clearly seen, there is a considerable peak around 1900 - 2100 Hz and the rest of the signal is similar to the other spectrum figures of surface interaction.

Table 2.9: Signal Lengths and Token Sizes for Each Speed - The Main Experiment Set - Holistic Approach

	Speed-1:	Speed-5:	Speed-8:
Overall Signal Length:	4662310	3000000	2000000
Signal Length (Per 25 Tokens):	186492	120000	80000
Record Time (Per 25 Tokens):	4.23 seconds	2.72 seconds	1.81 seconds
Signal Length (Per 50 Tokens):	93246	60000	40000
Record Time (Per 25 Tokens):	2.12 seconds	1.36 seconds	0.91 seconds

2.5.2 HOLISTIC APPROACH

The second approach, namely *the holistic approach* has yielded very successful results that are presented in chapter 5. In this approach, window frames for each token are selected such that there is no distinction between motor and interaction sounds. There can be pure motor regions in the spectrum average or some irrelevant other frames since the experiments are not conducted in totally silent environment. This approach is found to be a more natural perception approach, since in daily life, living creatures with hearing ability are not known to have any specific sound spectrum cancellation or deletion mechanisms so far to the best of our knowledge.

2.5.3 MOTOR NOISE EXTRACTION APPROACH

After the studies with holistic approach have reached to some maturity, spectral subtraction method is applied to some controlled experiment sets and performance improvement is examined. Spectral subtraction can be performed with

$$Z[k] = (S[k]^{\gamma} - \alpha Q[k]^{\gamma})^{1/\gamma}. \tag{2.6}$$

.

In here, $Q[k]$ stands for the power spectrum estimate of the motor noise signal and $Z[k]$ stands for the subtracted signal. With α as a weight to $Q[k]$ and γ as a power and root, two new parameters are introduced into analyses. As it can be seen in chapter 5, there is not much of an improvement obtained from various values of α and γ. This may be due to either there

is useful data removal caused bu subtraction or among selection of α and γ, better values are missed.

CHAPTER 3

FEATURE EXTRACTION

In chapter 1, the role of feature selection and extraction is explained briefly. In this chapter, it is aimed to explain all of the relevant details of this essential procedure that is implemented in this work. To define feature selection, one should start with definition of *features*. If the units of classification are called patterns as defined in 2, features can be stated as the distinctive properties of those entries that are intended to be classified [19]. For various problems that require automated decision making in various applications like industrial, medical, research and many others, application specific features are used. Stated as probably the most common data set used in pattern classification literature [32], Iris data set is one of the many examples to a set of features. Iris is a kind of plant with dazzling flowers and this data set includes the length and width of sepals and petals (in centimeters) of three different *iris* species: Iris Setosa, Iris Versicolour and Iris Virginica as attributes. The sepal and petal measures are the sets of features that differ from one Iris species to other. They can be expressed numerically and play important roles in the iris species classification.

3.1 APPLICATION SPECIFIC SELECTION

In the iris data set mentioned in the previous paragraph, the length and width of sepals and petals are the attributes that let the classification procedure to be applied. These four attributes are selected out of the quantitative or qualitative properties that are related to iris plants. In this case, four numerical values are found sufficient enough to determine the species. However, for various kinds of problems, there are not only numerical attributes to be involved. In addition to numerical attributes in pattern classification, there are also binary attributes called *predicates* and there can be nominal attributes applicable as well, depending on the classifier

input [19], [33].

By looking at different applications such as visual recognition of processed images compared to acoustic perception, the features in each case are expected to be different and quite application specific. In visual perception, the source data is a 2D signal whereas in acoustic perception via sound processing gets its data from 1D sound signal. Even for two different visual perception applications, the relevant features would be very different.

To illustrate the case explained in the previous paragraph, let one application be to identify the severity of cracks and irregularities of mass produced products on a conveyor belt and other to be a medical image of the brain to identify abnormalities. The features to identify and classify the cracks and irregularities on a product would obviously very different from the images of the legions or the abnormalities in the brain image.

In the pattern classification literature, this set of relevant attributes are referred to as *feature vectors*. The classification is applied to the data sets that are assumed to be definitive features to form feature vectors, however, additionally, there are more complex algorithms such that they are able to identify the success of the selected features by giving them a weight of contribution in the final decision [19].

3.2 THE PROBLEM OF HIGH DIMENSIONALITY

Even in the case of iris data set, which yields a rather simple classification problem compared to many other applications, the number of dimensions of attributes is four which is in fact impossible to visualize in three dimensions without any projection. This is often referred to as *The Curse Of Dimensionality* in the pattern classification literature. The need for more features obviously brings more dimensions to add into the feature vector and projection to one axis is not a preferable way to reduce dimensions since it results in loss of data. An example of loss of data with projection is illustrated in figure 3.1. In this example, there are two features that define two classes (The points with red and blue are separate classes). In the 2D feature space, there is a line between that separates these two classes almost perfectly, however when only one of the features is considered (either feature 1 or feature 2), there is not much of a success in classification.

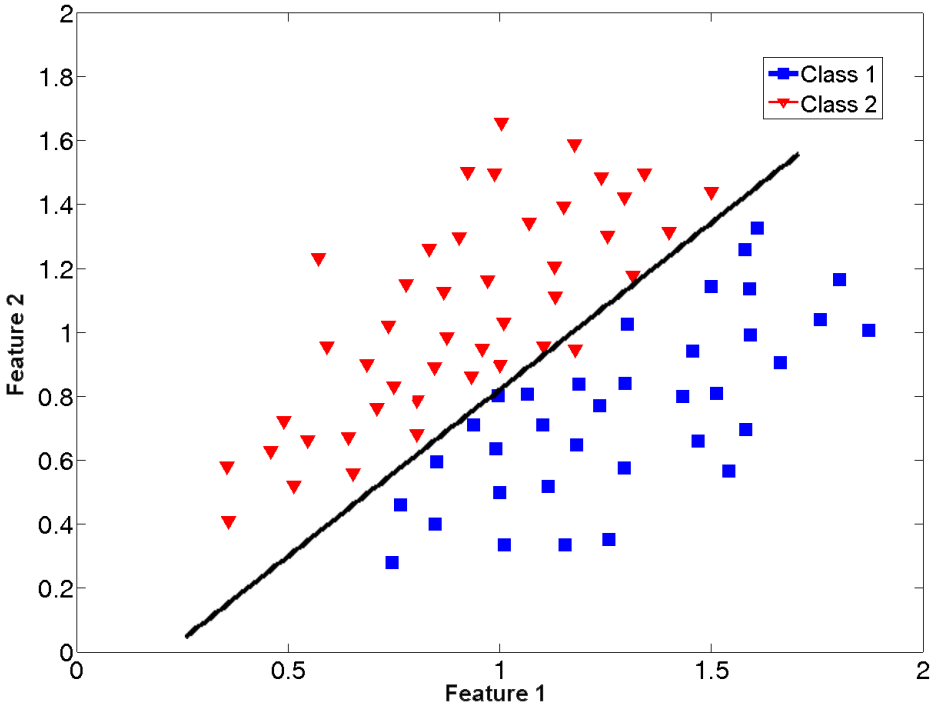


Figure 3.1: A two class problem. The projection to neither of the feature axes yields a successful classification.

Most of the pattern classifier algorithms that are implemented widely can tackle with high dimensional data. The feature vector required for a feasible classification may be much more complex and higher dimensional. A widely used application of pattern recognition in cameras, security systems and various other applications is human face detection. In the earlier approaches to face recognition problem identification of parts of face were used according to their geometrical properties however, for a more efficient recognition process, there are sophisticated statistical learning methods that handles with nonlinearities in face images developed in the recent years [34].

Many of the sources in the literature support that [19],[33], more number of features tend to increase the classification performance since they are relevant. However, it is clear that higher dimensional feature vectors lead to computational complexity. Moreover, increased number of features do not necessarily mean higher accuracy in classification. Irrelevant features may increase noise in feature matrix and they may lead to performance drop in classification by causing confusions. Therefore, there is an optimization process; the number of features should be selected as high as possible such that the performance will be increased and the computational complexity will not be increased up to an intolerable value.

3.3 SELECTION OF RELEVANT FEATURES

In most of the classification problems, there is often a huge amount of data like sampled sound data points or number of millions of pixels. If sound is considered, 1 second of sound record sampled at 441000 Hz contains 44100 points which is a huge number to handle and manipulate. By feature extraction, this huge number is reduced to a bunch of numbers that are inserted into a feature vector for further manipulation by classifier. Compared to millions of data, both in means of computational complexity and storage, the extracted feature vector is much more advantageous.

Another probable useful outcome of feature extraction is the elimination of noisy data. Features can be stated as the fundamental properties of a pattern. The overall data derived from each pattern is expected to include many imperfections related to measurement and data acquisition errors. The weight of these noisy elements are expected to be reduced since the selected features are expected to yield the most relevant information.

3.4 SIGNAL PROCESSING

As it is stated in the chapter 2, Fourier Transform of a sound signal yields very important results about its spectral contents when handled delicately. The power spectrum is derived from the energy of the FFT vector and this spectrum has the same number of points as the FFT transform. This is still too many number of points for a feature vector since a feasible FFT size is at least 32 and depending on the signal, the selected FFT size should be increased [12].

After power spectrum is derived, more operations related to feature extraction is performed to obtain the relevant features and to reduce the number of entries to the feature vector. An additional operation of calculation of zero crossing rate is done in time domain and the derived feature is added to the feature vector if desired. All of these procedures are going to be explained in the following subsections and the details of the program flow and programming approach is presented in the final subsection of this chapter.

3.4.1 FREQUENCY DOMAIN

Given in equation 2.2, in *Frequency Domain Examination of the Data* subsection, the frequency domain transform is applied to each of the tokens that are explained in chapter 2. In application level, there are various parameters that have effect on the power spectrum estimate like FFT size N, signal window size and overlap,smoothing window $w[n]$ type, Various FFT parameters are selected 3.1. While selecting those parameters, certain signal properties and analyses are taken into account. In [12], if the Nyquist frequency is given as 44100 Hz,

$$1/2\Delta = 44100 \tag{3.1}$$

then the time interval for sample Δ is found to be 0.000011338 s. If effective bandwidth is expressed with B_e, and accuracy is defined as σ_m/m_m where m_m is the mean value of the signal and σ_m is the standard deviation; then,

$$B_e T = 1/(\sigma_m/m_m)^2 \tag{3.2}$$

Table 3.1: Parameters in Frequnecy Domain Transfrom with FFT

FFT Length	Window Samples	Window Type	Window Overlap	Time Per Token
64	50	Hanning	50 %	0.9845 s
128	80	Hanning	50 %	0.8950 s
256	128	Hanning	50 %	0.6554 s
512	250	Hanning	50 %	0.6709 s
1024	800	Hanning	50 %	0.001477 s

expression may be useful in finding the length of the record required for the other given parameters [12]. Finally the number of sample points N should be related to Δ and T expressed in equations 3.1 and 3.2. Finally, if N is given as

$$N = T/\Delta \tag{3.3}$$

then all three equations are related and can be used in setting the analysis parameters [12]. It should also be noted that, as stated in 2.4.2; for feeding a discrete signal to an FFT algorithm, there is usually zero padding operation required to the powers of 2. Therefore, L_z zeros should be added to N length signal. Some reasonable sets of parameters are defined in 3.1. For a window length of 128 points, by using formula 3.3, T is found as 0.002902494 s approximately. With using formula 3.2, if an accuracy of 1/3 is desired, the effective bandwidth is found as approximately 3000 Hz. If an accuracy of 1/2 is desired, then the effective bandwidth is found as approximately 1000 Hz. These values may seem a bit coarse, however, when the overall spectrum is considered to be dropped down from around 100-10000 Hz to a bunch of features, these values seem reasonable and are considered in feature extraction operations.

3.4.2 TRAPEZOIDAL ENERGY BANDS

After estimating the averaged power spectrum, there comes the procedure of extraction of relevant features. In sound processing literature, there are many filtering methods of the spectrum. One of the applications that are widely used is the triangle shaped *Mel Frequency Filters* applied for recognition of human speech [5],[35]. In the current application, however, there is not a preliminary data on the informative bands since the robot footstep sound content

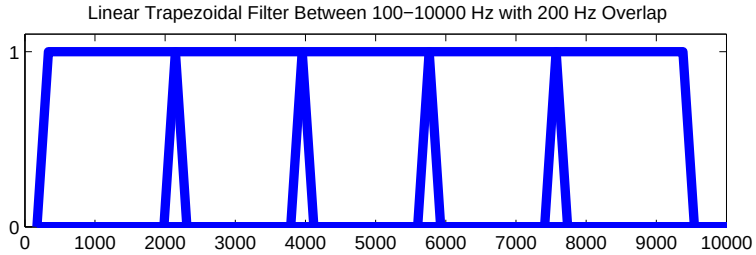


Figure 3.2: Linear scale trapezoidal energy filters that average spectral energy on predetermined number of overlapping frequency bands.

for this specific application is examined for the first time. Anyhow, the spectrogram images that are examined for a preliminary intuition about the sound signal in fact present useful information about the important energy bands in this signal. As explained in the previous chapter and seen on the spectrograms, the bands between 100 Hz - 500 Hz are quite dense in sound energy. The rest of the spectrum also seem to contain relevant data about the signal.

Although there is higher energy observed in lower frequency bands, the contribution of frequencies to classification is still unknown. Therefore, a general purpose frequency filter is formed to obtain averaged spectrum features. The filters formed have trapezoidal shapes and they overlap on each other. This amount of overlap is another experiment parameter and supplied in units of Hz.

The energy for each band is estimated by multiplying the averaged power spectrum matrix with a corresponding filter matrix. This filter matrix has values between 0 and 1. The larger the frequency overlap becomes, the finer the values get. This forms the inclined regions of the trapezoid. The flat sides of the trapezoids are either 1 or 0.

The energy for each band E_b is then defined as

$$E_b = \sum_{k=0}^{N/2} S[k]Z_b[k]; b = 1, 2, \cdots, B. \tag{3.4}$$

for b being the energy band (one trapezoidal area). It is clear that, as B gets higher, the final feature vector size increases and so is the computational complexity of the process. It is expected that the classification performance may increase too up to a certain value and therefore B is the value to be optimized. With considering the calculations in subsection 3.4.1, the number of trapezoidal energy bands are not selected to be more than 15 in all of the

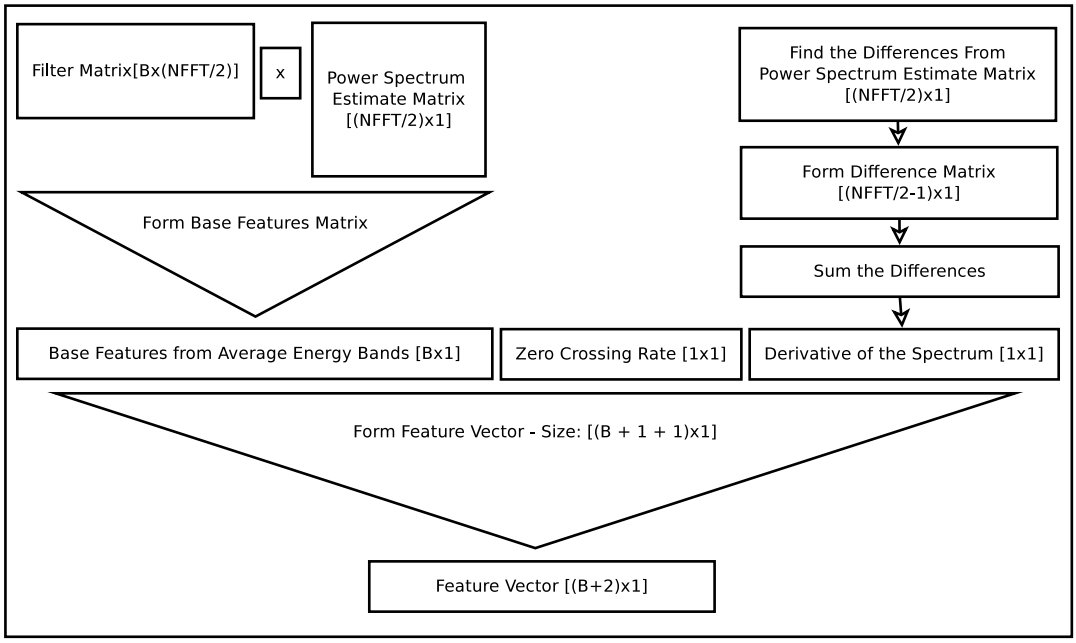


Figure 3.3: The final feature vector when all of the mentioned features are included

studies since after this number $((10000 - 100)/15 = 660$ Hz), the effective bandwidth of the analysis is not found enough in terms of accuracy for the selected FFT size and window size.

3.4.3 ZERO CROSSING RATE

Zero crossing rate as another important feature is added to the feature vector and its effects are studied on. With the help of formula 2.1 given in 2, r_{zc} is calculated. Being patented, zero crossing rate is known as distinguishing whether the content of a sound signal contains speech or not [36] since, due to the physical shape of the human sound producing organs, the speech signal has low zero crossing rate values and high spectral energy. Since in this case the recorded signals are unvoiced, the expected zero crossing rates are high. There may not be dramatical difference of r_{zc} for each surface however it is still believed to have a high contribution to classification and this assumption is verified in the following sections. The zero crossing rate feature is used with a coefficient of 0.001 for being comparable to other features in magnitude. This coefficient is determined heuristically with considering various feature sets.

3.4.4 DERIVATIVE OF THE SPECTRUM

As it is stated in chapter 2, there is a large number of features that can be derived from a sound signal. After forming a variable size $(B + 1)$ feature vector with average power spectrum features and zero crossing rate, there is a final feature added to the feature which is called derivative spectrum vector. This vector is denoted as ΔSF. For a power spectrum estimate $S[k]$ with N points, let $S[k]$ *spectrum difference* be

$$\Delta S[k] = |S[k] - S[k+1]| \tag{3.5}$$

where k is the kth element of the spectrum, there will be $N/2-1$ points of spectrum differences since the two consecutive spectrum elements are subtracted. This can be stated as a

$$a_{sds} = \sum_{k=0}^{(N/2)-1} \Delta S[k]; m = 1, 2. \ldots, M. \tag{3.6}$$

CHAPTER 4

CLASSIFIERS USED IN THIS WORK

In the chapter 3, the details of forming a feature vector with implementing the procedure of extracting the relevant features are given. In this chapter, it is aimed to explain the final step in the classification scheme, the pattern classification step. For pattern classification, there is a large number of algorithms available in literature [33] and in this work, only four of these algorithms are selected after an exhaustive analysis on performance of other classifiers is done.

A rule of thumb in the literature of pattern classification is *No Free Lunch Theorem*. This theorem states that there is no superiority of one classification over another before each of the algorithms are implemented. Even compared to random guessing, there is still no superiority [19]. Therefore, the performances of various classifiers should be tried and a classifier that performs well on this specific problem should be selected.

4.1 ALGORITHMS IMPLEMENTED

For the preliminary experiment set, a popular and widely known algorithm called *Vector Quantizer* is implemented by coding in MATLAB® environment. This algorithm have yielded very successful results for three classes. By looking at these results, the feasibility and the applicability of this work is demonstrated once again. However, when the main experiment set is collected, a performance drop is observed after four classes. To tackle with this problem, various other algorithms are implemented on the main experiment set with six classes and some of these algorithms have yielded successful results. The implementation is done in WEKA environment which is a Machine Learning Library with a neat user interface [33]. This is an open source library developed by University of Waikato, New Zealand.

In WEKA environment, all of the applicable algorithms are implemented for a selected data set ($B = 10$, r_{zc} and a_{sds} included). The ones with the highest performance are ranked which are Logistic Model Trees, Simple Logistic, Logistic, Random Forest, Functional Tree, Multiple Perceptron, LAD Tree, Nearest Neighbor,DTNB, Bayes Net, Random Tree and Simple CART. At the same time, the times for building the models are recorded. The same procedure is repeated for $P = 5$ and $P = 8$ and the algorithms that yield consistently good results in the shortest time are selected. During the very first trials with WEKA, the speed of the Naive Bayes algorithm is found outstanding although its performance is lower compared to the algorithms listed. Among the high performance algorithms, Functional Tree and Simple Logistic algorithms are selected. Simple Logistic algorithm is not as fast as Functinal Tree algorithm, however, it yields much better results for $P = 8$ case, this is why this algorithm is selected.

4.1.1 VECTOR QUANTIZATION

Vector Quantizer Algorithm (VQ), compared to more recent algorithms, is a relatively simple method however it is rather easy to implement and to visualize in two or three dimensional cases. In this work, it is also shown that, for the sound data derived from the footstep sounds of a dexterous robot body, it is very effective as a classifier, up to four classes.

4.1.1.1 CODEBOOK VECTORS

In VQ algorithm, a larger set of feature vectors in multi dimensional feature space are boiled down to a fewer set. These new set of features obtained from multiple feature vectors have the same size as all feature vectors and if the larger feature set is considered as being composed of clusters, these new feature sets represent the overall cloud for each class. These new sets of vectors are called *Codebook Vectors* and this approach is in fact originated from the needs for lossy compression of data in signal processing area [5]. This concept is illustrated in three dimensional feature vectors in figure 4.2.

These clusters are formed according to *k-means algorithm*, however, they can be formed by various different algorithms such as *Expectation Maximization* or *Farthest First* algorithms [33].

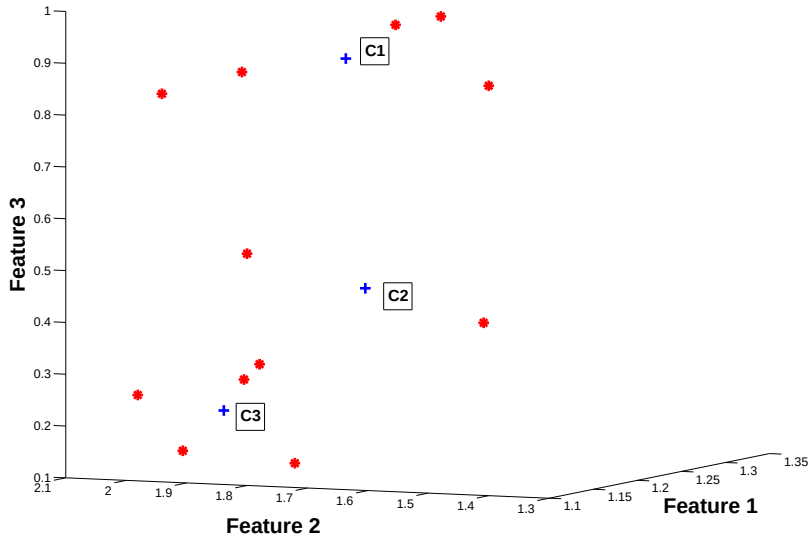


Figure 4.1: A cluster of data expressed with three features and three codebook vectors are calculated and shown in the figure as C1, C2 and C3 in three dimensions.

1.Initialize with the values of M, P_c, $\mu_1^j, \mu_2^j, \cdots, \mu_{P_c}^j$ for the jth class.
2.Classify M samples, with considering the nearest μ_i^j
3.Re-calculate μ_i^j
4.Iterate step 3 until there is no change in μ_i^j.
5.Return $\mu_1^j, \mu_2^j, \cdots, \mu_{P_c}^j$.

Figure 4.2: The k-means clustering algorithm - Adapted from [19]

4.1.1.2 VQ AS A SUPERVISED LEARNING METHOD

VQ is a frequently used classifier in machine learning as both a *supervised learning method* where the category of each pattern is provided and a model for each class in constructed so that the new classes are categorized based on this trained *model* and an *unsupervised learning method* where there is no such assignment and learning is more natural [19]. In the former case, the codebook vectors $\mu_1^j, \mu_2^j, \cdots, \mu_{P_c}^j$ formed from the *teaching classes* are collected together to form the models for each class, like Φ_j for the jth class. The new data set with no class assignment (named as *testing data set* in literature [19]) is then assigned to any of these models Φ_j, with respect to a certain measure like smallest Euclidean distance.

4.1.2 NAIVE BAYES ALGORITHM

Naive Bayes algorithm is simply the application of *Bayesian Decision Rule* while deciding the category of the patterns. This algorithm considers the prior data and for future decisions, makes use of this prior knowledge in a probabilistic manner. For a binary classification problem of classes l_j where $j = 1, 2$, prior probability stated as $Y(l_j)$ for the jth class and the conditional density expressed as $y(v|l_j)$, the *Bayes formula* is stated as

$$Y(l_j|v) = \frac{y(v|l_j)P(l_j)}{y(v)}, \tag{4.1}$$

$$y(v) = \sum_{j=1}^{2} Y(l_j|v)Y(l_j). \tag{4.2}$$

In here, the feature value is v and its probability to belong to class $j = 1$ or $j = 2$ can be estimated with this expression with making use of prior knowledge related to the effect of v on classification [19]. The Bayesian Decision Rule is then

Decide l_1, if $Y(l_1|v) > Y(l_2|v)$; otherwise decide l_2 [19].

In WEKA, Naive Bayes Classifier is selected to be implemented with a Normal Distribution estimator.

4.1.3 DECISION TREE ALGORITHMS IN GENERAL

In pattern classification literature, there is a wide branch of algorithms called *decision tree algorithms* that handle the data in a hierarchical way with forming elementary and leaf nodes and branches depending on the feature values [19]. This category of classification algorithms is widely accepted and used and with mathematically enhanced methods (one such method is explained in the following paragraphs), they are proven to be very powerful. In each decision tree, there is a root node that is placed on top of every other node. From this root node, there is a branching made with a certain set of rules. In the end, branches reach the leaf nodes and with following all of the nodes in a decision tree, an assignment to each of the patterns is completed. In other words, the output of a tree algorithm is a leaf node which assigns each pattern to a class [19].

4.1.3.1 MULTIVARIATE TREES

In the introductory part of this subsection, the definition of a decision tree is given and general properties of a decision tree is explained. In the basic decision tree approach, each decision node has to follow a unique branch and the other branches are discarded and can no longer have effects on the decision of the class of that specific pattern. Such trees are called *Univariate Decision Trees*. Although, being simpler to visualize and implement, this distinct choice in univariate tree decision is found to be problematic in many of the cases since many other probabilities are discarded so quickly. Especially in cases where the distribution of the informative and discriminative features in F dimensional feature space can not be exactly parallel to the measured data that forms patterns to be classified [19]. In such cases, there are trees called *Multivariate Decision Trees* that are formed with delicate mathematical rules and they consider the effect of multiple branches on each decision node.

4.1.3.2 THE LOGITBOOST ALGORITHM

In both statistics and pattern classification literature, there is a concept called *Maximum Likelihood* (ML). For a normal distribution of probability densities for $y(v|l_j)$, it can be safely assumed that the normal density is with mean μ_i, even though the exact values related to this density is not known. The Maximum Likelihood method maximizes the probability of getting

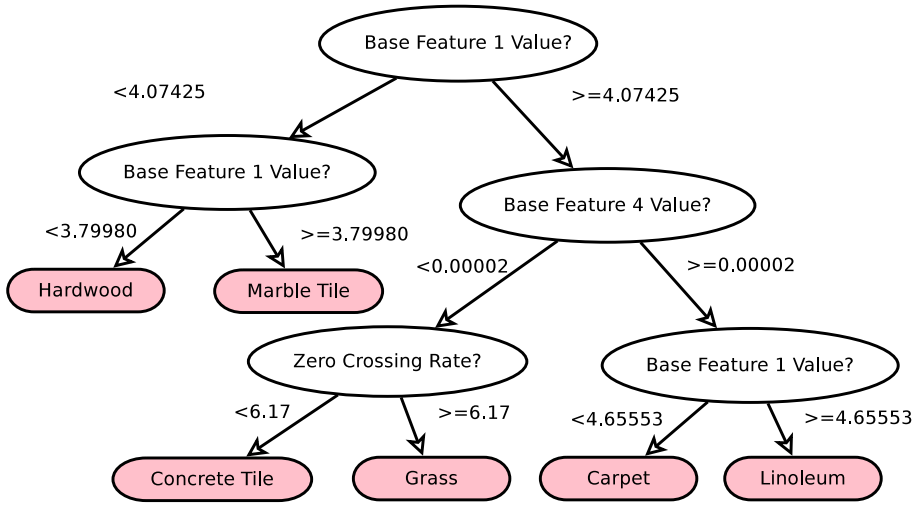


Figure 4.3: The univariate tree generated by Simple CART algorithm for P=1, $E_b = 8$. This simpler type of classification does not yield satisfactory results for out problem.

these samples that are observed in reality [19].

An improved method for classification is suggested with the name called *linear logistic regression*. The fitting procedure in this type of regression is based upon finding the maximum likelihood estimates for parameter β_j for class j, if a regression problem is given in as

$$f(v) = \beta v \tag{4.3}$$

where v is an input vector of features. For the linear logistic regression, the posterior probabilities of J classes are estimated. By implementing LogitBoost Algorithm, these posterior probabilities can be estimated [37].

Logitboost algorithm is given in Figure 4.4. This algorithm finds the maximum likelihood linear logistic model by forward stage-wise fitting of $F_j = \sum_m f_{mj}(v)$ where f_{mj} is stated as the arbitrary functions of the input variables that are fit by least squares regression [37]. This procedure is continued until convergence is obtained at iteration Q_i. In here, y^*_{ij} is given as the observed class membership probability taking value of 1 when v_i is of class j and 0 for .

1.Estimate the weights:

$h_i j = 1/r, i = 1, 2, \ldots, r\, j = 1, 2, \ldots, J, F_j(v) = 0$

$y_j = (v) = 1/J, \forall_J$

2.Repeat step (1) for $q = 1, 2, \ldots, Q_i$

a)Repeat for $j = 1, 2, \ldots, J$

i)For the jth class, calculate working responses and weights:

$$z_{ij} = \frac{y^*_{ij} - y_i(v_i)}{y_j(v_i)(1 - y_j(v_i))}$$

$$h_{ij} = y_j(v_i)(1 - y_j(v_i))$$

ii)Using weights $h_i j$, fit a least squares regression of $z_i j$ to v_i.

b)Set $f_{mj}(v) \leftarrow \frac{J-1}{J}(f_{mj}(v) - \frac{1}{J}\sum_{c=1}^{J} f_{mc}(v)), F_j(v) \leftarrow F_j(v) + f_{mj}(v)$

c)Update $y_j(v) = \frac{e^{F_j(v)}}{\sum_{c=1}^{J} e^{F_j(v)}}$

3.Output is $\arg\max_j F_j(v)$

Figure 4.4: Logitboost Algorithm applied to a J class problem - Adapted from [37]

Table 4.1: Parameters of FT Algorithm in WEKA

Binary Split (Conversion to Binary)	No
Error On Probabilities (Minimize RMS error (Selected) / Misclassification Error (Not Selected)	Not Selected
Minimum Number of Instances for Splitting in a Node:	15
Model Type:	FT
Number of Boosting Iterations:	15
Weight Trim Beta:	No

4.1.3.3 FUNCTIONAL TREES

When a better performance algorithm is sought, various algorithms are applied to the selected sets of the main experiment set in WEKA environment. Among these algorithms, Functional Tree Algorithm, is found to be both time saving and more accurate compared to 12 other high performance algorithms that are implemented.

The *Functional Tree Algorithm* (FT), combines multiple univariate trees with linear functions. As the decision tree gets larger, there are multivariate nodes are created and this tree is pruned with keeping *functional* leaves within the tree. This approach is stated as the first one to implement functional nodes and functional leaves all together [38].

In WEKA implementation, a logistic regression model is combined with a decision tree for

1.If Stop-Criterion($DataSet$)
-A Leaf Node is returned with a constant value.
2.Build a model with Constructor Φ
3.For all elements of $\vec{v} \in DataSet$
-Find $\hat{v_n} = \Phi(\vec{v})$
-Update $\vec{v}$ with new attributes $\hat{v_n}$.
4.Pick from the original $\vec{v}$ and the new attributes $\hat{v_n}$ in order to maximize a selected merit-function.
5.For all i of $DataSet$
-$Tree_i$ = GrowTree($DataSet_i, Constructor$)
6.Based on the selected attribute, $Tree$ is returned as a decision node that contains model Φ and descendants $Tree_i$.

Figure 4.5: Pseudo Code of the GrowTree Function of the Functional Tree Algorithm for inputs $DataSet, Constructor$- Adapted from [38]

finding maximum likelihood estimates with using LogitBoost algorithm [37]. Logistic regression as explained above is a linear statistical model that includes fitting the data set to a probabilistic function which usually yields a dichotomous output [39]. In case of LogitBoost algorithm, a linear logistic regression function is used to model the posterior class probabilities. In table 4.1, all of the WEKA parameters are given for classifications.

4.1.4 SIMPLE LOGISTIC

In simple logistic classifier, there is no decision tree formed but LogitBoost algorithm is combined with simple regression functions which are used as base learners. A base learner or a weak learner is like a classifier which is better than chance and includes only one node of decision [19]. The logistic model is to be fit using linear regression functions as base learners. Some parameters for this algorithm like maximum number of boosting operations are common with functional tree classifier, since they both make use of LogitBoost algorithm. In functional tree algorithm, stopping criteria is to reach the leaf node whereas in the case of simple logistic algorithm, heuristic stop is used which in fact fastens the procedure.

4.2 CROSS VALIDATION

Regardless of the choice of classification algorithm, there is a certain base rule in selection of patterns that is the testing data should not be used in model training. Otherwise the result would clearly be biased in favor of the correct decision. The token for testing is put into the algorithm database for the first time and decision is made only for newly introduced tokens. For a larger and statistically more reliable data set with larger number of patterns used in model training, *cross-validation* technique is preferred. In cross validation, the tokens excluding the test token are can be fed into classifier to form the model and when all of these tokens are fed, the technique is called *leave one out cross-validation* [19]. For cross validation, the number of excluding tokens are called *folds* and the selection becomes leave one out cross validation when the number of folds are selected as M, the number of tokens for each class. In this case, the model of the class that the selected token belong to will have $M-1$ tokens obviously since the test data cannot be fed into classifier model.

In cross validation, the class models are trained once again for each token. This makes this method to be more computationally complex. However, in this study, this approach is used in all classifiers, compared to separate training and test sets since this methods always providers more statistically reliable data with more number of tokens put into models. However, the results may be rather optimistic since, the training set and the testing set is obtained from the same data set.

4.3 THE EXECUTION PROCEDURE

So far, the patterns, the features and the methodology to be used in experimentation are all explained in a detailed manner. At this point, the implementation of all of these concepts is to be given. Since there is a lot of parameters that may affect the classification results, a planned and organized way of experimentation is followed and implemented to collected data.

4.3.1 THE PROGRAM FLOW

The MATLAB® code structure is formed with a function called `mean()` with experiment parameters such as experiment number. With the given experiment number, the program

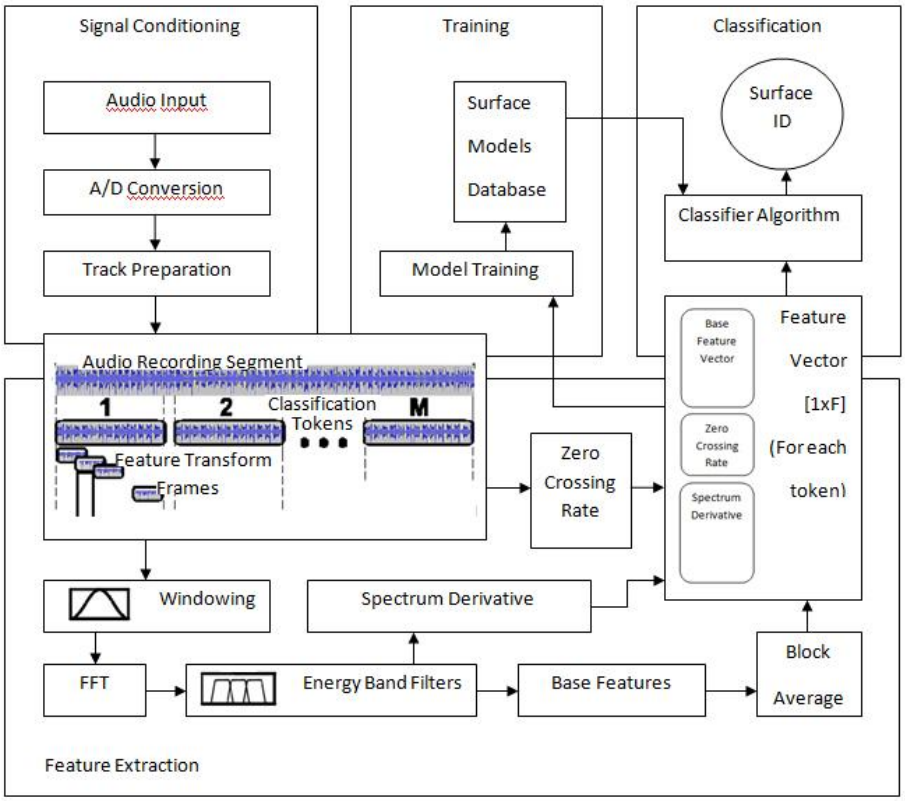


Figure 4.6: The flow of the overall procedure

browses into Experiments folder and loads the given experiment parameters. Later on, the main program opens the Surface Records folder and the time series signals are loaded one by one and saved as .mat files for the further manipulations. Depending on selection of feature extraction flag, the averaged power spectrum estimates of the signals are estimated and saved in folders. If this flag is not set, the program searches for any saved spectrum data and if there is no data found, the program throws an exception. In either of the feature extraction flag inputs, the program loads previously saved data and forms feature matrices with the given properties and records these matrices in a format that WEKA can load. For implementation of the VQ classifier, there is another parameter. When this parameter is selected, k-means algorithm is used with leave one out cross validation explained in 4.2.

The overall procedure is explained with a chart given in 4.6. In this chart, all the procedure starting from the signal preprocessing to the end of classification where an ID is assigned for each surface is shown. The procedure is the same for any selected classifier algorithm since this is supervised type of learning.

Table 4.2: Experiment Sets Table - In both of these experiment sets, Hanning window is preferred

Experiment Set - E_s	FFT Length	Window Size	Window Overlap	Token Size
1	1024	884	50%	20
2	1024	884	50%	25
3	256	128	50%	25
4	256	128	50%	50

Table 4.3: Experiment Numbers Table- In all of these experiments, the main data set for recordings is used. The classes for $C = 6$ cases are (C, L, R, H, A, G) and in other cases, the classes added or removed are denoted with + and - signs.

Experiment # - E_n	E_b	r_{zc}	ΔE_b	E_s	Classes	P
1	5-15	Yes	Yes	1	C,L,S	1
2	5-15	No	No	1	C,L,S	1
3	5-15	Yes	Yes	2	C,L,S	1
4	5-15	No	No	2	C,L,S	1
5	5-15	Yes	Yes	3	C,L,R,A	1,5,8
6	5-15	Yes	No	3	C,L,R,A	1,5,8
7	5-15	No	No	3	C,L,R,A	1,5,8
8	1-15	Yes	Yes	3	C,L,R,H,A,G	1,5,8
9	1-15	Yes	Yes	4	C,L,R,H,A,G	1,5,8
10	1-15	No	No	3	C,L,R,H,A,G	1,5,8
11	1-15	Yes	No	3	C,L,R,H,A,G	1,5,8
12	1-15	Yes	Yes	3	C,R,H,A,G	5
13	1-15	Yes	Yes	3	C,L,R,H,O,A,G	5

4.3.2 THE EXPERIMENT PLANNING

There can be a large number of parameters that would surely affect the results, no matter what minor or major the changes are. These parameters are mainly the selection of patterns, the features and FFT parameters and the classifier selection and classifier settings. Considering all these, the token size is taken as the first identifier and two type of experiments are formed with fixed FFT parameters and upon these selection, a controlled set of experiments are implemented. This list of experiment sets E_s are given in 4.2. In these sets, the main variables are the features to be added to the feature matrix and every experiment number E_n has an experiment set. The sets of experiments are given in 4.2.

Table 4.4: A Sample Confusion Matrix for $E_n = 3$, $B = 8$ and $P_c = 3$

	C	L	S
C	24	0	1
L	0	22	3
S	0	1	24

4.4 ANALYSIS OF CLASSIFICATION PERFORMANCE

In order to select and implement the best set of parameters, features and classification algorithms in further applications, there should be a performance criteria defined. In this study, the two main considerations in this selection have been the accuracy of classification with respect to certain metrics and computational complexity of the procedure implemented. There should be a metric such that it should yield the overall performance as well as the performances of classification per class. In the following subsections, the performance metric is explained in detail.

4.4.1 CONFUSION MATRICES

Confusion matrices are used quite frequently in expressing the performance of classifiers. A confusion matrix has C rows and columns where C is the number of classes in a classification problem. For a confusion matrix $\mathbf{A}$, the rows (or columns) stand for the true classes and columns (or rows) stand for the predicted classes. In this assignment, let $i = 1, 2, \ldots, C$ denote the row index and $j = 1, 2, \ldots, C$ denote the column index; the diagonal elements a_{jj} correspond to the correct guesses whereas the off-diagonal elements where $i \neq j$ correspond to incorrect guesses [40]. WEKA machine learning software uses this representation for performance measure as well.

A couple of sample confusion matrices are given in 4.4 and 4.5. Table 4.4 is for the preliminary experiment set and the table 4.5 is for the main experiment set.

Table 4.5: A Sample Confusion Matrix for Experiment, $E_n = 8$, $B = 8$ and $P = 1$

	C	L	R	H	A	G
C	17	2	1	0	1	4
L	6	42	0	0	1	1
R	0	0	25	0	0	0
H	0	0	4	21	0	0
A	1	1	0	0	23	0
G	3	2	1	0	2	18

4.4.2 OVERALL SUCCESS RATE

When all of the diagonal elements a_{jj} are summed, a measure on overall performance of a classifier can be defined. In

$$S_A = \frac{\sum a_{jj}}{\sum a_{ij} + \sum a_{jj}}.100(\%) \tag{4.4}$$

where the *success rate* of confusion matrix A is represented as S_A in percents. Since this measure is scalar, it can be an input to various discussions and plots of performance evaluation. For the confusion matrices given in 4.4 and 4.5 the success rates are 93.33 and 83.43 respectively.

4.4.3 SUCCESS RATE PER CLASS

Hence the confusion matrix yields detailed information for misclassification per each class, performance per class can also be expressed. Derived from the confusion matrix A, success rate per class is given as

$$S_j = \frac{a_{jj}}{a_{ij} + a_{jj}}.100(\%) \tag{4.5}$$

where j is the class whose performance is questioned. To illustrate, in 4.5, the success rate for G is 72 %, L is 84 % and R is 100 %.

4.5 CONSIDERATION OF COMPUTATION TIME

This work is intended to be used in a real life robotic application. As explained in 2 in detail, in order to be a useful information, the decision on the surface type should be as fast as possible, to contribute in the gait parameters in the dynamic environment described. For this reason, computational complexity of the overall process is very important in applicability of this property of the robot.

It is mostly theoretical to calculate the overall time spent on a computation since the computers used do not operate in real time and they are not totally deterministic. There is, however, a general estimate to be formed by averaging the time of Q runs for the performance. In this study, the computational complexity evaluations are based on such averaged benchmark values and unless otherwise stated, the number of runs is selected as $Q = 10$ and the time is averaged for 10 runs.

CHAPTER 5

EXPERIMENTAL RESULTS AND DISCUSSION

In this chapter, the experimental results of this work are presented. Although there are hints inserted in the overall text on these results, they are not supported by numerical values up to this point. In the following sections, following the experimental schemes that are determined carefully, the demonstration of success of classifications for the different experiment sets are done in a very detailed manner. There are more scans among the given experiment sets however, the number of all of these analyses is too much, therefore, a relatively reduced number of experiment results are presented in this thesis. In the graphic or table presentations, the general trend is preferred to be shown with the best representative cases and the rest of the similar results are mentioned in the text. However, the unexpected situations that have been observed are necessarily presented and discussed.

5.1 EARLIER VECTOR QUANTIZER WORK ON PRELIMINARY DATA SET

The preliminary data set is explained in section 2.2. The very first analyses are conducted in this data set with different parameters of VQ classifier like number of codebook vector and the number of base features and the addition of features r_{zc} or a_{sds}. Between experiment sets 1-2 and 3-4, the only difference is the addition of these features to base features. Between these two sets, the token size varies. The tables 5.1 and 5.2 show that the addition of these features improve the performance significantly from 75-85% to above 90%. It is expected that with a shorter token size (With larger token size M) the success rate should drop since the information available is a bit less. However, for the longer duration tokens, the success rate has been slightly lower. With the longer duration records, more noise is present which is

Table 5.1: Some outstanding results of varying number of average spectrum features - B, for fixed $P_c = 2$

Experiment Number - E_n	Variable = B	Performance
2	12	75.00 %
2	10	76.67 %
1	12	93.33 %
1	10	93.33%
4	12	85.00 %
4	10	74.67 %
3	12	98.33 %
3	10	94.67 %

Table 5.2: Some outstanding results of varying number of average spectrum features - B, for fixed $P_c = 3$

Experiment Number - E_n	Variable = B	Performance
2	12	78.33 %
2	10	84.00 %
1	12	98.33 %
1	10	93.33%
4	12	77.33 %
4	10	84.00 %
3	12	93.33 %
3	10	98.33 %

believed to shadow better decisions a little more than the experiment set with shorter token size. In other words, is more likely to be a coincidence, the general trend is expected to be the otherwise and this point is demonstrated in the experiments with the main experiment set.

As stated in 4, VQ algorithm applied in MATLAB® environment is used while doing the preliminary analyses. Instead of the built in`kmeans()` command, an implementation of the algorithm with the help of web sources [41], is preferred for learning and implementing a specific classifier. In this application, the experiment parameters that are specific for the classifier are listed as the number of k-means iterations to find the vector centroids $\mu_1^j, \mu_2^j, \cdots, \mu_{P_c}^j$, the initial number to start iteration and the number of codebook vectors for each class representation. Among these, the number of iterations is in fact in an endless loop therefore, the algorithm stops when there is no smaller distance that can be calculated. The initial number to start iterations is always selected as the first element of the feature vector.

As shown in table 5.1 and 5.2, the results of this particular set have been found quite successful therefore the applicability of this study is proven and a more broad set of records including different speeds are made.

5.2 VQ IMPLEMENTATION TO FOUR CLASSES OF MAIN DATA SET

The main set is recorded and first examined with VQ. When all of the six or seven classes (for $P = 5$ only) are examined, the success rate have dropped significantly (Discussed and shown in the following sections more in detail). For this reason, the number of classes is increased gradually and a set of experiments is done first with four classes. These experiments are covered in experiment sets $E_n = 5, 6, 7$. The results for these experiment sets for different speeds are all examined in detail for a better approach for classification with more number of classes.

As seen in figure 5.1, for $P_c = 10$ case, the performance of different speeds differs. Especially for $P = 8$ case, the success rate drops significantly and for $P = 5$, there is considerable amount of drop (around 8% in average) compared to the $P = 1$ case. Moreover, the success rate does not seem to be variating for different number of base features.

When the effect of the features r_{zc} and a_{sds} are examined in figure 5.2, it is seen that the improvement that is brought by these features is very significant around 20 % for $P = 1$. On the other hand. the improvement for $P = 5$ is very low and there is slight drop of performance for $P = 8$ case.

The effect of P_c is examined for different speeds. For $P = 1$ and $P = 5$ cases, the best results are taken around $P_c = 10 - 15$ and there is not much of an improvement in any of these cases. The success rate for $P = 1$ is in fact found applicable (around 80%) but for $P = 5$ case (around 70%), a performance improvement is sought. In figure 5.3, the search results for selected P_c values are given. Unfortunately, with tuning the number of vector centers, there is not much change observed.

The success rate for $P = 8$ case is not found satisfactory and there are ways for improvement is sought. In the previous paragraphs, an unexpected performance drop with additional features was mentioned. With the implementation of $E_n = 6$, the effect of this could be examined.

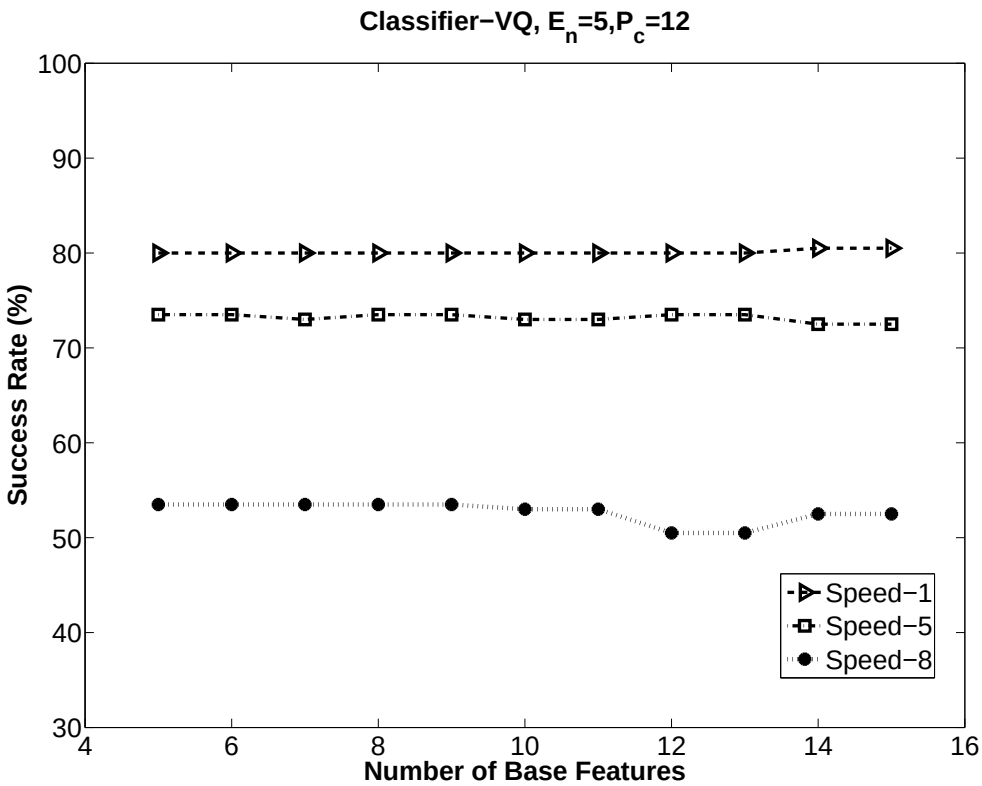


Figure 5.1: The performance of VQ for different speeds with features r_{zc} and a_{sds} included

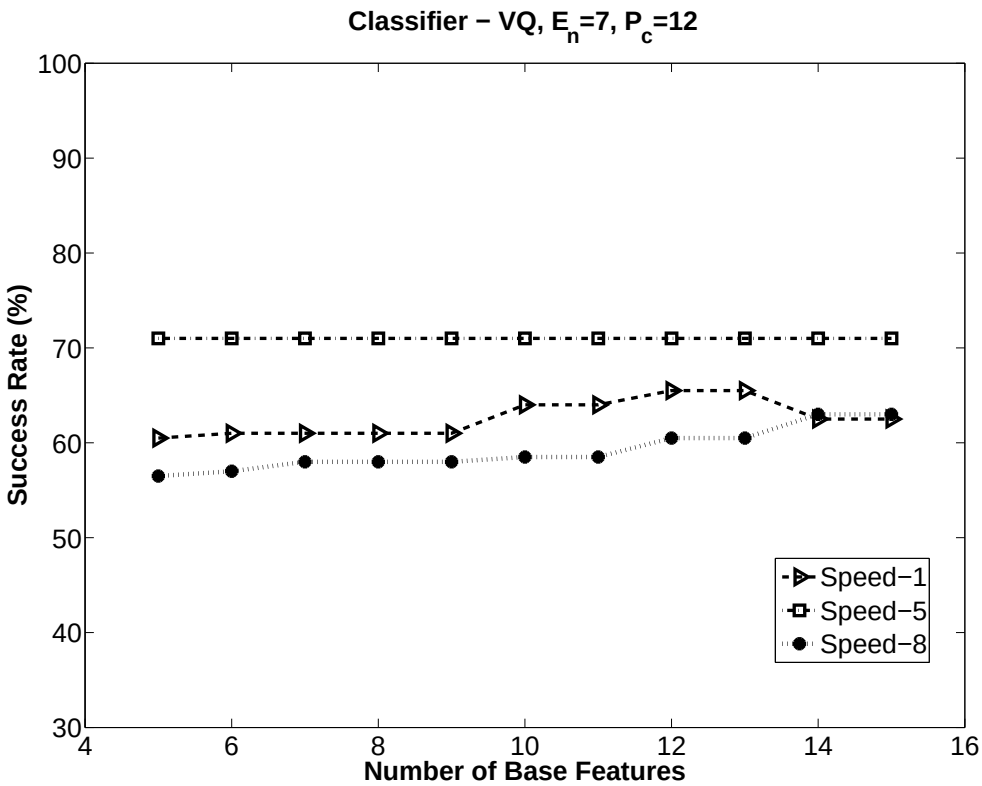


Figure 5.2: The performance of VQ for different speeds without features r_{zc} and a_{sds}

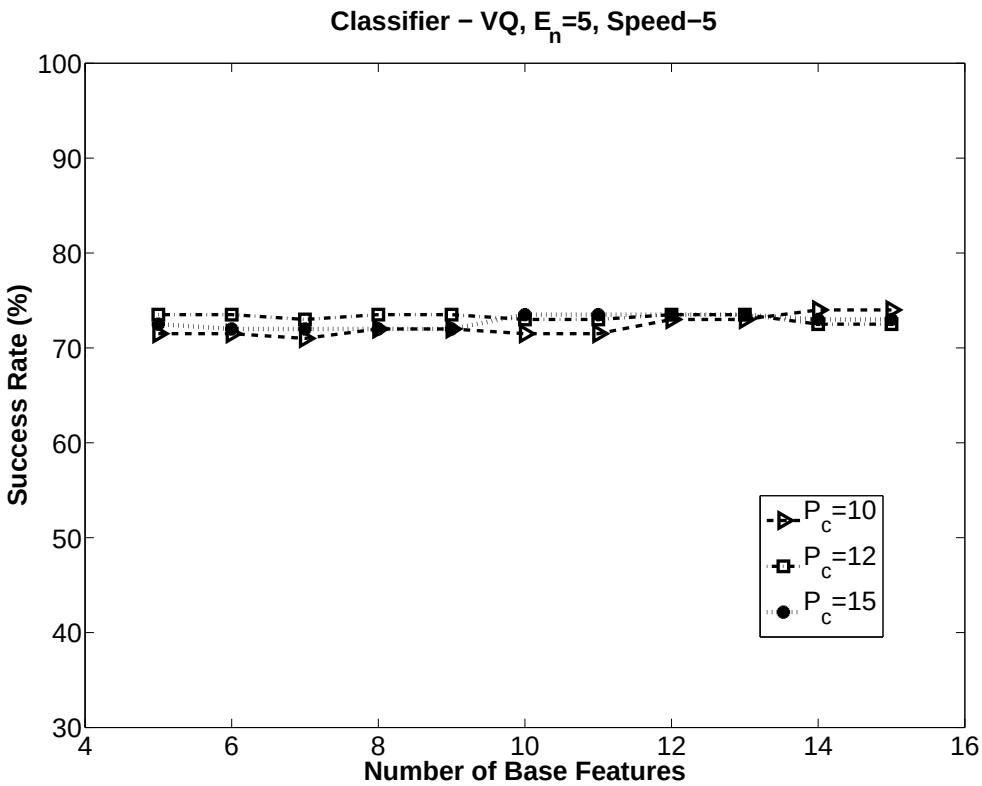


Figure 5.3: Observing the performance change with different values of P_c for $P = 5$

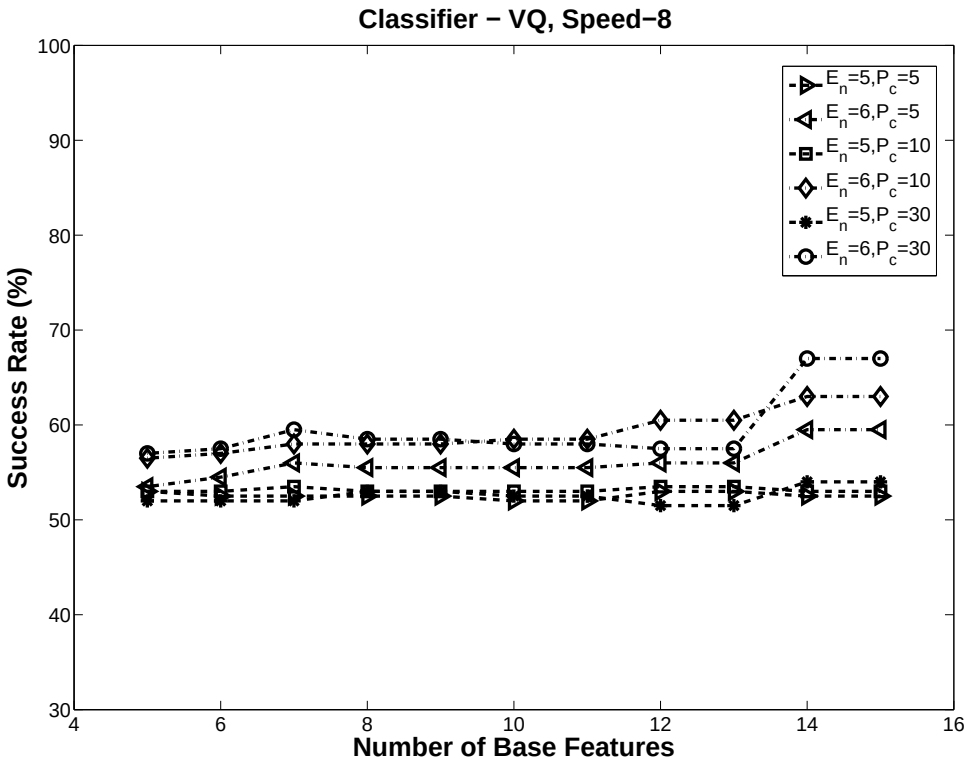


Figure 5.4: Seeking the best performance with tuning the values of P_c for $P = 8$

From the figure 5.4, it can be said that for VQ classifier for many diversifying values of P_c, the addition of feature a_{sds} makes a deteriorating effect on classification performance. Moreover, there is the expected increasing trend of success rate with increasing number of base features is finally observed with $E_n = 6$ for $P = 8$. Although there is an improvement found for $P = 8$ case which in general has very inadequate success rate around 50 %, the amount of it (approximately 15%) is not found satisfactory. This can also be stated as a reason for searching for a a different classifier, in addition to the problem with larger number of classes.

5.3 WEKA IMPLEMENTATION ON THE MAIN DATA SET

As stated in the previous section, the number of classes have increased, VQ in supervised mode did not perform quite well. Therefore, as explained in chapter 4, WEKA environment is preferred since there is a large number of classifiers available and easier to implement.

Once a classifier is selected among others after evaluating with respect to performance and time criteria stated in 4, then it can be embedded into the robot's controller PC.

After the VQ performance is seen to be dropping with five classes, before moving on to other classifiers, VQ algorithm implemented in MATLAB® environment is confirmed with WEKA's VQ. In WEKA, the supervised VQ is named as *Learning Vector Quantization Algorithm - 1* (LVQ1) and comes with WEKA 1.8 Classification Algorithms plug-in. All of the experiments conducted in MATLAB® environment are confirmed in WEKA environment and they have yielded the same results as expected.

As it is stated in previous chapter in 4, there are various algorithms implemented in WEKA environment. A sample performance of these algorithms on $E_n = 8, B = 10$ and their performance for $P = 1$ is given in figure 5.5. In this figure, there are algorithms observed with a better performance than FT and SL, however, the time passed for estimation is very high compared to FT or SL. For instance, it takes 7.5 s in average to train a model and assign a class to a model. Moreover, not every algorithm yields a results, i.e. Random Tree classifier has diverged for $P = 5$ case. For other speeds and other number of base features, this performance varies as expected, however, when the time and convergence is considered, FT and SL is seen to be much better after many number of trials.

5.3.1 TIME COMPLEXITY VS PERFORMANCE

Since there are a large number of experiments including all experiment sets, all of the experiments are not repeated for all classifiers and all feature sets. There are sample cases selected for each speed and each experiment set and sample classifications are done. In figure 5.6 the results for $E_b = 10$, r_{zc} and ΔE_b is given for different speeds, $P = 1, P = 5, P = 8$. In these graphs, SL classifier and FT classifiers are both observed to have higher performance compared to VQ and NB. In especially $P = 8$ case, SL shows higher performance, in other experiments sets with $E_b = 1, 2, \ldots, 15$ however, it should be noted that the time scale of the graphic is logarithmic. The time consumed by SL algorithm is relatively high compared to other algorithms. Yielding results around 2.5 s for each model training and classification procedure, computation by SL is not found feasible.

In WEKA environment, all of the applicable algorithms are implemented for a selected data

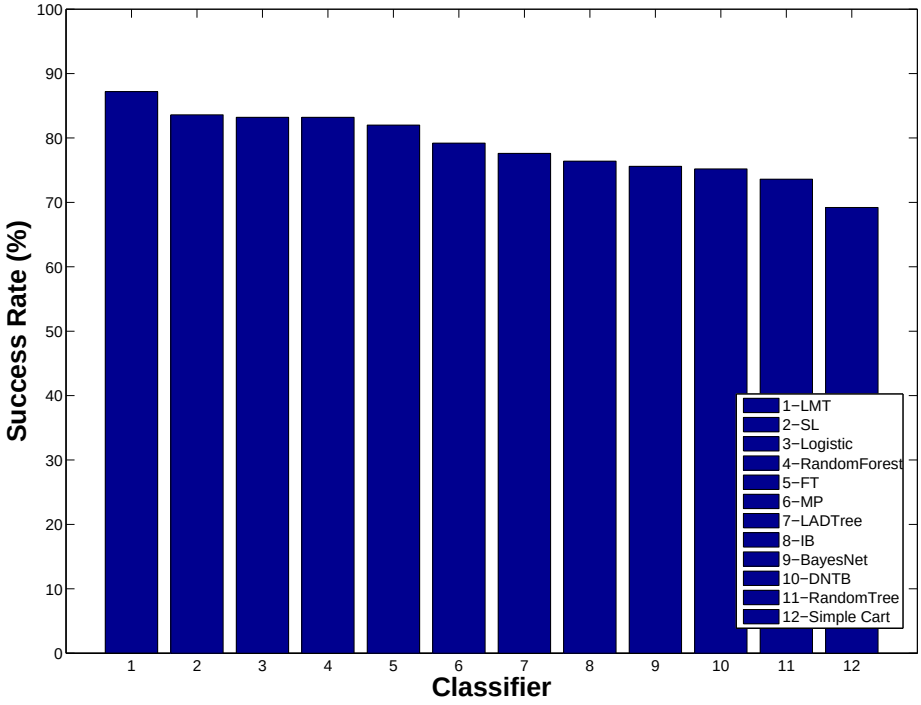


Figure 5.5: The performances of the applied algorithms in WEKA

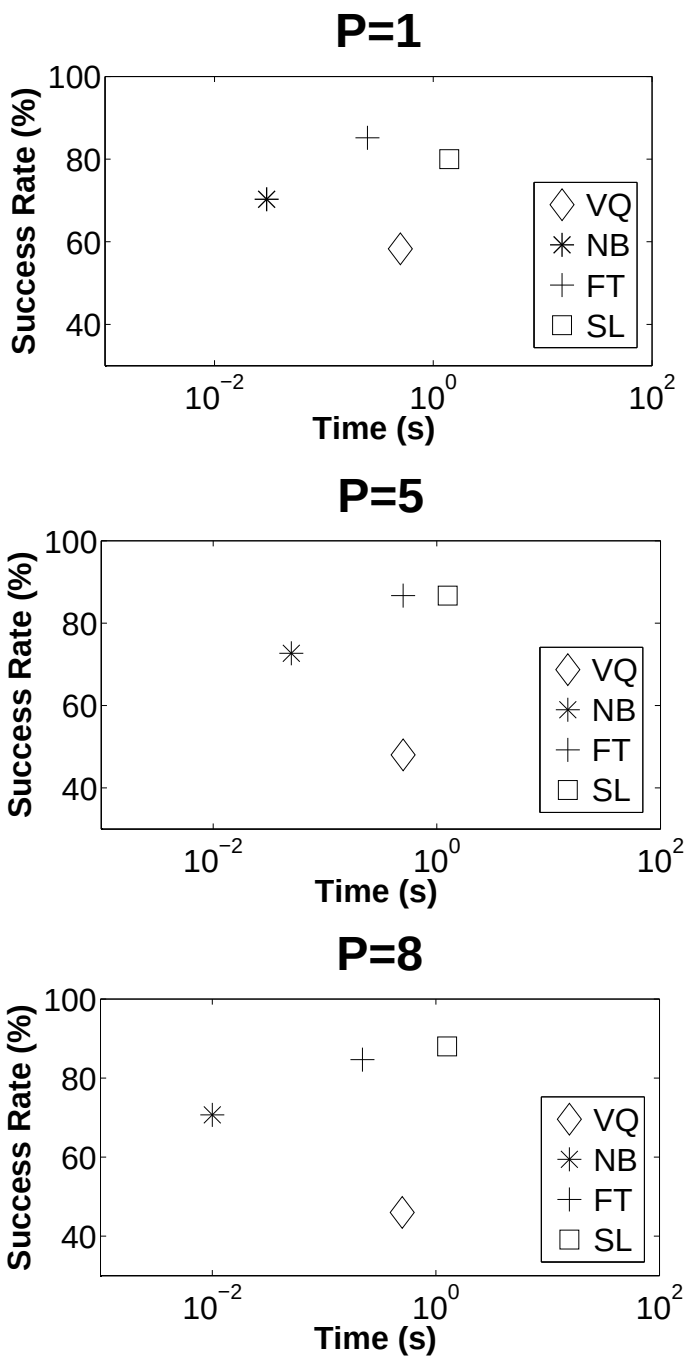


Figure 5.6: The Performances of Classification for Different Speeds, $B = 10$, $E_n = 8$ (Time axis is in log scale)

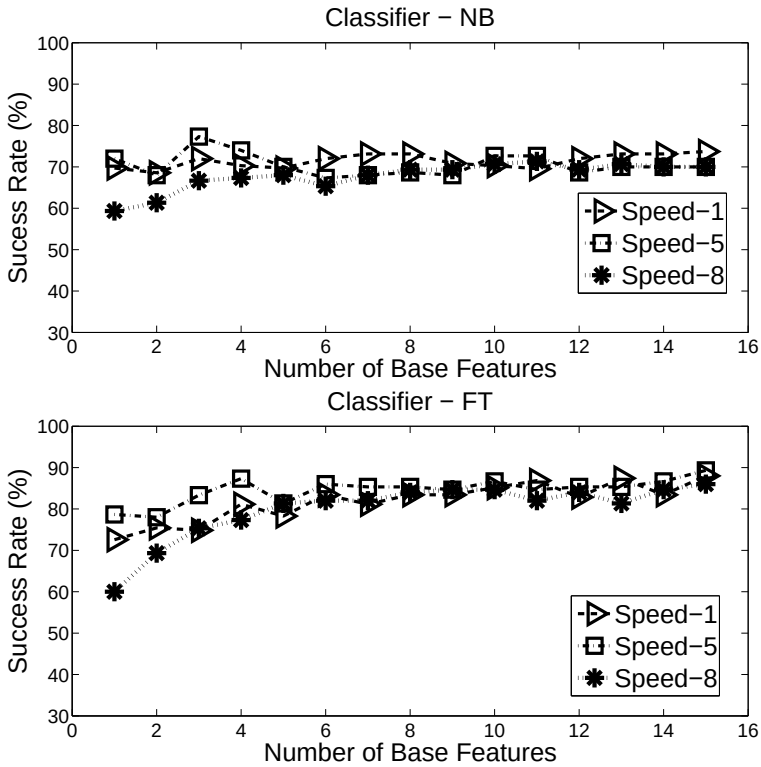


Figure 5.7: The Performances of Naive Bayes and Functional Tree Classifiers for $P = 1, P = 5, P = 8$ ($E_n = 8$)

set ($B = 10$, r_{zc} and a_{sds} included). The ones with the highest performance are ranked which are Logistic Model Trees, Simple Logistic, Logistic, Random Forest, Functional Tree, Multiple Perceptron, LAD Tree, Nearest Neighbor, DTNB, Bayes Net, Random Tree and Simple CART. At the same time, the times for building the models are recorded. The same procedure is repeated for $P = 5$ and $P = 8$ and the algorithms that yield consistently good results in the shortest time are selected. During the very first trials with WEKA, the speed of the Naive Bayes algorithm is found outstanding although its performance is lower compared to the algorithms listed. Among the high performance algorithms, Functional Tree and Simple Logistic algorithms are selected. Simple Logistic algorithm is not as fast as Functional Tree algorithm, however, it yields much better results for $P = 8$ case, this is why this algorithm is selected.

NB algorithm is seen to be quite fast compared to other algorithms. It is selected for this reason and its performance is compared with FT in the following graphics.

5.3.2 NAIVE BAYES VS FUNCTIONAL TREE ALGORITHM

For any confusion matrix A derived from any classifier, with the help of the success rate S_A stated in chapter 4 gives the most relevant information regarding the classifier performance. However, this does not necessarily mean that for a high success rate, success rate per class S_j is balanced per class. In other words, there can be some classes that are misclassified quite often and there may be classes that are classified perfectly. The results for classes are examined more in detail in this chapter, the success rate per class statements are embedded into discussions on classifiers.

The comparative evaluation between FT and NB performances is done in figure 5.7. In general, FT is seen to yield more successful results that has a clear increasing trend as the number of features increase compared to NB. In NB results, there is not a clear increasing trend. In both of these graphs, a peak is observed for $E_b = 3$ and $P = 5$. It can be commented that it is a coincidental case where very definitive features come together. Rather than focusing on local maximums in the graphs, general trend is observed and evaluated. However, it is found quite normal to find such outlying performance while scanning the parameter surfaces. For FT classifier, $P = 5$ case seems to be the most successful one and there is a varying performance of $P = 1$ and $P = 5$ for NB. For both of the classifiers, $P = 8$ case is the worst one. As stated above, SL classifier have performed on this speed higher compared to other classifiers, even to FT, however, it is not preferred since this classifier has been found slow for this application.

In terms of classes, NB performs quite well with classes R and H that can be safely stated as 88% or higher. However, there is no diagonal dominance obtained for class G at $P = 1$ case with this classifier in any of the experiment sets. This changes to class C for $P = 5$ and A for $P = 8$. This drops the overall performance with a considerable amount. G is confused with various other surfaces and this reminds of the performance drop trend of VQ. In VQ classifier, as the number of classes increased, the performance dropped considerably and especially the classification results of grass has been wrong in most of the time.

For FT classifier, success per class is rather homogeneous around 80% when $E_b > 5$, although the highest number of confusions are originated from A and G surfaces. For $E_b < 5$ the homogeneity is still kept within 10-15 % range and per class success rate is minimum 50% and diagonal dominance is not disturbed at all. The classes that are confused often do not seem

to change dramatically for different speeds for this classifier compared to NB. It is useful to note that the mostly problematic class for $P = 8$ is A for both classifiers. In speeds $P = 1$ and $P = 5$. the classification rates for R and H are higher than 90%.

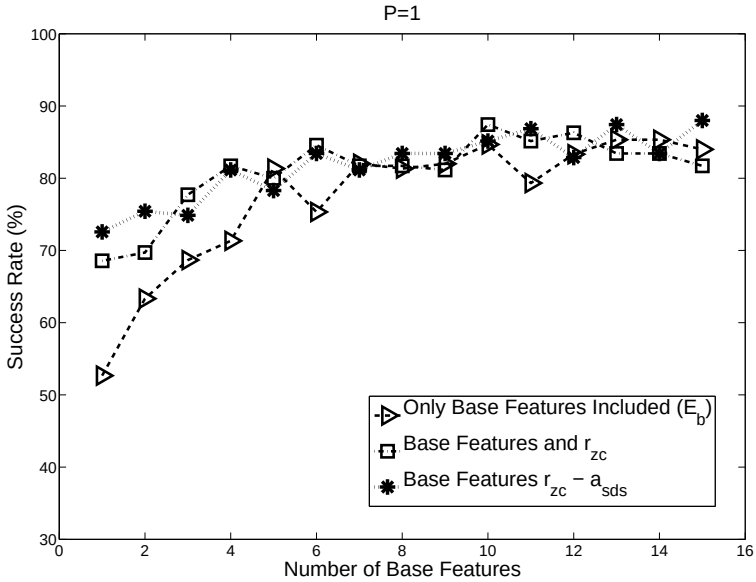
5.4 ANALYZING THE EFFECT OF THE SELECTED FEATURES

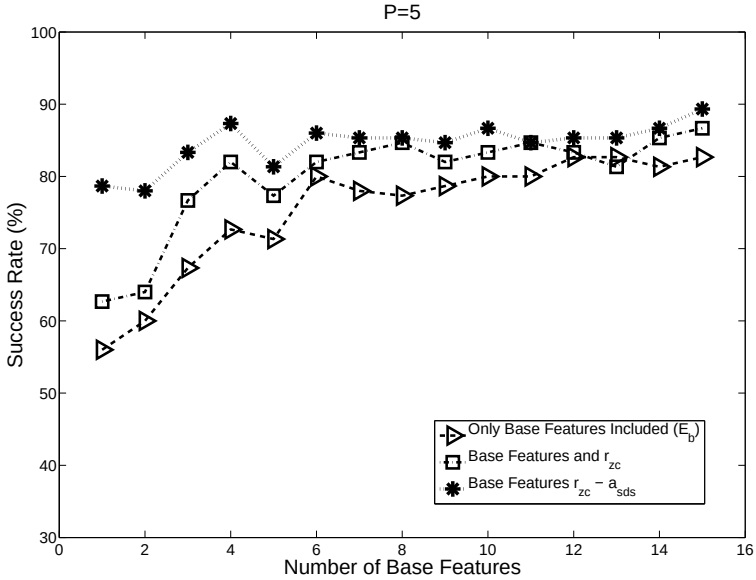
The number of base features is a critical concern for selection. In the beginning of the study it is assumed that up to a certain limit, the increase of the average spectrum sums named as base features E_b explained in 3 with its mathematical expression, is expected to result in the increase in classification performance. After some point however, there is a drop or a saturation expected. All these phenomena are observed in the graphs that sweep E_b values on x-axis and show the success rate in y-axis.

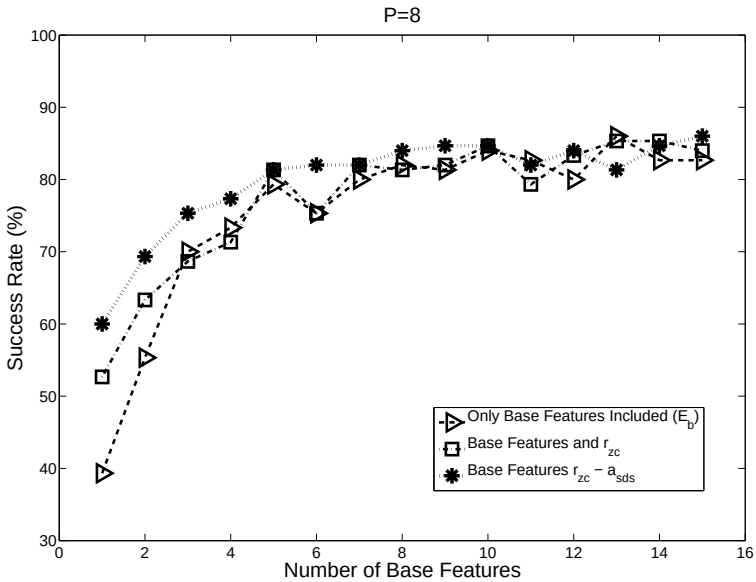
In the previous section, it is assumed that with the addition of zero crossing rate r_{zc} and sum of spectrum derivative ΔE_b to the feature vector as new features should improve the classification performance in terms of success rate although they bring additional computational complexity. Various experiment sets are formed based upon this assumption and the most suitable algorithms are decided this way. Therefore, this assumption is to be checked and verified once again with solid evidence on some common experiment sets. In the following paragraphs, the detailed analysis of the effect of addition of each new feature on performance is made. The number of base features, B is selected to be 15 at maximum due to the reasons explained in 3.4.1 in detail.

5.4.1 THE EFFECT OF NUMBER OF AVERAGE SPECTRUM FEATURES

In this work, all of the figures presented indicate the saturation or drop in success rate as the number of base features passes a certain limit. To illustrate this phenomenon, it can be stated that on figure 5.7, for the FT classifier, up to $E_b = 6$ the success rate seems to increase with a higher rate of change, excluding the $E_b = 4$ case for $P = 5$ which can be stated as an outlying performance for a relatively low number of features. The success rate is then observed to be saturating to values slightly higher than 80 % with some fluctuations for each speed. For NB classifier however, 70 % success seems to be a saturation value after $E_b = 2$ for $P = 8$ and the rest of the speeds seems not to be effected at all except the outlying case with $E_b = 4$ and


P=1
Success Rate (%)
Number of Base Features
Only Base Features Included (E_b)
Base Features and r_{zc}
Base Features $r_{zc} - a_{sds}$


P=5
Success Rate (%)
Number of Base Features
Only Base Features Included (E_b)
Base Features and r_{zc}
Base Features $r_{zc} - a_{sds}$


P=8
Success Rate (%)
Number of Base Features
Only Base Features Included (E_b)
Base Features and r_{zc}
Base Features $r_{zc} - a_{sds}$

$P = 5$ again. In figure 5.8 for FT, the increase and saturation trend is more clear since there are no additional features like $r_z c$ or ΔE_b to boost the performance for the number of base features.

As stated in chapter 3, the base features are in fact some sort of filters and their success mostly depend on how well that they can emphasize the differentiating features of the signals for different surfaces and for FT classifier, a fine performance can be derived with optimizing the number of features so that there is performance improvement and the complexity is not increased that much which can be $E_b = 7$ or $E_b = 8$ case. In the case of reduced number of base features, it is observed that C, L, A and G classes are more confused for NB, where the performance of L is improved with the increasing number of E_b.

5.4.2 THE EFFECT OF ZERO CROSSING RATE FEATURE

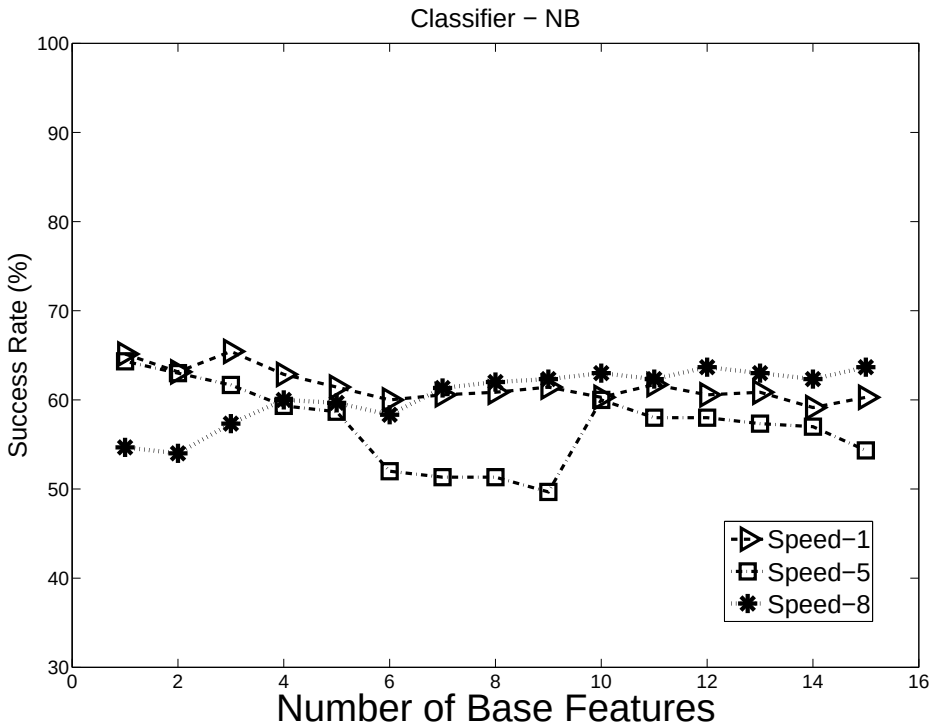
Figure 5.8 clearly shows that r_{zc} improves the performance for $P = 1$ and $P = 5$, $E_b < 5$ cases. After this point, there is relatively not much increase observed in performance; depending on speed, it is around 2-7 %. The performance boost is mostly seen with $P = 5$ case.

5.4.3 THE EFFECT OF DERIVATIVE SPECTRUM FEATURE

Similar to zero crossing rate, the addition of this feature shows its best effect on lower number of base features. After a certain point like $E_b = 5$ the performance improvement is not that high and there are even drops of 1-2% for some E_b values. In experiments $E_n = 5, 6, 7$ with VO classifier, it was seen than for $P = 8$ case, there was a significant drop. However with a more sophisticated probabilistic classification procedure, the possible detrimental effect of this feature is reduced and the cases only when this feature could be utilized contribute in overall classification. This cannot be directly proved but depending on the data and the results available, it is the best comment that could be suggested.

5.5 ANALYZING THE EFFECT OF TOKEN SIZE

For a shorter token size, the computation time is expected to drop. For this reason, the token size is reduced by half by setting parameter $M = 50$. In this set of experiments, the size of



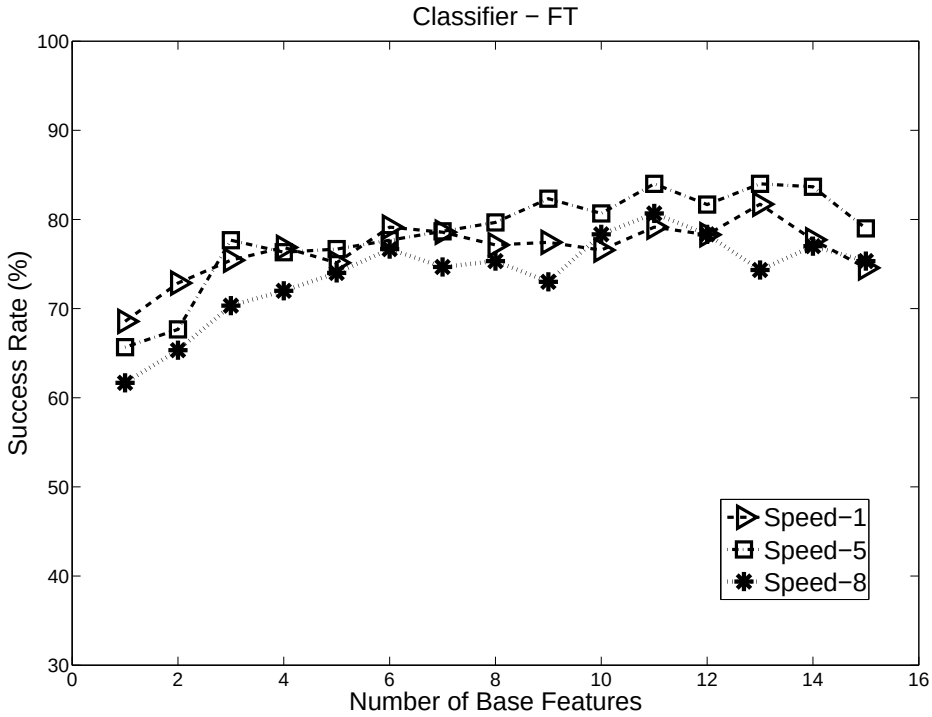


Figure 5.9: The Performances of Naive Bayes and Functional Tree Classifiers for a shorter token size, for $P = 1, P = 5, P = 8$ ($E_n = 9$)

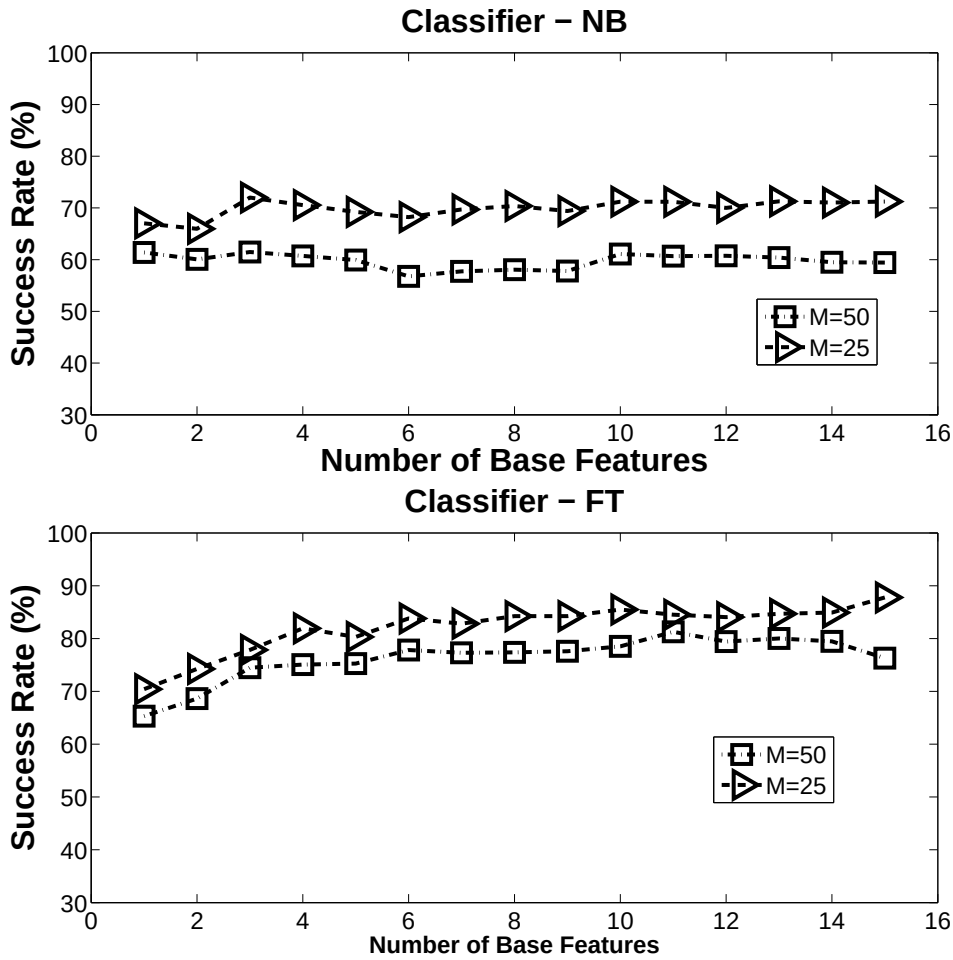


Figure 5.10: The Performance Comparison for Different Token Sizes M, for Average of Performances of $P = 1$, $P = 5$, $P = 8$

the overall sound data is the same, however, the data is divided into 50 tokens. As shown is figure 5.9, there is considerable amount of performance drop around 11-15 % especially for $P = 5$ case these values are likely to be close to 15 %. A more clear graph for comparison means is given in figure 5.10. These graphs for NB and FT are formed with averaged values for all three speed values. It is also useful to note that the aforementioned drop pattern for increased number of E_b is observed in FT graph for $M = 50$ case as well. There is only one case of improvement observed for case $E_b = 1$ and $P = 8$, however, this case is not applicable at all because $E_b = 1$ does not yield high success rates that are expect and in average there is no increase observed in any of the cases, this is found more considerable.

In terms of timing, where a benefit is sought by decreasing the token length, the estimation time is 0.62 s in average for $M = 25$ case and 0.61 s in average for $M = 50$ case. Depending on the application, this may seem to be a less improvement however there may cases where a real time decision making procedure is of concern. Therefore such an improvement could be found worth to apply in exchange of performance drop. There may be cases where accuracy around 70 % would be enough and the emphasis is put on fast decision making and there may be cases where the accuracy is important and time complexity can be a compromise. In this particular application higher accuracy is preferable, even though the off-line training time is increased considerably for multiple experiments.

5.6 ANALYSIS ON CLASSES

A new class is introduced to the system for $M = 25$. This new class O (Shown in 2.7 is a bare surface unlike the surfaces that are covered or partially covered like carpet, linoleum, grass and concrete tile with autumn leaves. Surface stiffness is expected to be higher for class H, since H surface is made of wood and not from concrete. As seen in the photographs, this surface is similar to marble tile in terms of geometrical shape and dimensions. The surface stiffness is estimated to be close to marble tile as well compared to other surfaces. Surface A is also a type of concrete tile however, there are areas covered with random soil and grass as well as the autumn leaves.

This new class is expected to lower the performance since it is quite similar to surface R and A. The results (an illustrative sample given in 5.3 in confusion matrix form) reveal that the

Table 5.3: A Sample Confusion Matrix for Experiment $E_n = 13$, $B = 10$

	C	L	R	H	O	A	G
C	18	1	0	0	2	3	1
L	1	22	0	0	0	1	1
R	0	0	18	3	4	0	0
H	0	0	0	24	1	0	0
O	0	0	6	3	16	0	0
A	1	1	0	1	0	22	0
G	0	2	0	0	0	2	21

surface is often confused with marble tile. There is a performance drop observed to be around 10% for R and slightly less for other classes and close to 10 % in general. There is a case where the success rate is the same for six class and seven class line, however, the confusion matrices of these two cases are not exactly the same. If the FT classifier is observed, it seems to yield considerably successful results around 80 % for values $E_b > 7$. For both classifiers, the success rate is increased with a considerable amount with five classes where linoleum surface is removed. As previously stated, this surface is often confused with carpet. In 2.8, the carpet surface is shown on linoleum surface. All of the carpet surfaces recorded are actually on linoleum surface. The overall performance of the classification with different number of classes is shown in figure 5.11. The success rate is observed to be very high in 5 class ($E_n = 12$) around 90 %.

5.7 THE EFFECTS OF SPECTRAL SUBTRACTION

Spectral subtraction of the motor noise is considered as an element for performance increase since the data includes a high amount of motor sound that is believed to mask the interaction sound. For this purpose, a motor model formed with 20 s of motor noise for each surface is extracted from the audio data manually and this data is controlled carefully by human ear for not containing any interaction sound. The motor model is subject to the same power spectrum derivation procedure explained in section 2. First introduced in 1979 [42], spectral subtraction concept is known for a long time in signal processing literature and it has impressive success on various applications like canceling engine noise.

Exhausting various α and γ values in 2.6, 0.5 for α and 2 for γ is observed to increase iden-

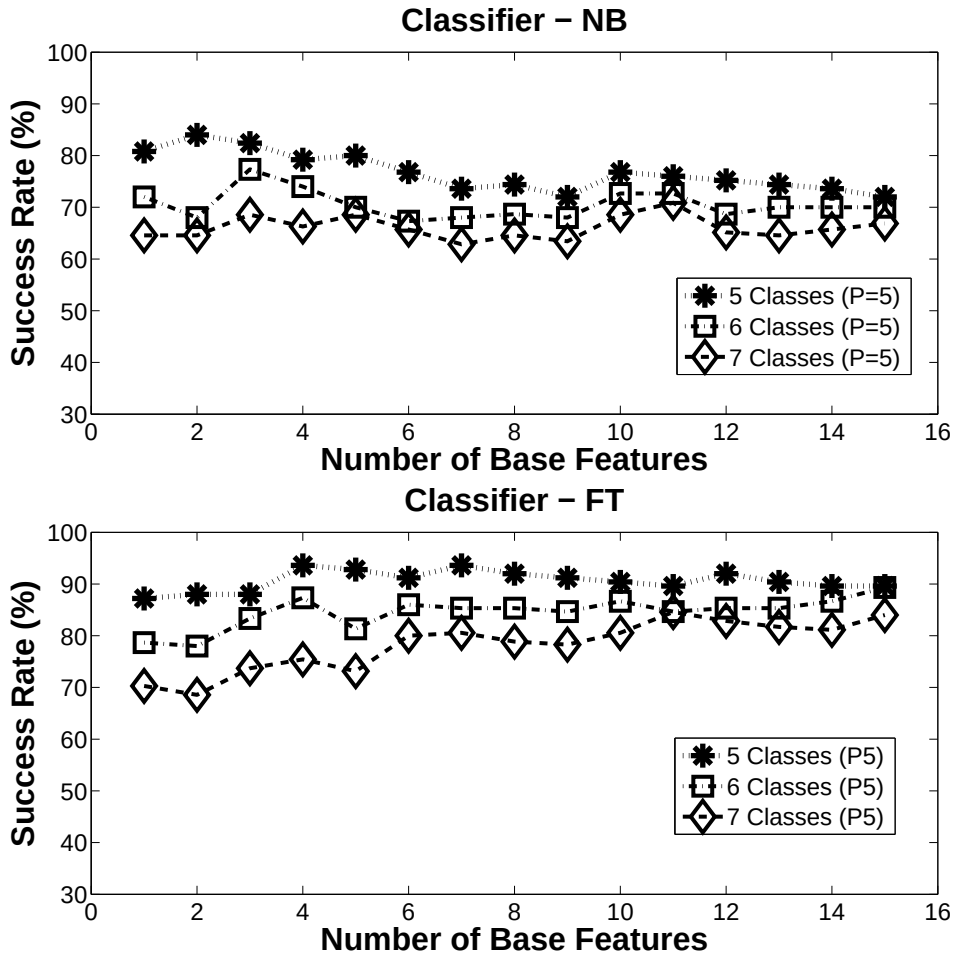


Figure 5.11: The Performance Change With 7 Classes for Classifiers NB and FT ($E_n = 12, E_n = 8$ and $E_n = 13$)

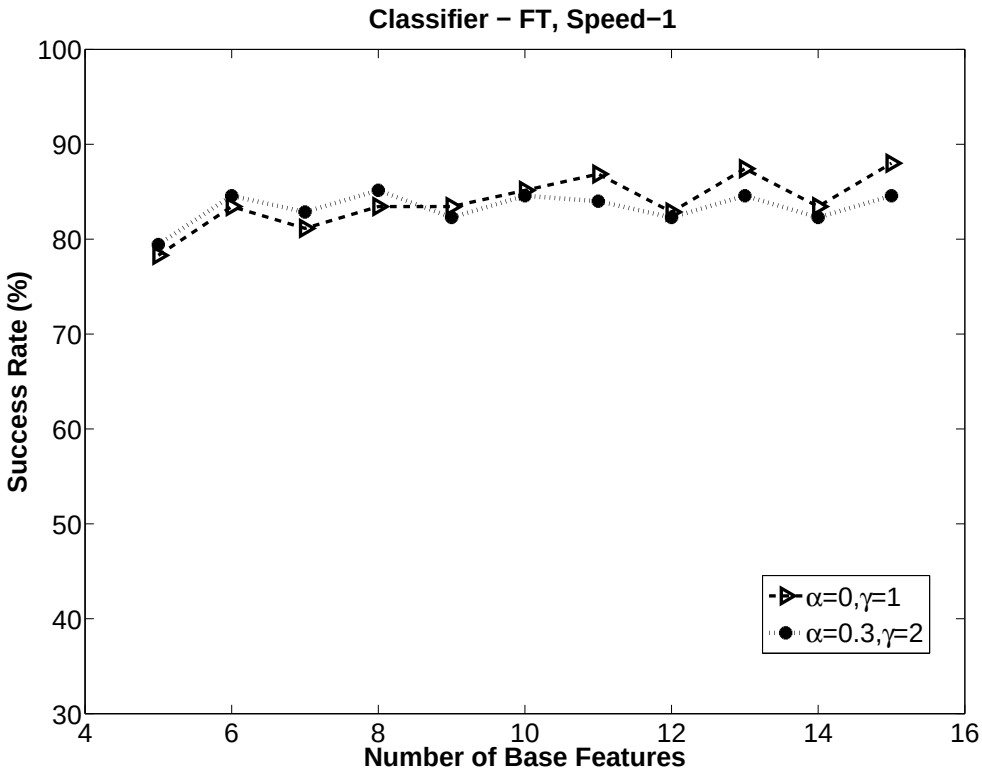


Figure 5.12: The effects of spectral subtraction for $P = 1$

tification performance for some of the cases. These cases are given in figures 5.12, 5.13 and 5.14. However, the difference is not as much as expected (around 3% at maximum). This may be due to continuation of the acoustic impact data in the signal although magnitude of the impact is to decay while the legs turn for the next impact and subtraction results in loss of this data. In other words, some useful data for classification may have been erased from the spectrum when some portion of the signal is emphasized. The best performance increase is observed for speed 8, M=25 seen in 5.14. This part of the study is not quite focused on, therefore, the methods that are applied may be insufficient and there is not many experiments conducted to reach a conclusion.

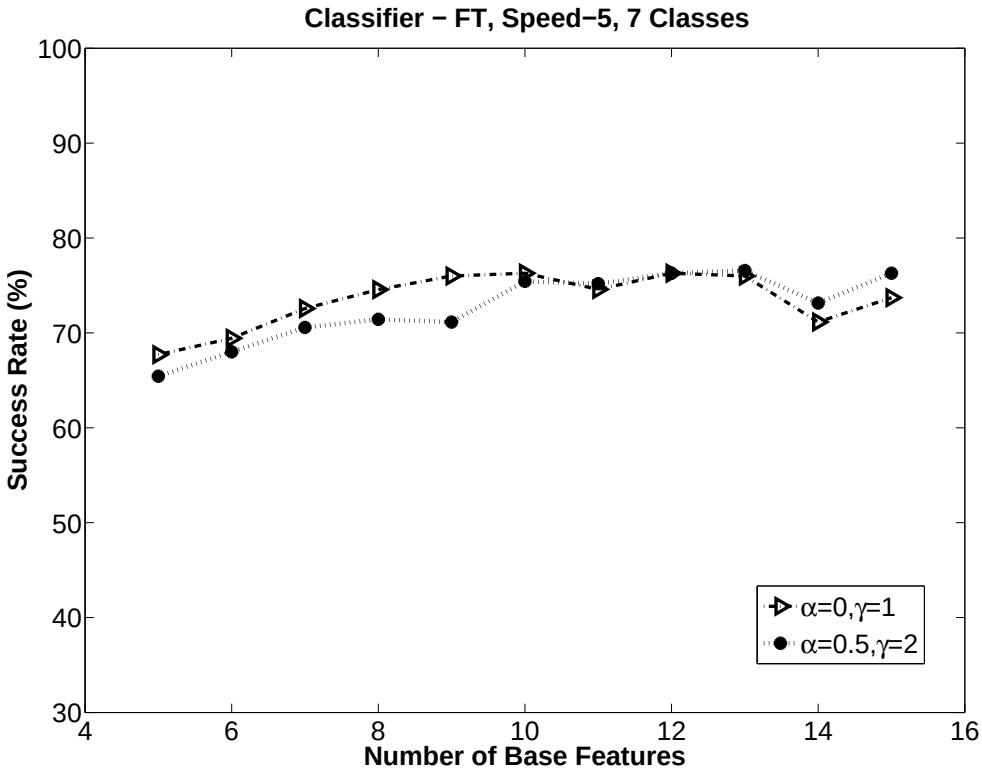


Figure 5.13: The effects of spectral subtraction for $P = 5$

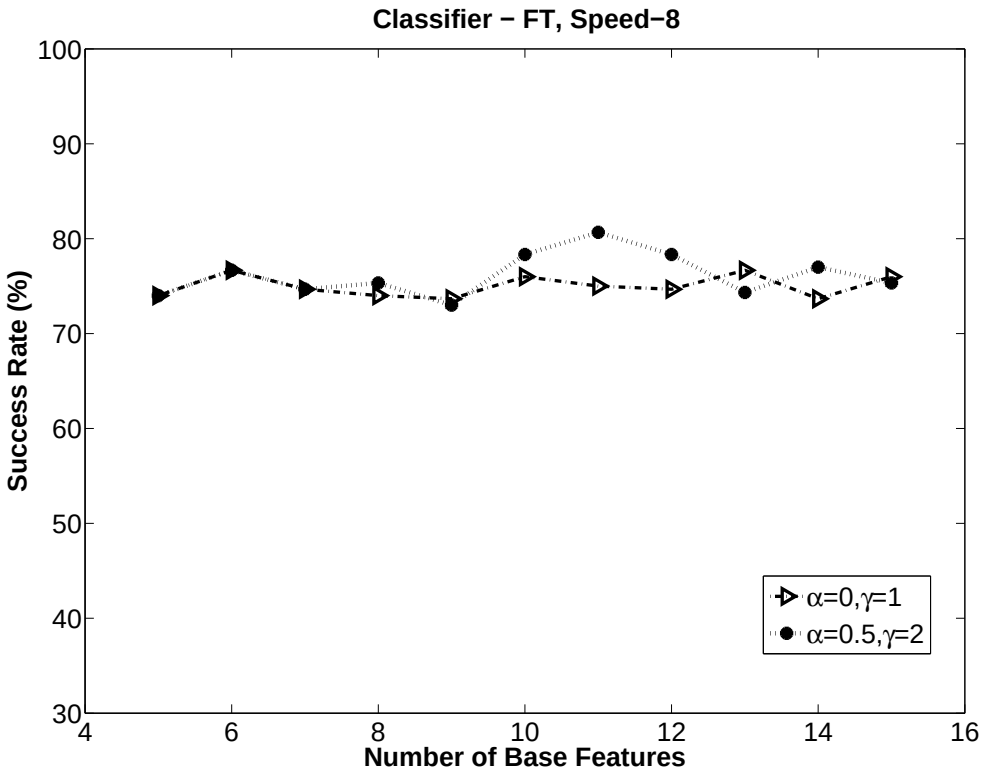


Figure 5.14: The effects of spectral subtraction for $P = 8$. The most successful results of this operation is observed for this speed.

CHAPTER 6

CONCLUSION

In this work, acoustic surface sensing and perception for a mobile robot is examined. With a legged mobile robot, there can be various sensory ways of determining the surface type and acoustic sensing is claimed to be a considerable alternative among other ways of sensing. The experimental setup, namely the SensoRHEX robot, that inspired the idea of applicability is able to travel in both indoor and outdoor surfaces and the data collected from available surfaces are examined for this purpose. With the aid of delicate feature extraction schemes and sophisticated pattern recognition algorithms, the results have yielded high success rates.

At this point it should be emphasized that when the same data was listened by miscellaneous people, the difference between most of the surfaces could not be told at all. There are various approaches tried on the recorded signals but in all of the experiments a more natural sound data is preferred. This means the ambient sound was not particularly reduced. However, the ambient data is checked for any misleading features that may effect the result of classification in a positive but wrong way since it is not related to the surface interaction sound. In other words, the data is collected carefully for not leading to any misconceptions but at the same time the random noise coming from the ambient is rather high and similar to daily hearing data of living creatures. The collected data is in fact a combination that is closely related to surface stiffness and damping values since there is an impact type interaction in the touch of each leg. Moreover, the data should also be related to surface roughness since there is ground-leg interaction during the rolling motion of the C shaped legs. There are also elements expected from the sound wave reflections that are related to the acoustic properties of the environment. In this study, the overall effect of these elements are included in tokens. The underlying physical mechanism is complicated and the separate effects of each factor is not focused on,

since the signal is not preferred to be recorded in a strictly controlled environment. On the other hand, the focus has been on seeking ways to utilize the footstep sounds that naturally occur in SensoRHEX platform.

The power spectrum estimate derived with discrete time Fourier transform (DTFT) is expected and found to be very informative of the surface type since there should be unique distribution of sound energy on different frequency bands for different types of ground interaction as well as the different types of surfaces. Multiplication with the trapezoidal filter bands have simplified the estimate to a selected number of features which is in fact a controlled parameter in experiments. These filters cover a reasonable spectrum of 100-10000 Hz and as the number of bands increase each band gets thinner. For the finer bands, functional tree algorithm performed much better compared to various other algorithms whereas the performance of Naive Bayes algorithm did not change very much and even decreased in some of the cases. Various algorithms are evaluated in terms of timing concerns and the one closer ones to the high success rate values and low time values in the computational complexity vs success rate graph are preferred. The zero crossing rate derived from the time domain data and the sum of derivative spectrum feature is put into classifier as features and their effect on success is examined in detail. In the end, the spectral subtraction of the motor sound from the spectrum is explained and applied. Contrary to the expectations, the results are not improved as expected. This may either be due to the suppression of informative features in the tokes with spectral subtraction or any inadequacy in application of the method. As an improvement, feature weighting should be reconsidered and instead of using weigths that are determined heuristically, the methods that consider variance of features like Mahalanobis distance can be used.

6.1 APPLICABILITY OF THE WORK

Although the similar surfaces can be confused more compared to different surfaces, the ratio of this confusion is not high to detriment the applicability of the approach. For a particular set of features derived with a controlled set of parameters and for some selected pattern classification algorithms success rates are high. However, there should be more data collected and analyzed for more statistical reliability. These successful sets for the selected data set can be stated as functional tree algorithm implemented to a signal obtained with a more holistic approach. In this work, the holistic approach can be explained as not cutting out the motor

noise that is embedded into signal and taking three-four footsteps of the robot into consideration before making a decision. Being the first classifier implementation, vector quantizer algorithm performed quite well up to four classes. However, when the number of classes exceeded five, the drop in performance has been very dramatic. For six classes, the succes rates have been maximum around 50 %. This rate is still higher than random selection of classes, which is $1/6$ for a six class problem by simple probability equation. The classification results for Functional Tree algorithm and Simple Logistic algorithm are considerably high and Naive Bayes algorithm performs approximately 10 % better than VQ. As the length of the selected signal for each patter decreases, the success rate drops however, the rate is still considerably high around 70 %. Even for a newly added class that is quite similar to one of the classes, the confusion increased as expected but did not lead to worse results. This is due to the acoustical difference between the two interactions with the two similar surfaces. The results are examined in terms of speeds and increase of speed has resulted in lower success rates. For the slow and mid speed, the success rate was quite high around 80 % for many feature sets and some very definitive feature sets are discovered. LogitBoost logistic linear regression algorithm performs very good results on highest speed however, it is relatively slow compared to Functional Tree and Naive Bayes algorithms for this application.

With a fast analysis embedded to its software, a legged robot can have models for surfaces that are formed from surface type in an empirical manner. This information can be useful in adjusting the gait parameters of the robot and can be even used as additional information in mapping of the traveled environment. In this study, the focus has been on finding the record duration, the spectral and time domain features and the classification algorithms that result in determining the type of the surface effectively. This data could then be directly used in adjustment of gait parameters, these gait parameters is expected to vary in a considerable amount for dissimilar surfaces. The data obtained from the surface type could also be combined with the outputs of other sensors like accelerometers for more complex decisions related to robot body dynamics. Another interesting analysis could be to analyze the current data measured from each hip motor and there can be correlation sought between the current sensor records with the acoustic data.

6.2 FUTURE DIRECTIONS

There are both theoretical and practical directions that may be followed after this work to continue. As a former direction, there can be a theoretical relation sought between the surface hardness, stiffness, damping and the acoustic emissions. In this model, the C shaped legs of the robot and the ground can be modeled as springs with a certain stiffness values. A mechanical model with damping and stiffness would model the contact of the robot body to the surface. In addition to this pure mechanical and vibrational view, there can be some work to do on motor noise with a more theoretical signal processing approach rather than directly focusing on the outcome of the discrete time Fourier transform as done in this study. A more detailed motor model can be constructed with estimating the important harmonics and validating them with the empirical data. Another study on the relation of the rolling sound of the legs to the surface roughness.

As a further direction, more work can be done in identification of surfaces while turning or going backward as well. A more interesting research can be conducted is to detect abnormalities in motor and gearbox sounds and such a work could be considered as an application of machine health monitoring. State correction could be done based on surface type determination. A hearing library of the robot could be created with new records from the same and the different surfaces and this can be fed with various off-line training.

It is believed that, as mentioned above if the both directions, the theoretical and the empirical work are handled separately or are combined, there can be various interesting outcomes. A fully autonomous robot behavior with gait parameter control that makes use of these surface information derived by acoustic means would surely include several important concepts of robotics and be an important step in the field of mobile robotics. A fault detection mechanism that does sub-system diagnostics can be applied to the robot's embedded software. Although this seems to be a fully empirical work, such a study can include important theoretical parts in signal processing and pattern classification fields. If the robotics in general and mobile robotics in particular are considered, acoustic sensing and perception of surface type is believed to be very useful in these areas and there can be rich amount of academic work done on these fields as well as various applications of research and development with considerable success.

REFERENCES

[1] Q. Bone and R. H. Moore. *Biology of Fishes*. Taylor & Francis Group, 2008.

[2] P. E. Nachtigall, D. W. Lemonds, and H. L. Roitblat. *Hearing by whales and dolphins*. Springer-Verlag New York, Inc., 2000.

[3] P. Rayson, G. Leech, and M. Hodges. Social differentiation in the use of english vocabulary: Some analyses of the conversational component of the british national corpus. *International Journal of Corpus Linguistics*, 2:133–152, 1997.

[4] Y. Bar-Cohen. *Biomimetics: biologically inspired technologies*. CRC/Taylor & Francis, 2006.

[5] L. L. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*. Prenctice Hall, 1978.

[6] H. G. Okuno and K. Nakadai. Real-time sound source localization and separation based on active audio-visual integration. In *Proceedings of the Artificial and natural neural networks 7th international conference on Computational methods in neural modeling - Volume 1*, 2003.

[7] C. Breazeala and L. Aryananda. Recognition of affective communicative intent in robot-directed speech. *Autonomous Robots*, 12:83–104, 2002.

[8] J. Valin, S. Yamamoto, J. Rouat, F. Michaud, K. Nakadai, and H. G. Okuno. Robust recognition of simultaneous speech by a mobile robot. *IEEE TRANSACTIONS ON ROBOTICS*, 23:2007, 2005.

[9] G. Amit, N. Gavriely, and N. Intrator. Cluster analysis and classification of heart sounds. *Biomedical Signal Processing and Control*, 4:26–36, 2009.

[10] C. K. Tan and D. Mba. Identification of the acoustic emission source during a comparative study on diagnosis of a spur gearbox. *Tribology International*, 38:469–480, 2005.

[11] A. V. Oppenheim and R. W. Schafer. *Discrete TIme Signal Processing*. Prenctice Hall., 2009 (3rd Ed.).

[12] D. E. Newland. *An Introduction to Random Vibrations, Spectral and Wavelet Analysis*. Prentice Hall, London, 3rd edition, 1994.

[13] W. W. Gaver. *Everyday listening and auditory icons*. PhD thesis, Univeristy of California in San Diego, 1988. Chair-Norman, Donald A.

[14] R. S. Durst and E. P. Krotkov. Object classification from analysis of impact acoustics. *Intelligent Robots and Systems, IEEE/RSJ International Conference on*, 1:90, 1995.

[15] R. S. Durst and E. P. Krotkov. Object classification from acoustic analysis of impact. Technical report, Robotics Institute, Carnegie Mellon University, 1993.

[16] E. Krotkov, R. Klatzky, and N. Zumel. Robotic perception of material: Experiments with shape-invariant acoustic measures of material type. In O. Khatib and K. Salisbury, editors, *Experimental Robotics IV*. Springer-Verlag, 1996.

[17] R. L. Klatzky, D. K. Pai, and E. P. Krotkov. Perception of material from contact sounds. *Presence: Teleoper. Virtual Environ.*, 9(4):399–410, 2000.

[18] J. L. Richmond. Automatic measurement and modelling of contact sounds. Master's thesis, The University of British Columbia, 2000.

[19] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[20] P. Fitzpatrick. *From First Contact to Close Encounters: A Developmentally Deep Perceptual System for a Humanoid Robot*. PhD thesis, Massachusetts Institute of Technology, 2003.

[21] E. Torres-Jara, L. Natale, and P. Fitzpatrick. Tapping into touch. In Luc Berthouze, Frédéric Kaplan, Hideki Kozima, Hiroyuki Yano, Jürgen Konczak, Giorgio Metta, Jacqueline Nadel, Giulio Sandini, Georgi Stojanov, and Christian Balkenius, editors, *Lund University Cognitive Studies*, volume 123, pages 79–86. Lund University Cognitive Studies, 2005.

[22] S. Chu, S. Narayanan, C. C. J. Kuo, and M. J. Matarić. Where am i? scene recognition for mobile robots using audio features. In *In Proceedings of the IEEE International Conference on Multimedia and Expo*, 2006.

[23] J. Sinapov, M. Wiemer, and A. Stoytchev. Interactive learning of the acoustic properties of objects by a robot. In *Not yet known*, 2008.

[24] J. Sinapov, M. Wiemer, and A. Stoytchev. Interactive learning of the acoustic properties of household objects. In *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*, pages 3937–3943, Piscataway, NJ, USA, 2009. IEEE Press.

[25] A. C. C. Warnock. Controlling the transmission of impact sound through floors. Technical report, Canada National Research Council, 1999.

[26] U. Saranli, M. Buehler, and D. E. Koditschek. Rhex - a simple and highly mobile hexapod robot. *International Journal of Robotic Research*, 20:616–631, 2001.

[27] Sony ecm-ds70p microphone specifications, 2010.

[28] G. K. Gultekin, E. Ege, and F. Uzer. An experimental robot motion model for sensorhex legged mobile robotic platform. Technical report, METU EE780 Technical Report, Jan. 2011.

[29] Audacity - the free, cross-platform sound editor.

[30] L. E. Kinslera, A. R. Frey, A. B. Coppens, and J. V. Sanders. *Fundamentals of Acoustics*. John Wiley & Sons, Inc., 4 edition, 1999.

[31] Y. Eprahim and H. L. Vantrees. A signal subspace approach for speech enhancement. *IEEE Transactions On Speech And Audio Processing*, 3:251–266, 1995.

[32] The iris data set.

[33] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringera, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explorations*, 11, 2009.

[34] A. K. Jain and S. Z. Li. *Handbook of Face Recognition*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[35] J. Mariani, editor. *Language and Speech Processing*. John Wiley & Sons, 2009.

[36] J. Marvan. Voice activity detection method and aparatus for voiced/unvoiced decision and pitch estimation in a noisy speech feature extraction, 2007.

[37] N. Landwehr and E. Frank M. Hall. Logistic model trees. *Machine Learning*, 59:161–205, 2005.

[38] J. Gama. Functional trees. *Machine Learning*, 55:219–250, 2004.

[39] D. W. Hosmer and S. Lemeshow. *Applied logistic regression*. John Wiley & Sons, 1989.

[40] R. Kohavi and F. Provost. Glossary of terms. *Machine Learning*, 30:271–274, 1998.

[41] K. Teknomo. kmeanscluster - simple k means clustering.

[42] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on ASSP*, 27(2):113–120, 1979.

APPENDIX A

SONY ECM-DS70P MICROPHONE SPECIFICATIONS

Technical specifications

General - Microphone - Connections - Miscellaneous - Manufacturer Warranty - Universal Product Identifiers

General	
Product Type	Microphone
Width	2.3 in
Depth	0.6 in
Height	2.2 in
Weight	0.4 oz
Color	Silver
Recommended Use	Portable audio system
Additional Features	Gold-plated plug
Localization	English
Microphone	
Type	External
Microphone Technology	Electret condenser
Microphone Operation Mode	Stereo (uni-directional x 2)
Connectivity Technology	Wired
Sensitivity	-38 dBV/Pascal
Response Bandwidth	100 - 15000 Hz
Noise Level	34 dB
Max Sound Pressure	110 dB
Audio Input Details	Uni-directional - 100 - 15000 Hz
Connections	
Connector Type	1 x microphone (mini-phone stereo 3.5 mm)
Miscellaneous	
Cables Included	Microphone cable - external - 3.3 ft
Included Accessories	Microphone clip

Figure A.1: SONY ECM-DS70P Microphone Specifications