Characterization and Analysis of sensor data using

Recurrence Network Analysis

A Dissertation Presented

by

SIVARIT SULTORNSEN

The Department of Mechanical and Industrial Engineering

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the field of

Interdisciplinary Engineering

Northeastern University

Boston, Massachusetts

May 2012
Abstract

For the past many decades, several concepts and measures for studying nonlinear sensor data have been proposed and investigated. There have been many attempts to understand behavior, reliability, and performance of sensor data. This dissertation presents novel methodologies for analyzing, classifying, and recognizing patterns of nonlinear sensor data based on recurrence network analysis.

First, a comprehensive overview of recurrence theory and their quantification possibilities is presented. New measures of recurrence networks are defined by using the complex network properties. These measures are intended to recognize and classify patterns of sensor data using the feature extraction method. In this dissertation, we introduce new methodologies to classify sensor data based on recurrence quantification analysis with artificial neural network classifier. In addition, we present methods to classify and recognize patterns of sensor data based on a recurrence network. The goal of the proposed method is to compute all symmetric patterns and super families of any similar structure, even if they are non-periodic or multivariate time series.

We introduce and apply the recurrence based feature extraction method to complex and nonstationary sensor data such as physiological signals and machining sensor signals. In the lung sounds classification problem, our results show that the proposed method gives 100% classification performance. In the classification of
electromyogram signals, our results show that the proposed method classifies these signals with 98.28% accuracy. Lastly, the proposed method was applied to estimate surface roughness in turning process. Acoustic emission signals are transformed into recurrence plots and a set of recurrence statistics are computed using the recurrence quantification analysis. The surface roughness parameters are estimated using a multilayer neural network, taking the recurrence statistics of acoustic signals as inputs. The estimation accuracy of the proposed method is in the range of 90.13% to 91.26%. Furthermore, these accurate results indicate that the proposed method is very effective and amenable for practical implementation.

Using the phase synchronization method, we construct the stock correlation network. It is used for observing, analyzing, and predicting the stock market dynamics. The proposed method captures the dynamic behavior of the time series of stocks and mitigates the information loss. It provides valuable insights into the behavior of highly correlated stocks which can be useful for making trading decisions. The network exhibits a scale free degree distribution for both chaotic and non-chaotic periods.

We illustrate similarities and dissimilarities with respect to the classification by considering large-scale sensor data. The selected applications of the introduced techniques to data from different applications demonstrate the ability of these techniques. We derive new methods to examine massive, multi-dimensional, multi-source, time-varying information stream of data.
# Table of Contents

Abstract ........................................................................................................................................ ii  
List of Figures ............................................................................................................................... vii  
List of Tables ................................................................................................................................. x  
1. Introduction ................................................................................................................................ 1  
  1.1 Overview .................................................................................................................................. 1  
  1.2 Review of Sensor Data Analysis .............................................................................................. 2  
  1.3 Purpose and Scope .................................................................................................................. 6  
2. Recurrence Analysis .................................................................................................................. 10  
  2.1 Recurrence Theory .................................................................................................................. 10  
  2.2 Phase Space Reconstruction ................................................................................................... 12  
  2.3 Recurrence Plots ...................................................................................................................... 17  
  2.4 Recurrence Quantification Analysis ....................................................................................... 19  
  2.5 Summary .................................................................................................................................. 23  
3. Recurrence Network .................................................................................................................. 24  
  3.1 Review of Complex Network ................................................................................................... 24  
  3.2 Structure of Complex Networks .............................................................................................. 26  
  3.3 Basic Properties of Complex Networks .................................................................................. 29  
      3.3.1 Node degree and degree distribution .............................................................................. 29  
      3.3.2 Shortest path length, diameter and betweenness .......................................................... 29  
      3.3.3 Clustering ..................................................................................................................... 31  
      3.3.4 Network motifs ............................................................................................................. 31  
  3.4 Transformation of Sensor Data into Recurrence Network ....................................................... 32  
  3.5 The Quantitative Analysis of Recurrence Networks ................................................................ 33  
      3.5.1 Edge density (global recurrence rate) ............................................................................ 34  
      3.5.2 Clustering coefficient of recurrence networks ............................................................. 34  
      3.5.3 Transitivity of recurrence networks .............................................................................. 36  
  3.6 New measures of Recurrence Networks ................................................................................ 36
3.6.1 The community structure of recurrence networks .................................................. 37
3.6.2 The modularity of recurrence networks ................................................................. 40
3.6.3 The Shannon’s entropy of degree distribution ......................................................... 41
3.7 Summary .................................................................................................................... 41

4. Recurrence based Classification and Estimation Analysis ........................................... 43

4.1 Recurrence based Feature Extraction Method ............................................................. 44
   4.1.1 Recurrence preprocessing (step 1) ......................................................................... 44
   4.1.2 Feature extraction (step 2) .................................................................................... 47
   4.1.3 Neural network classifier and estimator (step 3) .................................................... 49

4.2 Experiments ................................................................................................................ 50
   4.2.1 Recurrence preprocessing (Step 1) ......................................................................... 51
   4.2.2 Feature extraction (Step 2) .................................................................................... 53
   4.2.3 Neural network and support vector machine classifier (Step 3) ......................... 53

4.3 Summary .................................................................................................................... 54

5. The Phase Synchronization Method based Correlation Networks ............................ 56

5.1 Review of Phase Synchronization Method ................................................................. 57
5.2 Review of an Example Correlation Network ............................................................. 61
5.3 Construction of Correlation Networks using Phase Synchronization Method .......... 63
5.4 Scale-Free Network in the Phase Synchronization based Correlation Network ......... 65
5.5 Experiments ................................................................................................................ 65
5.6 Summary .................................................................................................................... 68

6. Recognition of patterns in Sensor data using Recurrence Network Analysis ............ 69

6.1 Features of Recurrence Networks .............................................................................. 69
   6.1.1 Isomorphic Networks .......................................................................................... 69
   6.1.2 Network Motifs .................................................................................................. 70
   6.1.3 Degree Centrality Measurement ....................................................................... 70

6.2 Experiments ................................................................................................................ 71
   6.2.1 Similarity patterns of sensor data using isomorphism ......................................... 71
   6.2.2 Superfamily phenomenon of sensor data using network motifs ....................... 73

6.3 Summary .................................................................................................................... 77
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>Lung Sounds Classification via Recurrence Quantification Analysis</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>7.1.1 Review of lung sounds classification</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>7.1.2 A recurrence based feature extraction method for lung sounds classification</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>7.1.3 Experiments and results</td>
<td>88</td>
</tr>
<tr>
<td>7.2</td>
<td>Classification of Electromyogram using Recurrence Quantification Analysis</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>7.2.1 Review of electromyogram classification</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>7.2.2 A recurrence based feature extraction method for electromyogram classification</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>7.2.3 Experiments and results</td>
<td>96</td>
</tr>
<tr>
<td>7.3</td>
<td>Phase Synchronization Approach to Stock Correlation Network</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>7.3.1 Review of stock correlation network</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>7.3.2 The phase synchronization method based stock correlation network</td>
<td>102</td>
</tr>
<tr>
<td></td>
<td>7.3.3 Experiments and results</td>
<td>104</td>
</tr>
<tr>
<td>7.4</td>
<td>Recurrence based Approach to Estimate Surface Roughness in turning processes</td>
<td>106</td>
</tr>
<tr>
<td></td>
<td>7.4.1 Review of surface roughness estimation</td>
<td>107</td>
</tr>
<tr>
<td></td>
<td>7.4.2 Recurrence based feature extraction</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>7.4.3 Experiments and results</td>
<td>116</td>
</tr>
<tr>
<td>8.1</td>
<td>Methodical Developments</td>
<td>121</td>
</tr>
<tr>
<td></td>
<td>8.1.1 Recurrence based classification and estimation analysis</td>
<td>121</td>
</tr>
<tr>
<td></td>
<td>8.1.2 Correlation network using phase synchronization method</td>
<td>122</td>
</tr>
<tr>
<td>8.2</td>
<td>Applications</td>
<td>123</td>
</tr>
<tr>
<td></td>
<td>8.2.1 Lung sounds classification</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>8.2.2 Electromyogram classification</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>8.2.3 Stock correlation network using phase synchronization method</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>8.2.4 Surface roughness estimation in turning processes</td>
<td>126</td>
</tr>
<tr>
<td>Reference</td>
<td></td>
<td>127</td>
</tr>
</tbody>
</table>
List of Figures

Fig. 2.1 The fraction of false nearest neighbors as a function of the embedding dimension for the Lorentz system and the Henon system ......................................................... 15

Fig. 2.2 Histogram of the Lorentz system based on mutual information .................. 16

Fig. 2.3 Histogram of the Henon system based on mutual information .................. 16

Fig. 2.4 Examples of recurrence plot of different signals ........................................ 18

Fig. 2.5 Three commonly used norms for the neighborhood with the same radius around a point (black dot) shown for the two-dimensional phase space: (A) L1-norm, (B) L2-norm, and (C) L∞-norm [13] ................................................................. 19

Fig. 3.1 Adjacency matrix and graphical representation of different networks .......... 28

Fig. 4.1 Methodology of sensor data classification based on recurrence quantification analysis and neural networks ................................................................. 43

Fig. 4.2 Feature extraction using the recurrence quantification analysis ............... 48

Fig. 4.3 The neural network architecture for classification and estimation .............. 50

Fig. 4.4 (a) An example of sampled signals and (b) transformation of sampled signal into recurrence plots ................................................................. 52

Fig. 5.1 An example of a sinusoidal signal without noise and a sinusoidal signal with additive noise ................................................................. 60

Fig. 5.2 The correlation probability of recurrence (CPR) between a sinusoidal signal without noise and a sinusoidal signal with additive noise ......................... 61

Fig. 5.3 The construction of a phase synchronization based correlation network from
5 sinusoidal signals and 5 exponential signals .......................................................... 66

**Fig. 5.4** The 10 generated sampled signals with additive noise .......................... 67

**Fig. 6.1** (a) Identical subsignals but shifted in time, (they have completely different amplitudes at any time instance of sampling), (b) subsignals with different forms but which are approximately equal in amplitude, and (c) transformation of sampled signals into complex network using Recurrence Networks .............................................. 72

**Fig. 6.2** (a) Sampled sinusoidal signal, and (b) the pattern of sinusoidal signal using network motifs ................................................................. 74

**Fig. 6.3** (a) Sampled noisy sinusoidal signal, and (b) the pattern of noisy sinusoidal signal using network motifs ................................................................. 75

**Fig. 6.4** (a) Lung sound signal, and (b) the pattern of lung sound signal using network motifs ................................................................. 76

**Fig. 7.1** Examples of lung sound signal ................................................................. 80

**Fig. 7.2** The steps of the recurrence based feature extraction method for lung sounds .. 83

**Fig. 7.3** Examples of the calculated recurrence plots: (a) normal inspiratory phase of bronchial sound, (b) normal expiratory phase of bronchial sound, (c) crackle, (d) rhonchi, and (e) wheeze ................................................................. 85

**Fig. 7.4** The distribution of training data based on five categories of lung sounds ....... 86

**Fig. 7.5** Comparison of accuracy performance of the classification of lung sounds ...... 89

**Fig. 7.6** Electromyogram signals of neuromuscular diseases ..................................... 92

**Fig. 7.7** The steps in the proposed method for classification of EMG signals ........... 94

**Fig. 7.8** The distribution of EMG based on three classes of neuromuscular disorder .... 95

**Fig. 7.9** Position of the neuromuscular phases in the \((DET, LAM, RR)\) feature space ... 98
Fig. 7.10 Comparison of accuracy performance of the classification of neuromuscular disorders ................................................................. 99

Fig. 7.11 Network parameters of the stock correlation network constructed using phase synchronization method for years 2000 - 2009 ........................................ 105

Fig. 7.12 Steps of the proposed method for estimation of surface roughness parameters ........................................................................ 112

Fig. 7.13 An example of an acoustic emission signal and its recurrence plot .......... 114

Fig. 7.14 The neural network architecture for surface roughness estimation ............ 116

Fig. 7.15 Setup for acquiring AE signals from a rotating workpiece ................. 117

Fig. 7.16 Measured surface roughness values versus estimated values for test data ... 119
List of Tables

Table 4.1 Accuracy classification of sampled sinusoidal and exponential signals with additive white noise ................................................................. 54
Table 7.1 Comparison of all experimental results ............................................. 88
Table 7.2 The classification accuracy of neuromuscular disorders in our methodology 97
Table 7.3 Statistics of estimation error ................................................................. 119
Acknowledgment

This work could not have been achieved without help and support of many people.

First, I wish to express my gratitude to my advisors, Prof. Kamarthi and Prof. Zeid, who have given me the chance for this PhD study and, of course, have encouraged and inspired me with new ideas and with fruitful insights and discussions.

I am grateful to the members of Network/Nano Science and Engineering Laboratory for the talks, discussions, the good and cooperative atmosphere and the countless coffee, tea and cake. In particular I would like to thank Dr. Norbert Marwan of the Potsdam Institute for Climate Impact Research, Germany, who has given me the guidance and the help regarding the recurrence theory. Further I sincerely thank Arjun Duvvuru and Srinivasan Radhakrishnan for the interesting talks and discussions about complex networks, for sharing their ideas and for the fruitful co-operation and I wish them success and good luck for their future.

My special thanks go to my parents and siblings for instilling in me the seeds of interest and determination that led me to this career choice, and my lovely wife and wonderful daughter for being extremely patient and making the process worthwhile.

Finally I would like to take this opportunity, to personally, thank everyone who has assisted me, in any way, in getting my PhD degree.
Dedication

I dedicate this work to my parents, family, and friends.
Chapter 1

1. Introduction

1.1 Overview

The investigation of sensor data in nature and engineering has discovered that the underlying nonlinear processes have to be taken into account in order to understand and model these systems. When dealing with sensor data, large amounts of data accumulate quickly. Often, measurements in time are dependent even as the behavior of sensor data changes slowly or turbulently. In the last few decades, data analysis using linear methods were further improved and enriched with new methods which were derived from dynamical systems. Many researchers have tried to measure and analyze nonlinear sensor data (Kantz et al., 1997; Marwan et al., 2007). These dynamics motivate the use of feature extraction methods and dynamical systems that describe lengthy sensor data with a few features that capture the pattern of sensor data. There exist multiple feature extraction methods (Nanopoulos et al., 2001; Kakizawa et al., 1998; Morchen et al., 2003; Deng et al., 1997), however, many of these current methods are not highly successful in characteristic classification of sensor data due to their complexity, nonlinearity, and non-stationarity. Another problem is that most methods, such as scaling laws or fractal dimensions need rather long or stationary data series – both are not typical features of time series data which are collected from natural phenomena. Thus, the results of data analysis should be considered with healthy dose of skepticism. There are plenty of opportunities to develop new techniques of nonlinear sensor data analysis.
The problem of nonlinear sensor data recognition and classification is important to many applications. In investigation of nonlinear sensor data the goal is to classify and detect interesting patterns that are helpful to recognize the regularities in complex information and thereby improve the understanding of the system.

The main aim of this dissertation is to investigate recognition and classification techniques of nonlinear sensor data based on recurrence quantification analysis and recurrence network analysis. Methods for recognition and classification of patterns in nonlinear sensor data are proposed. An important point of classification and recognition methodologies is to extract the specific features of nonlinear sensor data based on recurrence quantification analysis and recurrence network analysis. This methodology is designed to combine modern nonlinear approaches for data analysis in order to both improve accuracy and performance of classification. Our goal is to achieve near perfect accuracy on classification problems.

1.2 Review of Sensor Data Analysis

Related work in sensor data analysis has mainly focused on classification and recognition. The problem of sensor data classification is an important task in sensor data analysis that has attracted increasing interest in the past several decades.

The historical approaches to the problem of classification among different classes of sensor data can be divided into two distinct categories: optimality and feature extraction. The optimality approach, found in the engineering and statistics literature (Fukunaga et al., 1990; Duda et al., 2001), assume a specific probability density function of the separate groups and then develop solutions that satisfy minimum error criteria.
Typically, in the sensor data case, one might assume a difference between classes expressed through differences in the theoretical mean and covariance functions and use some methods to develop an optimal classification function.

Subsequently, a second class of techniques might be described as a feature extraction approach. This approach looks at quantities that tend to be good visual information for well-separated populations and have some basis in physical theory or intuition. Less attention is paid to finding functions that are approximations to some well-defined optimality criterion.

There is extensive research on the feature extraction approaches. For example, Nanopoulos et al. (2001) proposed a time series classification method based on feature extraction. He extracted four basic statistical features from Control Chart Pattern data and used them as input in a multi-layer perceptron neural network for time series classification. Their experiments results showed the robustness of the method against noise and time series length compared to other methods using each and every data points for the same purpose.

Kakizawa et al. (1998) proposed the method of discrimination for multivariate time series by using the Kullback-Leibler discrimination information and the Chernoff information measure. He proposed parametric models for discriminating and clustering multivariate time series with applications to environmental data.

By using two popular feature extraction techniques, the Discrete Wavelet Transform and the Discrete Fourier Transform, Morchen et al. (2003) has demonstrated
the advantages of feature extraction for time series clustering in terms of computational efficiency and improved clustering performance of a benchmark dataset.

For a given time series, the parameters of the Auto Regression Moving Average (ARMA) model are estimated and used as a limited dimensional vector for the original time series in a classification problem (Deng et al., 1997). However, using ARMA parameters is not a reliable method because different sets of parameters can be obtained even from time series with similar structure that could affect the clustering results dramatically.

As demonstrated on real datasets by Ge et al. (2000), an approach for time series pattern matching based on segmental semi-Markov models has shown usefulness, flexibility, and accuracy. The time series is modeled as $k$ distinct segments with constraints on how the segments are linked as the representation of the data before applying a Viterbi-like algorithm to compute the similarity. The Viterbi-like algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states.

Compression-based Dissimilarity Measures (CDM) is proposed by Keogh et al. (2004) and others to compare long time series structure using co-compressibility as a dissimilarity measure. This measure can then be directly used in data mining algorithms, such as hierarchical clustering. Their extensive experiments have demonstrated the ability in handling different-length and missing-value time series. There are dozens of similar examples in the literature.

Over the decades many researchers have been investigating the behavior of dynamical systems via sensor data (Katok et al., 1995; Birkhoff et al., 1931; Feller et al.,
These investigations capture the characteristic features of sensor data. It allows us to study and analyze the nonlinear sensor data. Several applications take dynamical systems and sensor data approach for pattern recognition (Marwan et al., 2002; Zbilut et al., 1992; Packard et al., 1980; Marwan et al., 2007; Bukkapatnam et al., 1995). The studies of recurrence theory have been proposed to recognize the patterns of sensor data (Marwan et al., 2007; Zbilut et al., 1992).

Most methodologies of nonlinear time series analysis were introduced by Kantz et al. (1997). In contrast, their methods do not extract typical features of data series which are collected from nature. In 1994, a quantification of recurrence plots was developed by Zbilut et al. (1994). Recently, Marwan et al. (2007) extended the plots with new measures of complexity for recurrence quantification analysis which quantifies the number and duration of recurrences of a dynamical system presented by its state space trajectory. The investigation of recurrence statistics, which is called recurrence quantification analysis (Marwan et al., 2007), has been applied to recognize the patterns of sensor data (Masugi et al., 2009; Yang et al., 2011; Xie et al., 2011). This and further evidence suggest that recurrences contain all relevant information about a system’s behavior.

Recently, Zhang et al. (2008) introduced complex networks constructed from time series including periodic signals, periodic signals with noise, chaotic signals, hyperchaotic signals and white fractal noise. They showed the local properties of the networks, and the distribution of subgraphs within networks. More recently, recurrence networks were introduced (Donner et al., 2010). Their concept of transforming a time series to a complex network allows reinterpretation of many network-theoretic measures in terms of characteristic phase space properties of a dynamical system.
1.3 Purpose and Scope

We study nonlinear sensor data problems to extract the characteristics and to classify sensor data, to construct recurrence networks, and to conduct recurrence quantification analysis. Firstly, we develop a new feature extraction method based on recurrence theory and dynamical concepts. We investigate new feature extraction methods based on nonlinear data analysis approaches as follows:

(a) Feature Extraction based on Recurrence Quantification Analysis

For this scheme, we transform nonlinear sensor data into phase trajectories and recurrence plots. Thus, we study their properties using recurrence quantification analysis. For this purpose, we investigate the classification method based on feature extraction of recurrence plots.

(b) Feature Extraction based on Recurrence Network

In order to investigate the nonlinear sensor data via recurrence networks, we transform the structure of sensor data into adjacency matrix using recurrence techniques. After we generate large-scale recurrence networks, we study their characteristics using complex network analysis. More importantly, we investigate a method to recognize these patterns and classify them using a neural network. Based on our finding we aim to find interesting properties using recurrence network analysis.

We introduce the new connections between the recurrence theory and the classification methods to deal with the complexity and nonstationary of sensor data. After
we develop these new schemes of nonlinear sensor data classification, we test our methods on some specific datasets and present the following applications:

(a) A recurrence quantification analysis based approach to lung sounds classification
   (Sultornsanee et al., 2011a)

(b) Classification of electromyogram signals using recurrence quantification analysis
   (Sultornsanee et al., 2011b)

(c) Recurrence based approach to estimate surface roughness in turning processes

We perform our evaluation on data collected from our own experiments and other public source data. Our goal is to achieve acceptable classification accuracy.

Secondly, we develop the correlation network using phase synchronization method (Sultornsanee et al., 2011c). It is built on the sensor data correlation. It is used for studying and analyzing the sensor data dynamics. The phase synchronization method captures the dynamic behavior of the sensor data and mitigates the information loss. Existing correlation methods, such as the minimum spanning tree (Vandewalle et al., 2001; Bonnanno et al., 2003; Mantegna et al., 1999), planar maximally filtered graph (Tumminello et al., 2005), and winner take all (Tse et al., 2010), lost information due to the connection criterion and thereby fail to include certain highly correlated sensor data. To test the proposed method we use the weekly closing stock prices of the S&P index (439 stocks) from 2000-2009. The phase synchronization based correlation network method provides valuable insights into the behavior of highly correlated stocks which can be useful for making trading decisions.
Lastly, we describe the similarity measures within recurrence networks for recognizing patterns in sensor data. Some properties of complex network are described by the statistics of nonlinear sensor data. It should be noted that many recurrence networks that have same basic global properties may have different local structures. Conversely, the recurrence networks with different global properties may have similar local structure. Mounting evidence suggests that there might be strong ties between the global topological properties and key local patterns of a network (Zhang et al., 2006). We deeply analyze the similarity measures for recognizing the patterns of sensor data. We have found that the isomorphism and network motifs of recurrence networks are related to similar patterns in sensor data.

In this dissertation, we focus our work on classifying physiological signals, and machining signals. We investigate the overall analysis and classification of these signals. The reminder of the dissertation is structured as follows. Chapter 2 covers the recurrence theory, plot, and quantification analysis. This chapter introduces the eight most relevant recurrence statistics to nonlinear sensor data. Chapter 3 describes some important concepts of recurrence network. We cover specific theorems and definitions. Also we introduce the new measures of recurrence networks: modularity, community structure, and the Shannon’s entropy of degree distribution. In Chapter 4, we introduce a new method which is the classification and recognition patterns of sensor data. In Chapter 5, we introduce the correlation network using phase synchronization method which describes the relationship between each sensor data and use for observing, analyzing and predicting the sensor data dynamics. In Chapter 6, we describe some similarity measures within recurrence networks for recognizing patterns in sensor data. In Chapter 7, we
apply our methods to physiological signals and machining acoustic emission data.

Chapter 8 presents conclusions of the research.
Chapter 2

2. Recurrence Analysis

Recurrence Analysis is one of the modern nonlinear data analysis approaches. We investigate this method because it focuses on system characterization and knowledge discovery for prediction. This chapter summarizes some crucial concepts and properties which we apply to extract features of nonlinear sensor data. It serves as a basis for the dissertation research. We cover principal theorems and definitions.

2.1 Recurrence Theory

The Poincare recurrence theory is a fundamental characteristic of many dynamical systems and was introduced by Poincare in 1890. In the following century, much progress has been proposed in the theory of dynamical systems. Especially, in the last decades of the 20th century, the development of fast and efficient computers, new and well-known mathematical structures have been discovered in this field. It has been recognized that in a larger framework recurrences are part of one of the three broad classes of asymptotic invariants (Katok et al., 1995):

(a) Growth of the number of orbits of various kinds and of the complexity of orbit families (an important invariant of the orbit growth is the topological entropy);

(b) Types of recurrences; and

(c) Asymptotic distribution and statistical behavior of orbits.
The first two classes are of essentially topological in nature; the last one is naturally related to ergodic theory (Birkhoff et al., 1931; Neumann et al., 1932).

Of the different types of recurrences, the Poincare recurrence is of particular interest to our work. It is based on the Poincare Recurrence Theory (Katok et al., 1995).

However, the theory only guarantees the existence of recurrence but does not tell how long it takes the system to recur. Especially for high-dimensional complex systems the recurrence time might be extremely long. For the Earth’s atmosphere the recurrence time has been estimated to be about $10^{30}$ years, which is many orders of magnitude longer than the time the universe exists so far (Dool et al., 1994).

For hyperbolic systems, the recurrence or first return time appears to exhibit certain universal properties (Balakrishnan et al., 2001):

(a) The recurrence time has an exponential limit distribution;
(b) Successive recurrence times are independently distributed;
(c) As a consequence of (a) and (b), the sequence of successive recurrence times has a limit law that is a Poisson’s distribution.

These properties, which are also important characteristics of certain stochastic systems, such as finite aperiodic Markov chains (Feller et al., 1949), have been introduced for deterministic dynamical systems exhibiting sufficiently strong mixing (Sinai et al., 1970). They have also been shown valid for a spacious class of systems that remains hyperbolic (Hirata et al., 1993).

Recently, recurrence and return times have been studied with respect to their statistic and linked to different other basic characteristics of dynamical systems, such as
the Pesin’s dimension (Afraimovich et al., 1997). The multi-fractal properties of return
time statistics have been studied (Hadyn et al., 2002). They have been linked to rates of
mixing (Young et al., 1999), and the relationship between the return time statistics of
continuous and discrete systems has been investigated (Balakrishnan et al., 2000). It is
important to realize that recurrence plots can help to understand and also provide a visual
impression of these fundamental characteristics.

However, for the study of recurrence plots, the first class of the asymptotic
invariants is important, namely invariants which are linked to the growth of the number
of orbits of various kinds and of the complexity of families of orbits. They consider the
return times and focus on the times at which these recurrences occur, and for how long
the trajectories evolve close to each other (the length of diagonal structures in recurrence
plots is linked to these times).

These considerations show that recurrences are deeply rooted in the theory of
dynamical systems. Many applications have been applied to the study of recurrence
times. Additionally, Eckmann et al. (1986) have introduced recurrence plots and the
recurrence matrix. This matrix contains information about the primitive dynamical
system and can be applied for the analysis of measured time series.

2.2 Phase Space Reconstruction

The states of natural or technical systems change in time. The study of complex dynamics
is an important work in diverse scientific disciplines and their applications.
Understanding, describing and forecasting such changes is importance for our daily life.
Formally, a dynamical system is represented by a phase space, a continuous or discrete
time, and a time evolution law. The elements or points of the phase space represent potential states of the system. Based on work by Takens et al. (1981) and Sauer et al. (1991), the state of a system at an instance time \( i \) can be specified by \( m \) embedded dimensions and \( T \) time delays. The dynamics of time-series data can be expressed by a reconstruction of the phase space trajectory \( \tilde{x}(i) \):

\[
\tilde{x}(i) = (x_i, x_{i+T}, x_{i+2T}, \ldots, x_{i+(m-1)T})
\]  

(2.1)

in the \( m \)-dimensional phase space of the system. Fundamentally, the time-evolution law is a rule that allows determining the state of the system at each moment of time \( T \) from its states at all previous times. Thus the most general time-evolution law is time dependent and has finite memory. However, we define time-evolution laws which enable the calculation of all future states given a state at any particular moment.

For the analysis of time series, embedding parameters, the dimension \( m \) and the delay \( T \), have to be chosen. The selection of parameters, the embedded dimension \( m \) and the time delay \( T \), play an important role in constructing the phase space vectors. However, some studies (Takens et al., 1981; Sauer et al., 1991) have found that it is reasonable to use the embedded dimension \( m = 1 \) and the time delay \( T = 1 \) because these two parameters can be considered to represent the same dynamical system in different coordinate systems. If the classification accuracy is unacceptable, one can reconstruct the phase space trajectories by using estimated parameters as shown in Equations (2.2) and (2.3). The estimation of the smallest sufficient embedding dimension \( m \) has been introduced by using the false nearest-neighbors algorithm (Kennel et al., 1992). Also the mutual information function (Fraser et al., 1986) has been used for selecting an
appropriate time delay $T$.

In order to perform the phase space reconstruction, we need to estimate the values of the two parameters $m$ and $T$. A method to determine the minimal sufficient embedding dimension $m$ was introduced by Kennel et al. (1992). It is called the false nearest neighbor method. It obtains the optimum embedding dimension for phase space reconstruction. By checking the neighborhood of points embedded in projection manifolds of increasing dimension, the algorithm eliminates false neighbors. It means points apparently lying close together due to projection are separated in higher embedding dimensions. A natural criterion for catching embedding errors is that the increase in distance between two neighbored points is large when going from dimension $d$ to $d+1$. The false nearest neighbor algorithm is described as:

$$\left[ \frac{R_{d+1}^2(t, \tau) - R_d^2(t, \tau)}{R_d^2(t, \tau)} \right]^{1/2} > R_{tol}$$

where $t$ and $t_r$ are the times corresponding to the neighbor and the reference point respectively. $R_d$ denotes the distance in phase space with embedding dimension $d$, and $R_{tol}$ is the tolerance threshold.

Equation (2.2) is explained as follows. For each $\tilde{R}_r$ in the time series, we look for its nearest neighbor $\tilde{R}_r$ in $m$-dimensional space. We calculate the distance $R_d(t, \tau)$ and iterate both points in Equation (2.2). If it exceeds a given tolerance threshold $R_{tol}$, this point is marked as having a false nearest neighbor. Two examples are shown in Fig. 2.1. One is for the Lorentz system and another one is for the Henon system. One clearly sees
that, as expected, \( m = 2 \) is sufficient for the Henon and \( m = 3 \) is sufficient for the Lorentz system.

The time delay mutual information was proposed by Fraset et al. (1986) as a tool to determine an appropriate time delay \( T \). The idea is described as:

\[
S = \sum_{ij} p_y(T) \ln \frac{p_y(T)}{p_i p_j}
\]

where, for some partition on the real numbers, \( p_i \) is the probability of finding a time series value in the \( i \) interval, and \( p_y(T) \) is the joint probability that an observation falls into the \( i \) interval and the observation time \( T \) later falls into the \( j \) interval. Two examples are demonstrated. One is for the Lorentz system as shown in Fig. 2.2 and another one for the Henon system as shown in Fig. 2.3.

Fig. 2.1. The fraction of false nearest neighbors as a function of the embedding dimension for the Lorentz system and the Henon system
Fig. 2.2 shows the first minimum occurs at $T = 0.16$. The second slightly lower minimum at $T = 0.62$. We consider the time delay $T$ of the Lorentz system is 0.16. Fig. 2.3 shows the natural delay time $T = 1$ iteration. Given that the mutual information has no local minimum but it is only monotonically decreasing. This suggests that $T = 1$ is a choice reasonably supported independently by the mutual information calculation.

![Lorentz Attractor](image1)

**Fig. 2.2.** Histogram of the Lorentz system based on mutual information

![Henon Map](image2)

**Fig. 2.3.** Histogram of the Henon system based on mutual information
2.3 Recurrence Plots

The recurrence plot is defined as the tool which measures recurrences of a trajectory $\vec{x} \in \mathbb{R}^d$ in phase space. The technique of recurrence plot (RP) was introduced to show a visual representation of recurrences in phase space (Marwan et al., 2002; Marwan et al., 2007). RP allows us to make a qualitative interpretation of hidden information of a time series. It enables us to investigate the $m$-dimensional phase space vector through two-dimensional visualization. It also allows us to analyze the chaotic and non-chaotic patterns of a time series. The recurrence plot regularly visualizes recurrences as shown in Fig. 2.4 and can be correctly formulated by the $N \times N$ matrix:

$$R_{ij}(\varepsilon) = H(\varepsilon - \|\vec{x}_i - \vec{x}_j\|), \quad i, j = 1,..., N \tag{2.4}$$

where $R_{ij}$ is an element of the matrix, $N$ is the number of points measured in $m$-dimensional phase space, $\varepsilon$ is a threshold distance, $H$ is the Heaviside function, and $\|\cdot\|$ denotes a suitable norm in the phase space considered. In this process, $\varepsilon$ threshold distance is normally set for plotting recurrence pointers, which fall in the neighborhood of a fixed size. Therefore, Marwan et al. (2007) introduced the following notion:

$$\vec{x}_i \approx \vec{x}_j \iff R_{ij} = 1 \tag{2.5}$$

Recurrences take place in phase space vectors which are used to create the recurrence plots. The selection of the threshold distance $\varepsilon$, which is an essential parameter of a recurrence plot, is an important step. Several studies for the choice of the threshold distance $\varepsilon$ have been investigated (Mindlin et al., 1992; Zbilut et al., 2002;
Thiel et al., 2002), but they are not highly successful. The choice depends on the application or problem. A systematic investigation depends on the application. Based on recurrence statistics, it has been shown that the threshold distance $\varepsilon = 0.1$ is quite reasonable (Marwan et al., 2007). If it doesn’t work well, we have to reconstruct the recurrence plot by trial and error while adjusting the threshold distance $\varepsilon$.

![Examples of Recurrence Plot of different signals](image)

**Fig. 2.4.** Examples of Recurrence Plot of different signals

The recurrence plot is generated by plotting the recurrence matrix, Equation (2.4), and using different colors for its binary entries, e.g., plotting a black dot at the coordinates $(i, j)$, if $R_{i,j} = 1$, and a white dot, if $R_{i,j} = 0$. Both axes of the recurrence plot are time axes that increase in value rightwards and upwards. Since $R_{i,j} = \prod_{\varepsilon=1,j=1}^{N} 1$ by definition (Marwan et al., 2007), the recurrence plot has always a black main diagonal line. Furthermore, the recurrence plot is symmetric by definition (Marwan et al., 2007) with respect to the main diagonal, i.e., $R_{i,j} = R_{j,i}$ as shown in Fig. 2.4. In order to compute recurrence plot, an appropriate norm has to be chosen. The most frequently used norms (Marwan et al., 2007) are the $L_1$-norm, the $L_2$-norm (Euclidean norm) and the $L_\infty$-norm.
(Maximum or Supremum norm). Fig. 2.5 shows that the neighborhoods of these norms have different shapes.

**Fig. 2.5.** Three commonly used norms for the neighborhood with the same radius around a point (black dot) shown for the two-dimensional phase space: (A) $L_1$-norm, (B) $L_2$-norm, and (C) $L_\infty$-norm (Marwan et al., 2007)

### 2.4 Recurrence Quantification Analysis

The recurrence quantification analysis (RQA) was introduced by Zbilut et al. (1992) for measuring quantitative information hidden in a RP. It performs a nonlinear data analysis to quantify the number of occurrences and the duration of recurrence of a dynamical system presented by its state space vector. It is actually a measure of information complexity. The main advantage of the recurrence quantification analysis is that it can provide useful information even for nonlinear and multivariate data.
Marwan et al. (2007) have introduced new measures of complexity to quantify RPs based on the small-scale structures within. RP mostly contains single dots and lines which are vertical/horizontal or parallel to the mean diagonal referred to as line of identity (LOI). Since an RP is symmetric, horizontal and vertical lines correspond to each other; thus only vertical lines are considered for RQA. The lines capture a typical behavior of the phase space trajectory. While the diagonal lines represent some segments of the phase space trajectory which run parallel for time instance, the vertical lines represent some segments which remain in the same phase space region for some time duration.

The RQA measures are based on the recurrence point density, and diagonal and vertical line structures in an RP. The recurrence quantification analysis measures were defined by Zbilut et al. (1992) and Marwan et al. (2007) and are summarized here for reference:

1. Recurrence rate ($RR$) is the density of recurrence points in an RP. This coincides with the correlation sum. It is the estimator of the correlation integral, which is the mean probability that the states at two different times are close:

$$RR = \frac{1}{N^2} \sum_{i,j=1}^{N} R_{i,j}$$  \hspace{1cm} (2.6)

where $R_{i,j}$ is an element of the matrix and $N$ is the number of points on the phase space trajectory.

2. Determinism ($DET$) is the fraction of recurrence points forming diagonal lines. Diagonal lines represent epochs of similar time evolution of states of the system. The
determinism is given by:

\[ DET = \frac{\sum_{l=l_{\text{min}}}^{N} lP(l)}{\sum_{i,j} R_{i,j}} \]  

(2.7)

where \( P(l) \) is the histogram of the length \( l \) of the diagonal lines, \( l_{\text{min}} \) is the minimum acceptable diagonal line length, and \( N \) is the number of points on the phase space trajectory.

3. Laminarity (\( LAM \)) is the percentage of recurrence points forming vertical lines. Vertical lines are typical for intermittency, which is the alternation of phases of apparently periodic and chaotic dynamics. Therefore, LAM is related to the amount of laminar states in the system:

\[ LAM = \frac{\sum_{v=v_{\text{min}}}^{N} vP(v)}{\sum_{v=1}^{N} vP(v)} \]  

(2.8)

where \( P(v) \) is the histogram of the lengths \( v \) of the vertical lines, \( v_{\text{min}} \) is the minimum acceptable vertical line length, and \( N \) is the number of points on the phase space trajectory.

4. Mean diagonal line length (\( L \)) is the mean prediction time or the inverse of the divergence of the system (\( K_{2} \)-entropy). Thus, it is the mean of the diagonal line lengths:

\[ L = \frac{\sum_{l=l_{\text{min}}}^{N} lP(l)}{\sum_{l=l_{\text{min}}}^{N} P(l)} \]  

(2.9)
where \( P(l) \) is the histogram of the length \( l \) of the diagonal lines, \( l_{\text{min}} \) is the minimum acceptable diagonal line length, and \( N \) is the number of points on the phase space trajectory.

5. **Trapping Time** (\( TT \)) measures the mean time that the system is trapped in a particular state with very slowly change. Thus, it is the average of the vertical line lengths:

\[
TT = \frac{\sum_{v=v_{\text{min}}}^{N} v P(v)}{\sum_{v=v_{\text{min}}}^{N} P(v)}
\]

(2.10)

where \( P(v) \) is the histogram of the lengths \( v \) of the vertical lines, \( v_{\text{min}} \) is the minimum acceptable vertical line length, and \( N \) is the number of points on the phase space trajectory.

6. **Longest diagonal line** (\( L_{\text{max}} \)) is the length of the longest diagonal line:

\[
L_{\text{max}} = \max(\{l_i; i = 1 \ldots N_i\})
\]

(2.11)

where \( N_i \) is the number of diagonal lines in the recurrence plot.

7. **Longest vertical line** (\( V_{\text{max}} \)) is the length of the longest vertical line:

\[
V_{\text{max}} = \max(\{v_i; i = 1 \ldots N_v\})
\]

(2.12)

where \( N_v \) is the number of vertical lines in the recurrence plot.

8. **Entropy** (\( ENTR \)) is the Shannon entropy of the probability distribution of the diagonal lengths \( P(l) \). In other words, the entropy of the line distribution measures the
complexity of the recurrence structure:

\[
ENTR = -\sum_{l = l_{\text{min}}}^{N} P(l) \ln P(l) \quad (2.13)
\]

where \( P(l) \) is the histogram of the length \( l \) of the diagonal lines, \( l_{\text{min}} \) is the minimum acceptable diagonal line length, and \( N \) is the number of points on the phase space trajectory.

### 2.5 Summary

Recurrence plots have been widely used for a variety of purposes such as analyzing nonlinear systems as well as detection of characteristic behavior signals embedded in noise. In this chapter, we gave a comprehensive overview covering recurrence based methods with an emphasis on recent developments. After a brief outline of the theory of recurrences, the basic idea of the recurrence plot with its variation was presented. This included the quantification of recurrence plots, i.e., the recurrence quantification analysis, which is highly effective to detect transitions in the dynamics of system from nonlinear sensor data. Recurrence theory and their quantification are extremely versatile. It is thus used as an analysis tool. In our research, the recurrence quantification analysis is useful to extract the features of nonlinear sensor data. Thus the recurrence based feature extraction method will be introduced later in Chapter 4.
Chapter 3

3. Recurrence Network

Recurrence network analysis is a combination of recurrence analysis of time series and complex network statistics (Donner et al., 2010). The basic idea is to identify the recurrence matrix of a time series or, in general, of a dynamical system with the adjacency matrix of a complex network. This allows calculating statistical measures coming of complex network theory to describe recurrence properties of the dynamical system.

In this chapter, we study the characteristic behavior of sensor data via properties of complex network. We would extract features of sensor data via complex networks. Also we study the interplay between the structure and dynamic of recurrence networks. A classification and recognition method is proposed by using the feature extraction via complex networks and classifying via artificial neural network.

3.1 Review of Complex Network

We further discuss the complex network which is one of the most popular theoretical approaches for analyzing the structural features of complexity information. We summarize the definitions of network and graph theory. We define metrics such as the shortest path length, the clustering coefficient, and degree distribution, which provide a basic characterization of network systems. The large size of many networks makes
statistical analysis a proper tool for a useful mathematical characterization of these systems. For this concept, classical graph theory has been extended by large-scale topological features such as complex networks. So this state of the research is motivated by the desire for understanding and modeling the structure of a complex network to gain new insights about its important mechanisms, and to better analyze its dynamical and characteristic behavior.

The great success of complex networks has been recently established to solve problems in time series (Zhang et al., 2006). Recently, Recurrence Networks were introduced by Donner et al. (2010). They transformed time series data through complex networks.

In very general terms a network is any system that has a representation as a graph whose nodes identify the elements of the system and in which the set of connecting links represent the relation or interaction among those elements. Perhaps the simplest useful model of a network is the random graph, first introduced by Rapoport et al. (1957) and by Erdos et al. (1959). The random graph has been well studied by mathematicians (Bollobas et al., 1985; Bollobas et al., 1998; Janson et al., 1999; Karonski et al., 1982) and many results have been proven. Most of the interesting features of real world networks have been studied by researchers in the last few years. However these studies focused on networks that are not like random graphs.
3.2 Structure of Complex Networks

The graph theory is the study of graphs or networks. The structure of graphs is used to model pairwise relations between objects from a certain collection. A graph is a collection of vertices or nodes and a collection of edges that connect pairs of vertices. A graph may be undirected, meaning that there is no distinction between the two nodes associated with each edge, or its edges may be directed from one node to another.

An undirected graph $G$ is defined by a pair of sets $G = (V, E)$, where $V$ is a set of elements, called vertices or nodes, and $E$ is a set of unordered pairs of different nodes, called edges or links. The edge $(i, j)$ joins the nodes $i$ and $j$, which are said to be adjacent or connected. It is also common to call connected nodes neighbors or nearest neighbors. The total number of nodes in the graph is denoted by $N$; it defines the order of the graph. It is worth remarking that in many biological and physical contexts, $N$ defines the physical size of the network since it identifies the number of distinct elements composing the system. However, in graph theory, the size of graph is identified by the total number of edges $E$.

An interesting class of undirected graphs is formed by hierarchical graphs where each edge (known as child) has exactly one parent (node from which it originates). Such a structure defines a tree and if there is a parent node, or root, from which the whole structure arises, then it is known as a rooted tree.

A directed graph $D$, or digraph, is defined by a set of nodes $V$ and a set of ordered pairs of different nodes $E$ that are called directed edges. In a graphical representation, the
directed nature of the edges is depicted by means of an arrow, indicating the direction of each edge. In an undirected graph the presence of an edge between nodes $i$ and $j$ connects the nodes in both directions. On the other hand, the presence of an edge from $i$ and $j$ in a directed graph does not necessarily imply the presence of the reverse edge between $j$ and $i$.

From a mathematical point of view, it is convenient to consider a representation of a graph. There are two standard representations of graphs which are adjacency matrix representation and adjacency list representation. In adjacency matrix representation, each graph of $N$ nodes is represented by a $N \times N$ matrix (that is a two-dimensional array) in which entry $a_{ij}$ ($i, j = 1, \ldots, N$) is equal to 1 when the link exists, and zero otherwise. The diagonal of the adjacency matrix is equal to zeros. In adjacency list representation, each graph uses incidence matrix, a $N \times K$ matrix in which entry $b_{ik}$ is equal to 1 when the node $i$ is incident with the link and zero otherwise.
Fig. 3.1. Adjacency matrix and graphical representation of different networks

For undirected graphs the adjacency matrix is symmetric, $x_{ij} = x_{ji}$, and therefore contains redundant information. For directed graphs, the adjacency matrix is not symmetric. In Fig. 3.1 we show the graphical illustrations of different undirected and directed graphs and their corresponding adjacency matrices. In the graphical representation of an undirected graph, the dots represent the nodes and pairs of adjacent nodes are connected by a line (edge). In directed graphs, adjacent nodes are connected by arrows, indicating the direction of corresponding edge.
3.3 Basic Properties of Complex Networks

3.3.1 Node degree and degree distribution

The first, and simplest property, is the node degree or connectivity. It is based on the idea that important nodes are defined as the number of links or edges incident upon a node. It is defined in terms of the adjacency matrix as (Nieminen et al., 1974):

\[ k_i = \sum_{j=1}^{N} a_{ij} \quad (i, j = 1, \ldots, N) \]  

(3.1)

where \( k_i \) is the degree of node \( i \) and \( N \) is the total number of nodes in the graph.

The most basic topological characterization of a graph can be obtained in terms of the degree distribution \( P(k) \), defined as the probability that a node chosen uniformly at random has degree \( k \) or, equivalently, as the fraction of nodes in the graph having degree \( k \) (Boccaletti et al., 2006).

3.3.2 Shortest path length, diameter and betweenness

Shortest path or geodesic play an important role in the transport and communication within a network. Assume one needs to travel between two cities, the shortest path provides an optimal way, since one would achieve a fast route. For this reason, geodesic plays an important role in the characteristic of a network. It is useful to represent all shortest path lengths of a graph in term of matrix as (Watts et al., 1998):
\[ L = \frac{1}{N(N-1)} \sum_{i \neq j} d_{ij} \]  \hspace{1cm} (3.2)

where \( d_{ij} \) is the length of the geodesic from node \( i \) to node \( j \).

The maximum value of \( d_{ij} \) is called the diameter of the graph \( G \), and is denoted as \( Diam(G) \). A problem with this definition occurs if there are disconnected nodes in the graph. To avoid such a problem the summation in Equation (3.2) is limited only to connected nodes belonging to largest connected component (Watts et al., 1998).

The communication of two-adjacent nodes \( j \) and \( k \), depends on the nodes belonging to the paths connecting \( j \) and \( k \). Moreover, a measure of the relevance of a given node can be obtained by counting the number of geodesics going through it, and defining so-called node betweenness. The betweenness is one of the standard measures of node centrality. More precisely, if \( b_i \) is the betweenness of a node \( i \), \( n_{jk} \) is the number of geodesics linking two nodes \( j \) and \( k \), and \( n_{jk}(i) \) is the number of geodesics linking the two nodes \( j \) and \( k \) that contain node \( i \). The betweenness centrality of node \( i \) can be defined as (Freeman et al., 1991):

\[ b_i = \sum_{j \neq k} \frac{n_{jk}(i)}{n_{jk}} \]  \hspace{1cm} (3.3)

The concept of betweenness can be extended also to edges. The edge betweenness is defined as the number of shortest paths between pairs of nodes that run through those edges (Newman et al., 2004).
3.3.3 Clustering

Clustering or transitivity is a typical property of associated networks where two individual networks and a common friend are likely to know each other (Wasserman et al., 1994). In terms of a graph, transitivity means the presence of a high number of triangles. This can be quantified by defining the transitivity of the graph as the relative number of transitive triples (Boccaletti et al., 2006):

\[ T = \frac{3 \times \text{numbers of triangles in graph}}{\text{numbers of connected triples of vertices in graph}} \]  

(3.4)

3.3.4 Network motifs

Network motifs are connectivity patterns. The concept of motifs was introduced by Shen-Orr et al. (2002). The significant motifs in a graph are based on matching algorithms counting the total number of occurrences of each node in the graph and in the randomized ones. The statistical significant is described by the Z-score, defined as (Shen-Orr et al., 2002):

\[ Z_M = \frac{n_M - \langle n_M^{\text{rand}} \rangle}{\sigma_{n_M}^{\text{rand}}} \]  

(3.5)

where \( n_M \) is the number of times the subgraph \( M \) appears in graph \( G \), and \( \langle n_M^{\text{rand}} \rangle \) and \( \sigma_{n_M}^{\text{rand}} \) are respectively the mean and standard deviation of the number of appearances in the random network.
3.4 Transformation of Sensor Data into Recurrence Network

Recently, several concepts have been proposed for transforming time series into complex network representations (Donner et al., 2010). Recurrence is one of several concepts that describe fundamental properties of many dynamical processes. Donner (2010) proved the duality between the recurrence matrices in the study of dynamical systems. He also proved that the adjacency matrix of a complex network can be used to construct complex networks from time series.

A recurrence network is a complex network whose adjacency matrix is given by the recurrence matrix of a time series. When observing a scalar time series \( x(t) (t = 1, \ldots, N) \), one may use a suitable \( m \)-dimensional time delay by embedding \( x(t) \) with delay \( \tau \) \([72]\), \( \{X^{(m)}(t) = (x(t), x(t+\tau), \ldots, x(t+(m-1)\tau))\} \), for obtaining a recurrence plot as a graphical representation of the binary recurrence matrix:

\[
R_{i,j}(\epsilon) = \theta(\epsilon - x_i - x_j)
\]  

(3.6)

where \( \theta \) is the Heaviside function, \( \| \| \) denotes a suitable norm in the considered phase space, and \( \epsilon \) is a threshold distance that should be reasonably small.

We define the adjacency matrix of a recurrence network by

\[
A_{i,j} = R_{i,j} - \delta_{i,j}
\]  

(3.7)
where $\delta_{i,j}$ is the main diagonal introduced here in order to avoid artificial self-loops. It corresponds to the consideration of the smallest possible Theiler window in traditional recurrence quantification analysis (Donner et al., 2010).

### 3.5 The Quantitative Analysis of Recurrence Networks

In this section, a brief review of quantitative characteristics for studying the properties of time series by means of recurrence network is presented (Donner et al., 2010). When looking at recurrence networks, one of the main insights is provided by means of complex networks. The definition of nodes in recurrence networks is the state $x(t)$ in phase space reconstruction shown in Equation (2.1). The definition of edges is the recurrence of states (volume). The recurrence networks are considered as the undirected graphs or networks.

While the definitions of nodes and edges have already been given, we now provide a geometrical interpretation of an important network entity, the path, within the framework of recurrence networks. A path between two nodes $i$ and $j$ in a simple graph without multiple edges can be written as an ordered sequence of the nodes, i.e. $(i, k_1, \ldots, k_{l_{i,j}-1}, j)$, where the associated number of edges $l_{i,j}$ measures the length of the path (Donner et al., 2010). Due to the natural interpretation of nodes, edges and paths, the topological characteristics of a recurrence network closely capture the fundamental phase space properties of the dynamical system that has generated the considered time series.
In the following, we will provide a detailed analysis of the corresponding analogies for different network properties that are defined on local and global scale. It has to be emphasized that these quantities can be considered as measures within the framework of RQA.

### 3.5.1 Edge density (global recurrence rate)

As a first measure that allows us to quantify the importance of nodes in a recurrence network, the mean degree of all nodes is defined in terms of the adjacency matrix as (Donner et al., 2010):

\[
\langle k \rangle = \frac{1}{N} \sum_{i=1}^{N} k_i = \frac{2L}{N}
\]  

(3.8)

as a simple characteristic quantity of this distribution, where

\[
L = \sum_{i<j} a_{ij} = \rho \frac{N(N-1)}{2}
\]  

(3.9)

is the total number of edges in the recurrence network. The mean degree centrality \( \langle k \rangle \) is directly proportional to the edge density \( \rho \) of the network or, alternatively, its recurrence plot equivalent, the global recurrence rate \((RR)\) shown in Equation (2.6).

### 3.5.2 Clustering coefficient of recurrence networks

The concept of clustering of a graph refers to the tendency observed in many natural networks to form cliques in the neighborhood of any given nodes. The clustering
coefficient of a node \( v, (C_i) \) characterizes the density of connections in the direct neighborhood of this node in terms of the density of connections between all nodes that are incident with \( v \). In this sense, clustering implies the property that, if the node \( i \) is connected to the node \( j \), and at the same time \( j \) is connected to \( l \), then with a high probability \( i \) is also connected to \( l \). The clustering of an undirected graph can be measured by means of the clustering coefficient which measures the local group cohesiveness (Watts et al., 1998). Given a node \( i \), the clustering \( C(i) \) of a node \( i \) is defined as the ratio of the number of links between the neighbors of \( i \) and the maximum number of such edges. If the degree of node \( i \) is \( k_i \) and if these nodes have \( e_i \) edges between them, we have

\[
C(i) = \frac{e_i}{k_i(k_i - 1)/2} \tag{3.10}
\]

where it is worth remarking that this measure of clustering only has a meaning for \( k_i > 1 \).

For \( k_i \leq 1 \) we define \( C(i) \equiv 0 \). Given the definition of \( e_i \), it is easy to check that the number of edges among the neighbors of \( i \) can be computed in terms of the adjacency matrix \( x \) as:

\[
e_i = \frac{1}{2} \sum_{j\neq i} x_{ij} x_{ij} x_{ji} \tag{3.11}
\]

Also the average clustering coefficient of a graph is simply given by
\[ \langle C \rangle = \frac{1}{N} \sum_i C(i) \]  

(3.12)

### 3.5.3 Transitivity of recurrence networks

Transitivity is based on triplets of nodes. A triplet consists of three nodes that are connected by either two (open triplet) or three (closed triplet) undirected ties. A triangle consists of three closed triplets, one centered at each of the nodes. This is often called global clustering coefficient (Wasserman et al., 1994). This measure gives an indication of the clustering in the global network, and can be applied to recurrence networks (Donner et al., 2010). Formally, it has been defined as:

\[
T = \frac{3 \times \text{numbers of triangles in graph}}{\text{numbers of connected triples of vertices in graph}}
\]

(3.13)

where the factor 3 is due to the fact that each triangle is associated with three nodes. This definition corresponds to the concept of the fraction of transitive triples introduced in sociology.

### 3.6 New measures of Recurrence Networks

Although the recurrence network method (Donner et al., 2010) has successfully transformed the time series into recurrence networks, the author has only described the basic global properties of the network. It should be noted that many networks that have the same basic global properties may have different local structures. In other words, the complex network with different global properties may have similar local structures.
Actually mounting evidence suggests that there might be strong ties between the global topological properties and key local patterns of network (Zhang et al., 2008).

In this section, new quantitative characteristics for studying the properties of sensor data by means of recurrence network are introduced. These properties include the community structure, the modularity, and the Shannon’s Entropy of degree distribution. It should be remarked that these new quantities can be considered as measures within the framework of RQA.

### 3.6.1 The community structure of recurrence networks

Several methods for discovering and evaluating community structures in complex networks have been investigated (Newman et al., 2004). The main idea of these methods is to divide the network nodes into densely connected subgroups. In this section, we describe the simple community detection method which is the Girvan-Newman algorithm introduced by Newman et al. (2004). The Girvan–Newman algorithm is one of the methods used to detect communities in complex systems. The notion of a "community structure" is related to that of clustering, though it isn't quite the same. A community consists of a subset of nodes within which the node-node connections are dense, and the edges to nodes in other communities are less dense.

The hierarchical clustering method is based on assigning a weight for every edge and placing these edges into an initially empty network, starting from edges with strong weights and progressing towards the weakest ones. The edges with the greatest weights within the community are the most central ones. Although traditional in community detection, the method presents some pathologies. One of them for instance, is the
inability to classify in a community a node which is connected to the network with only one edge.

The Girvan–Newman algorithm (Newman et al., 2004) works the opposite way. Instead of trying to construct a measure that tells us which edges are the most central to communities, it focuses on these edges that are least central and the edges that are most "between" communities. The communities are detected by progressively removing edges from the original graph, rather than by adding the strongest edges to an initially empty network.

Vertex betweenness (Newman et al., 2004) has been studied in the past as a measure of the centrality and influence of nodes in networks. For any node $i$, vertex betweenness is defined as the number of shortest paths between pairs of nodes that run through it. It is a measure of the influence of a node over the flow of information between other nodes, especially in cases where information flow over a network primarily follows the shortest available path. The Girvan–Newman algorithm extends this definition to the case of edges, defining the "edge betweenness" of an edge as the number of shortest paths between pairs of nodes that run along it. If there is more than one shortest path between a pair of nodes, each path is assigned equal weight such that the total weight of all of the paths is equal to unity. If a network contains communities or groups that are only loosely connected by a few intergroup edges, then all shortest paths between different communities must go along one of these few edges. Thus, the edges connecting communities will have high edge betweenness (at least one of them). By removing these edges, the groups are separated from one another and so the underlying community
structure of the network is revealed.

The algorithm's steps for community detection are summarized below:

1. Calculate betweenness scores using Equation (2.3) for all edges in the network.
2. Find the edge with the highest score and remove it from the network.
3. Recalculate betweenness for all remaining edges.
4. Repeat steps 2 and 3 until no edges remain.

The fact that the only betweennesses being recalculated are only the ones which are affected by the removal may lessen the running time of the process' simulation in computers. However, the betweenness centrality must be recalculated with each step, or severe errors occur. The reason is that the network adapts itself to the new conditions set after the edge removal. For instance, if two communities are connected by more than one edge, then there is no guarantee that all of these edges will have high betweenness. According to this method, we know that at least one of them will have, but nothing more than that is known. By recalculating betweennesses after the removal of each edge, it is guaranteed that at least one of the remaining edges between two communities will always have a high value.

The end result of the Girvan–Newman algorithm is a dendrogram. As the Girvan–Newman algorithm runs, the dendrogram is produced from the top down (i.e., the network splits up into different communities with the successive removal of links). The leaves of the dendrogram are individual nodes.

In terms of dynamical systems, the community structure method separates regions
3.6.2 The modularity of recurrence networks

The measurement of the strength of the community division is called the *modularity*. It is a function of the particular division of the network into groups, with larger values indicating stronger community structure. Consider a network composed of \( n \) nodes connected by \( m \) links. Let \( A_{ij} \) be an element of the adjacency matrix of the network, which gives the numbers of links between node \( i \) and \( j \). The modularity is defined to be the fraction of the number of links that fall within the communities minus the expected number of such links. The randomization of the links is done so as to preserve the degree of each node. Thus the expected number of links falling between two node \( i \) and \( j \) following randomization is \( \frac{k_i k_j}{2m} \) where \( k_i \) is the degree of node \( i \), \( k_j \) is the degree of node \( j \), and the actual minus expected number of links between the same two nodes is \( A_{ij} - \frac{k_i k_j}{2m} \). Summing over all pairs of nodes in the same group, the modularity \( (Q) \) is then defined (Newman et al., 2004) by:

\[
Q = \frac{1}{4m} \sum_{ij} \left[ A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),
\]

where \( c_i \) is the group to which node \( i \) belongs, \( c_j \) is the group to which node \( j \) belongs, and \( \delta(c_i, c_j) \) is the Kronecker delta. The value of the modularity lies in the range \([-1,1]\). It is positive if the number of links within groups exceeds the number expected on the
basis of chance.

3.6.3 The Shannon’s entropy of degree distribution

The entropy of degree distribution ($DENT$) is the Shannon’s entropy of the degree distribution in the network. In other words, the entropy of degree distribution measures the complexity of the recurrence rate:

$$DENT = -\sum_{k_{\text{min}}}^{k_{\text{max}}} P(k) \ln P(k)$$

(3.15)

where $P(k)$ is the degree distribution in network described in Section 3.3.1, $k_{\text{min}}$ is the minimum of degree distribution, and $k_{\text{max}}$ is the maximum of degree distribution.

3.7 Summary

This chapter has reconsidered the analysis of nonlinear sensor data from complex systems by means of complex network theory. The recurrence network is applicable to univariate as well as multivariate sensor data with or without embedding. In addition, recurrence networks can be applied for studying sensor data with non-equidistant timescales as well as temporal variations in non-stationary data and allow the construction of simple significance tests with respect to the associated network-theoretic measures. It provides a variety of statistical measures from network theory computed for recurrence network in terms of phase space properties of dynamical systems.

Using statistical characteristics of complex networks such as clustering coefficient, transitivity, the number of communities, the modularity, and the Shannon’s entropy of degree distribution, we are able to extract more features of nonlinear sensor
data. These features can use for constructing the feature vector described in Chapter 5. We would like to underline that due to the duality between the recurrence matrix of a sensor data and the adjacency matrix of the associated recurrence network, RQA might be another promising candidate for this purpose, yielding interesting complementary insights on complex networks in a variety of different situations.
Chapter 4

4. Recurrence based Classification and Estimation Analysis

In Chapters 2 and 3 we summerized the definitions of recurrence analysis and recurrence network approaches. These approaches allow us to extract the features of sensor data from the recurrence quantification analysis. We use these features to classify sensor data via neural network classifier. The proposed feature extraction method has three main steps as shown in Fig. 4.1.

Fig. 4.1. Methodology of Sensor data Classification based on Recurrence Quantification Analysis and Neural Networks
The first step is to transform the sensor data into phase space trajectories and then create the corresponding recurrence plot as described in Sections 2.2 and 2.3. Also the recurrence plot is transformed into the corresponding recurrence network as described in Section 3.4. Each data set is fed to a separate recurrence preprocessing which generates the recurrence plot and the recurrence network based on that individual data set. This step can be viewed as recurrence preprocessing activity prior to feature extraction. The second step computes the statistics of the recurrence plot and the recurrence network through the recurrence quantification analysis as described in Sections 2.4, 3.5 and 3.6. The statistics of recurrence plot and recurrence network contain the characteristic behavior of the sensor data. In other words, a set of recurrence statistics is extracted to form the feature vector. In the last step, the feature vector is processed through the artificial neural networks to classify the sensor data. This methodology is verified by conducting an extensive set of experiments.

4.1 Recurrence based Feature Extraction Method

4.1.1 Recurrence preprocessing (step 1)

According to our scheme shown in Fig. 4.1, recurrence preprocessing plays an important role to convert sensor data into recurrence plot and recurrence network in order to extract their features using recurrence quantification analysis. In this work, the sensor data is preprocessed by transforming it into phase space reconstruction using Equation (2.1) and then creating recurrence plot using Equation (2.4). Usually, a phase space does not have a low enough dimension to be pictured. Higher-dimensional phase space can only be
visualized by projection into either two or three dimensional sub-spaces. However, recurrence plot enables us to investigate the \( m \)-dimensional phase space trajectory through a two-dimensional representation of its recurrences. According to recurrence plots, we investigate the characteristics of sensor data including their features and patterns.

This is the first step to convert \( N \) sensor data into a set of \( N \) recurrence matrices; \( N \) sensor data are preprocessed by transforming them into \( N \) sets of phase space trajectories to visualize them as \( N \) recurrence plots. Also the \( N \) recurrence plots are transformed into \( N \) recurrence networks. The selection of parameters in Equation (2.1) and Equation (2.4) directly affects the performance of the classification accuracy because the recurrence statistics depend on these parameters.

The selection of parameters, the embedded dimension \( m \) and the time delay \( T \), plays an important role in constructing the phase space vectors. However, some studies (Kennel et al., 1992; Fraser et al., 1986; Marwan et al., 2007) have found that it is reasonable to use the embedded dimension \( m = 1 \) and the time delay \( T = 1 \) because these two parameters can be considered to represent the same dynamical system in different coordinate systems. If the classification accuracy is unacceptable, one can reconstruct the phase space trajectories by using estimated parameters as shown in Equations (2.2) and (2.3). The estimation of the smallest sufficient embedding dimension \( m \) has been introduced by using the false nearest-neighbors algorithm (Kennel et al., 1992). Also the mutual information function (Fraser et al., 1986) has been used for selecting an appropriate time delay \( T \).
Recurrences take place in phase space vectors which are used to create the recurrence plots. The selection of the threshold distance $\varepsilon$, which is an essential parameter of a recurrence plot, is an important step. Several studies for the choice of the threshold distance $\varepsilon$ have been investigated (Mindlin et al., 1992; Zbilut et al., 2002; Thiel et al., 2002) but they are not highly successful. The choice depends on the application or problem. A systematic investigation is depending on the application. Based on recurrence statistics, it has been shown that the threshold distance $\varepsilon = 0.1$ is quite reasonable (Marwan et al., 2007). If it doesn’t work well, we have to reconstruct the recurrence plot by trial and error to adjust the value of the threshold distance $\varepsilon$.

The summary of this step is described as follows:

1. Apply the phase space reconstruction using Equation (2.1) to $N$ sensor data individually and obtain the $N$ sets of phase space trajectories, one set for each sensor data.

2. Transform $N$ sets of phase space trajectories into $N$ recurrence matrices using Equation (2.4). Each matrix has a size of $M \times M$, where $M$ is the number of phase space trajectories, it is defined as

$$M = \frac{P}{m}$$

(4.1)

where $P$ is the size of time series and $m$ is an embedded dimension of phase space trajectory.

3. Transform $N$ recurrence matrices into $N$ recurrence networks using Equation (3.7).
4.1.2 Feature extraction (step 2)

When the sensor data is too large to analyze, we revert to feature extraction. Thus the sensor data is transformed into a reduced representation set of features. Transforming the sensor data into the set of recurrence quantification features is called feature extraction. Some specific recurrence quantification features in different sensor data sets are analyzed and investigated. In terms of classification and estimation, we investigate the dynamic and structural features of recurrence quantification analysis based on characteristic of sensor data individually. All features of sensor data that are extracted using recurrence quantification analysis are shown in Fig. 4.2.

After transforming a set of $N$ sensor data into $N$ sets of recurrence matrices and $N$ sets of recurrence networks in step 1, the desired recurrence statistics are extracted through recurrence quantification analysis. These statistics are used to form the feature vector. The details of this step are as described below:

1. Calculate the recurrence statistics of $N$ recurrence matrices using Equations (2.6) - (2.13).
2. Calculate the cluster coefficient ($CC$) of $N$ recurrence networks as described in Section 3.5.2.
3. Calculate the transitivity ($TRAN$) of $N$ recurrence networks as shown in Equation (3.13).
4. Calculate the number of communities ($NoC$) of $N$ recurrence networks as described in Section 3.6.1.
5. Calculate the modularity ($MOD$) of $N$ recurrence networks as shown in Equation
6. Calculate the Shannon’s entropy of degree distribution \((DENT)\) using Equation (3.15).

7. Construct the feature vectors \(F_i\), which contain all calculated values in Steps (1) – (5), defined as:

\[
F_i = (RR, DET, L, L_{max}, ENTR, LAM, TT, V_{max}, CC, TRAN, NoC, MOD, DENT)
\]  

where \(i = (1, 2, ..., N)\) and \(N\) is the total number of all sensor data sets. In other words, each feature vector contains thirteen recurrence statistics.

Fig. 4.2. Feature Extraction using the Recurrence Quantification Analysis
4.1.3 Neural network classifier and estimator (step 3)

Artificial neural networks play an important role as a classifier and estimator of sensor data. A neural network learns the relationship between feature extraction and actual target from a set of representative training examples. Once the neural network learns the mapping between recurrence measurements and actual target, it can compute estimates and classification for any given inputs from feature extractions.

In Fig. 4.3 we show a multilayer neural network used as a classifier and estimator of parameters in this research. This neural network has the ability to “understand” the information captured in feature vectors (inputs) and to estimate a process state or a continuous process variable (output), as long as there exists a continuous relationship between the features vectors and the response variable and a sufficiently rich set of training examples is available. From a practical point of view, however, the interpretation of sensor signals is made easier by their transformation to a set of feature vectors using recurrence quantification analysis. Once the neural network learns the mapping between a set of the feature vectors and the output values, it computes the classification and estimation of output parameters for any sensor data reading.

To solve our criteria at hand, a neural network is created by adaptive gradient descent with momentum training. Besides the training function, hyperbolic tangent sigmoid functions are integrated. In the proposed sensor data classification and estimation scheme, the data from each signal is fed to a separate recurrence preprocessing and neural network which provides the classification and estimation based on that individual sensor data measurements. It provides a more accurate and robust classification and estimation.
4.2 Experiments

We apply our methodology shown in Fig. 4.1 to the classification of sinusoidal and exponential signals. This is an example of the classification of sensor data using the recurrence based feature extraction method. The classification of similarity patterns of sensor data has been verified.
We generate sampled sinusoidal and exponential signals with additive Gaussian noise. Firstly, we generate sinusoidal signals $x(t) = a \sin(\omega t)$ for $a = 1$ and $\omega = 20\pi$. The sampling frequency ($F_s$) and the carrier frequency ($F_C$) are 100 Hz and 10 Hz respectively. The time is running from 0 to 11 seconds in steps of 0.01 seconds. The mean and variance of additive Gaussian noise is 0 and 0.03, i.e., $Z \sim N(0, 0.03)$, respectively. Secondly, we generate exponential signals $x(t) = e^{-at}$ for $a = 2$. The sampling frequency ($F_s$) and the carrier frequency ($F_C$) are 100 Hz and 10 Hz respectively. The time is running from 0 to 11 seconds in steps of 0.01 seconds. The mean and variance of additive Gaussian noise is 0 and 0.03, i.e., $Z \sim N(0, 0.03)$, respectively.

We use two sets of experiments for testing our methodology. The data from Set 1 experiments are used for training. The data in Set 2 experiments are used to study the performance of the classifier from artificial neural networks and support vector machines. The data in Set 2 experiments test the performance of the classifier at the same conditions that are used to train the neural network and support vector machine. Set 1 has 20 data sets including 10 sampled sinusoidal and 10 sampled exponential signals, and Set 2 has 10 data sets including 5 sampled sinusoidal and 5 sampled exponential signals.

4.2.1 Recurrence preprocessing (Step 1)

In step 1, we transform 30 sampled signals into 30 sets of phase space vectors by using the embedded dimension $m = 1$ and the time delay $T = 1$. Each set has 1,024 phase space
vectors. From each set of 1,024 phase space vectors, we create a corresponding recurrence plot by using the threshold distance $\varepsilon = 0.1$ as shown in Fig. 4.4.

<table>
<thead>
<tr>
<th>Sinusoidal Signal</th>
<th>Exponential Signal</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Sinusoidal Signal" /></td>
<td><img src="image2" alt="Exponential Signal" /></td>
</tr>
</tbody>
</table>

**Fig. 4.4.** (a) An example of sampled signals and (b) Transformation of sampled signal into Recurrence Plots

Equations (2.2) and (2.3) show the calculation of appropriate parameters. We use the embedded dimension $m = 1$, the time delay $T = 1$ and the threshold distance $\varepsilon = 0.1$
for testing the performance of classification technique. If the classification accuracy is not met, we would recalculate the embedded dimension $m$, the time delay $T$, and reconstruct the recurrence plot by trial and error adjustment of the threshold distance $\varepsilon$. After generating all recurrence plots, we transform them into the corresponding recurrence networks.

### 4.2.2 Feature extraction (Step 2)

According to section 4.1.2, we calculate the features of each sensor data as shown in Fig. 4.4 which are transformed into recurrence plots and recurrence networks. Based on recurrence quantification analysis, we obtain 8 features of recurrence plots and 5 features of recurrence networks from each sensor data. The difference is the characteristic of recurrence plots and recurrence networks as described in Chapter 2 and 3. The main advantage of recurrence plots and recurrence networks is that they provide useful features even for short and non-stationary data, where other methods fail.

### 4.2.3 Neural network and support vector machine classifier (Step 3)

The comparison of artificial neural networks and support vector machines is shown in Table 4.1. To satisfy our criteria, in this experiment, a neural network is created by adaptive gradient descent with momentum training. Besides that training function, hyperbolic tangent sigmoid functions are integrated.

Table 4.1 gives the accuracy percentage of classification between neural network and support vector machine. Our results show perfect accuracy classification for classifying between sampled sinusoidal and exponential signals.
Table 4.1. Accuracy Classification of sampled sinusoidal and exponential signals with additive white noise

<table>
<thead>
<tr>
<th>Classification Methods</th>
<th>Classification Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neural Network Classifier (NN)</td>
<td>100.00</td>
</tr>
<tr>
<td>Support Vector Machine (SVM)</td>
<td>100.00</td>
</tr>
</tbody>
</table>

4.3 Summary

The recurrence based feature extraction method has been proposed and initially tested. The proposed system can be considered general-purpose in the sense of being able to classify patterns of sensor data using feature extraction of recurrence quantification analysis. This greatly facilitates the recognition of the pattern of sensor data from recurrence plots and recurrence networks.

The results reported in experiment show 100% classification accuracy. This proposed method is useful as a tool for data analysis. For example, we might study characteristics and behavior of sensor signals from machining data, such as acoustic emission signals, and biomedical signals, such as electromyogram and lung sounds. One benefit of recurrence is phase space reconstruction that allows for high-dimensional systems. It is difficult to get information rarely from one-dimension in sensor data.

Finally, we have demonstrated that this proposed method could classify sensor data using artificial neural networks and support vector machines as classifier as well. In
Chapter 6, we study the characteristics and behavior of sensor data for a deeper understanding of the patterns of the sensor data. At present these algorithms convert sensor data into recurrence plots and also transform into recurrence networks. We classify sensor data based on feature extraction of recurrence quantification analysis. Moreover, a natural bridge between recurrence and sensor data has now been built. In Chapter 7, this proposed method has been applied to some practical applications of interest.
Chapter 5

5. The Phase Synchronization Method based Correlation Networks

This chapter provides a brief introduction to the construction of correlation networks using the phase synchronization method. Linear cross-correlation (Hogg et al., 1995) is usually the base line criterion to evaluate the similarity between two linear time series but it may not be the optimal measure of similarity for nonlinear sensor data. Due to the spatial proximity and volume conduction the cross-correlation between the nonlinear time series may yield very high correlation values. It will therefore be very difficult to decide whether the measured correlation is simply. The phase synchronization method (Romano et al., 2005) is usually the base line criterion to evaluate the level of dynamical similarity between two nonlinear time series of the same frequency with identical phase angles with each cycle. The advantage of the phase synchronization method is capable to capture the similarity between two nonlinear time series.

We introduce the novel correlation network using the phase synchronization. The basic approach for constructing the correlation network involves two steps. The first step aims at finding the correlation between each pair of sensor data using the phase synchronization method. The second step applies a criterion to connect the sensor data based on their correlation. The proposed method has the advantage, that it can assess the similarity between multichannel sensor data to a larger extent than linear
measures and is less prone to spurious similarities when the dynamical behavior is considered.

The proposed method constructs the stock correlation network as described in Section 7.3.2. This topic is of huge interest among financial practitioners, since the correlations between the components of a portfolio serve as main inputs in both portfolio optimization and risk management. On top of this, correlation networks are of interest because they can be used in creating a top-level description of the markets as a whole.

5.1 Review of Phase Synchronization Method

The phase synchronization (PS) is the process by which two or more cyclic signals tend to oscillate with a repeating sequence of relative phase angles (Pikovsky et al., 2001). It is usually applied to two signals of the same frequency with identical phase angles with each cycle. One example of phase synchronization of multiple oscillators can be seen in the behavior of Southeast Asian fireflies (Strogatz et al., 2002). At dusk, the flies begin to flash periodically with random phases and a Gaussian distribution of native frequencies. As night falls, the flies, sensitive to one another’s behavior, begin to synchronize their flashing. After some time all the fireflies within a given tree will begin to flash simultaneously in a burst. Thinking of the fireflies as biological oscillators, we can define the phase to be 0° during the flash and ±180° exactly halfway until the next flash. Thus, when they begin to flash in unison, they synchronize in phase.
The concept of recurrence can be used to detect indirectly phase synchronization in a wide class of nonlinear signals and also systems corrupted by noise, where other methods are not so appropriate (Romano et al., 2005). In this chapter, the phase synchronization is applied to study the characterization of sensor signals. Different signals are detected to be phase synchronized when the difference of their respective phase is bounded. The approach to detect phase synchronization is based on the recurrence of the trajectories of nonlinear system in phase space.

Romano et al. (2005) defined the phase $\Phi$ in periodic systems using Equation (2.4). The phase increases by $2\pi$ when the trajectory completes one rotation around the center or, equivalently, when $\|x(t+T) - x(t)\| = 0$ where $T$ is the period. For nonlinear systems, he assigned an increment of $2\pi$ to the phase when $\|x(t+T) - x(t)\| \sim 0$, or when $\|x(t+T) - x(t)\| < \varepsilon$. That means a black points in the recurrence plot at the coordinates $(t, t+\tau)$ can be interpreted as an increment of $2\pi$ for the phase in the $\tau$ time interval. Hence, using the recurrence plot based phase, there exists a $n_i : m_i$ relationship between the phases, where $i$ denotes a time index. This is because the unstable periodic orbits that synchronize have, in general, different periods. However, it is impossible to compute a statistic that quantifies that instantaneous $n_i\Phi_1 = m_i\Phi_2$ relationship between the phases because $n_i$ and $m_i$ depends on time and the recurrence based phase depends on $\varepsilon$ threshold distance. This can be overcome by computing a statistical measure on how often $\Phi_1$ and $\Phi_2$ increase by $2\pi$ or multiples of $2\pi$ within the $\tau$ time interval.
Therefore, for a given system, the probability \( P(\tau) \) that the system returns to the \( \varepsilon \)-neighborhood of a former point \( x_i \) of the trajectory after \( \tau \) time interval is computed. Then the correlation between \( P(\tau) \) allows us to detect and quantify by phase synchronization between the two systems. By applying the sensor data to a recurrence plot, the estimate of this probability \( P(\tau) \) can be written (Romano et al., 2005) as:

\[
P(\tau) = \frac{\sum_{i=1}^{N-\tau} \Theta(\varepsilon - \|\bar{x}_i - \bar{x}_{i+\tau}\|)}{N - \tau}, \text{ where } \tau = 1, 2, \ldots, K \leq N-1
\]  

(5.1)

where \( N \) is the number of measured points \( \bar{x}_i \) in the sensor data, \( \varepsilon \) is a closeness threshold distance, \( \Theta \) is the Heaviside function, \( \|\| \) denoted a suitable norm in the phase space under consideration, and \( K \) is an arbitrary integer less than or equal to \( N-1 \). It is application specific. In our research, we selected \( K = 100 \) because we observed that the time delay does not go more than 100 time steps.

It is an autocorrelation function which can define higher-order correlations between the points of the trajectory in \( \tau \) time interval. We can detect and identify phase synchronization by using the coincidence of the positions of the maxima of \( P(\tau) \) for both the systems. Therefore, a criterion to quantify the phase synchronization is the cross correlation coefficient between \( P_1(\tau) \) and \( P_2(\tau) \) which can be defined as the correlation probability of recurrence (CPR). It is defined (Romano et al., 2005) as follow:

\[
CPR = \frac{\max_{\tau} \{\overline{P_1(\tau)} \overline{P_2(\tau)}\}}{\sigma_1 \sigma_2}, \text{ where } \tau = 1, 2, \ldots, K
\]  

(5.2)
where $\overline{P}_1(\tau)$ and $\overline{P}_2(\tau)$ are mean centered (mean value is subtracted), i.e.,

$$\overline{P}_1(\tau) = P_1(\tau) - \overline{P}_1$$

and

$$\overline{P}_2(\tau) = P_2(\tau) - \overline{P}_2,$$

and $\sigma_1$ and $\sigma_2$ are the standard deviations of $P_1(\tau)$ and $P_2(\tau)$ respectively.

If the two systems are synchronized, the CPR approaches close to one. If they are not synchronized, the CPR will be close to zero and hence one can expect a drift in the probability of recurrences results in low values of CPR.

For example, we generate two sinusoidal signals with length $N = 256$, as shown in Fig. 5.1. Using the phase synchronization method, we calculate the estimate of probability $P_1(\tau)$ on the first sampled signal and $P_2(\tau)$ on the second sampled signal individually as shown in Fig. 5.2. The correlation probability of recurrence (CPR) between these signals is equal to 0.55.

![Fig. 5.1. An example of a sinusoidal signal without noise and a sinusoidal signal with additive noise](image-url)
Fig. 5.2. The correlation probability of recurrence (CPR) between a sinusoidal signal without noise and a sinusoidal signal with additive noise

5.2 Review of an Example Correlation Network

We present stock correlation network as an example network. A stock correlation network is a type of financial network based on stock price correlation used for observing, analyzing and predicting the stock market dynamics. In the last decade, the financial correlation networks have attracted more attention from the research community. A study on company ownership based network showed a power law distribution with majority of companies controlled by small number of people (Mandere et al., 2011). Another study focused on board of directors where the network was created between companies if represented by the same member on board (Mandere et al., 2011). The board membership network resulted in a power law with small number of board members representing large number of companies (Mandere et al., 2011). Several studies
have proposed network based models for studying the stock correlation network
(Mantegna et al., 1999; Vandewalle et al., 2001; Bonnanno et al., 2003; Onnela et al.,
2003). Stock correlation network has proven its efficacy in predicting market movements.
Chakrabortia et al. (2003) showed that the average distance between the stocks can be a
significant indicator of market dynamics. Their work focused on stock market (1985–
1990) that included the stock market crash of 1987 (Black Monday). Andrew et al. (2007)
worked on the network of different hedge funds and observed the patterns before the
August 2007 stock market turbulence.

The basic approach for building the stock correlation network involves two steps.
The first step aims at finding the correlation between each pair of stocks considering their
Corresponding time series. The second step applies a criterion to connect the stocks based
on their correlation. The popular method for connecting two correlated stocks is the
minimum spanning tree method (Vandewalle et al., 2001; Bonnanno et al., 2003;
Mantegna et al., 1999). The other methods are, planar maximally filtered graph
(Tumminello et al., 2005), and winner take all method (Tse et al., 2010). In all three
methods, the procedure for finding correlation between stocks remains the same. The
approaches for constructing a stock correlation network involve the following steps:

1. The time series related to the stock is selected (e.g., daily prices, weekly prices, and
   trading volumes)

2. The cross correlation for each pair of stocks is computed and the cross correlation
   matrix \( [C_{ij}] \) is constructed.
3. Minimum spanning tree (MST) and planar maximally filtered graph (PMFG) method uses a metric distance $d_{ij}$ to establish links between the stocks. The $d_{ij}$ metric is defined (Vandewalle et al., 2001; Bonnanno et al., 2003; Mantegna et al., 1999) as:

$$d_{ij} = \sqrt{2(1 - C_{ij})}, \text{ where } -1 \leq C_{ij} \leq 1$$

(5.3)

The MST and PMFG methods leads to loss of information, i.e., some high correlated nodes are discarded and low correlated nodes are retained because of the topological reduction criteria (Tse et al., 2010). The WTA connection criterion overcomes the drawback of MST and PMFG. In WTA method, the link between two stocks is established based on a threshold $\lambda$: a link exists between node $i$ and $j$, if $C_{ij} > \lambda$, otherwise $i$ and $j$ are not connected.

Tse et al. (2010) showed that for large values of threshold (e.g., 0.7, 0.8, or 0.9) the stock correlation networks exhibit a scale free behavior. For small values of threshold, the network tends to be fully connected and does not exhibit scale free distribution.

5.3 Construction of Correlation Networks using Phase Synchronization Method

The main idea of constructing the phase synchronization based correlation network is as follows: let the set of sensor data represents the set of vertices of the network. A certain threshold $\theta$ is specified such that $-1 \leq \theta \leq 1$ and an undirected edge connecting the vertices $i$ and $j$ if the correlation probability of recurrence ($CPR_{i,j}$) is greater than or equal
to a threshold $\lambda$. Different values of threshold $\lambda$ define the networks with the same set of vertices, but different set of edges. Let the graph $G = (V, E)$ represent the phase synchronization based correlation network, where $V$ and $E$ are the set of vertices edges respectively. $E$ is defined as:

$$ E = \begin{cases} e_{i,j} = 1, & i \neq j \text{ and } CPR_{i,j} \geq \lambda \\ e_{i,j} = 0, & i = j \end{cases} $$ (5.4)

The properties of the phase synchronization of sensor data are dependent upon the choice of the threshold $\lambda$. We generally observe that the total number of connections increases with decreasing $\lambda$, and as $\lambda$ approaches 0, the network becomes fully connected, as expected. The power-law degree distribution holds for large $\lambda$, and becomes unconstructed as $\lambda$ decreases, which is again consistent with the fact that the network becomes connected as $\lambda$ goes down.

It should be noted that we construct networks for different values of threshold $\lambda$ in the range 0 to 1. In other words, the CPR value is subjected to a threshold $\lambda$ in order to establish a link between two nodes. The phase synchronization based correlation matrix is defined as:

$$ A_y = \Theta(\lambda - CPR_y) $$ (5.5)

where $A_y$ is the adjacency matrix; $\Theta$ is the Heaviside step function; and $\lambda$ is a threshold. The links are established between time series $i$ and time series $j$ if the value of $CPR_{ij}$ is greater than or equal to the threshold value $\lambda$. 

64
5.4 Scale-Free Network in the Phase Synchronization based Correlation Network

A scale-free network (Barabasi et al., 1999) is a network whose degree distribution follows a power law, at least asymptotically. The fraction $P(k)$ of nodes in the network has $k$ connections to other nodes goes for large values of $k$ as:

$$P(k) \sim ck^\gamma$$  \hspace{1cm} (5.6)

where $c$ is a normalization constant and $\gamma$ is a parameter whose value is typically in the range $2 < \gamma < 3$, although occasionally it may lie outside these bounds.

5.5 Experiments

We apply our proposed method as described in Section 5.3 to the construction of a correlation network between sinusoidal and exponential signals. This is an example of the construction of correlation network using phase synchronization method. The groups of similarity patterns of sensor data have been verified.

We use two sets of experiments for testing our proposed method. The data from Set 1 experiments are 5 sinusoidal signals, and Set 2 are 5 exponential signals. We generate sampled sinusoidal and exponential signals with additive Gaussian noise. Firstly, we generate sinusoidal signals $x(t) = a \sin(\omega t)$ for $a = 1$ and $\omega = 20\pi$. The sampling frequency ($F_s$) and the carrier frequency ($F_c$) are 100 Hz and 10 Hz respectively. The time is running from 0 to 11 seconds in steps of 0.01 seconds. The mean and variance of additive Gaussian noise is 0 and 0.03, i.e., $Z \sim N(0,0.03)$,
respectively. Secondly, we generate exponential signals \( x(t) = e^{-at} \) for \( a = 2 \). The sampling frequency \( (F_S) \) and the carrier frequency \( (F_C) \) are 100 Hz and 10 Hz respectively. The time is running from 0 to 11 seconds in steps of 0.01 seconds. The mean and variance of additive Gaussian noise is 0 and 0.03, i.e., \( Z \sim N(0,0.03) \), respectively.

The choice of the \( \lambda \) threshold in Equation (5.4) is 0.9 because normally the random sinusoidal signals and exponential signals have totally different patterns. Also we choose the high \( \lambda \) threshold value because the network will separate the groups of similarity signals as shown in Fig. 5.3. The result shows that the similar signals are linked. The correlation network shows two groups of nodes between sinusoidal signals and exponential signals. The details of the sampled signals are shown in Fig. 5.4.

**Fig. 5.3.** The construction of a Phase Synchronization based Correlation Network from 5 sinusoidal signals and 5 exponential signals
Fig. 5.4. The 10 generated sampled signals with additive noise
5.6 Summary

The phase synchronization based correlation network method has been proposed and initially tested. The proposed system can be considered general-purpose in the sense of being able to construct the correlation network of sensor data using phase synchronization method. This greatly assists to recognize the patterns of sensor data from the network.

The results reported in the experiment show the correlation network allows separations of sensor data into groups. Consequently, this proposed method is useful as a tool for network analysis of different systems ranging from sensor signals to financial data (such as stock indices). The phase synchronization method captures the dynamic behavior of sensor data and mitigates the information loss.

Finally, we have demonstrated that this proposed method could construct the correlation network of sensor data using phase synchronization. Moreover, a natural bridge between correlation network and sensor data has now been built. In Chapter 7, this proposed method has been applied to the stock price correlation. This proposed method provides valuable insights into the behavior of highly correlated stocks which can be useful for making trading decisions. The network exhibits a scale-free degree distribution for both chaotic and non-chaotic periods.
Chapter 6

6. Recognition of patterns in Sensor data using Recurrence Network Analysis

In Chapter 3, we introduced a procedure to transform sensor data into complex networks. In this chapter, we review approaches for studying the recognition and measures of sensor data using recurrence network analysis. More importantly, we present the framework of our research experiment. The framework is based on complex network analysis.

6.1 Features of Recurrence Networks

We introduce some interesting features of recurrence networks based on recognition of patterns in sensor data. In particular, the strengths and possible limitations of features are discussed.

6.1.1 Isomorphic Networks

Two networks which contain the same number of network vertices connected in the same way are said to be isomorphic (Weisstein et al., 1969). In graph theory, an isomorphic of graphs $G$ and $H$ is a bijection between the vertex sets of $G$ and $H$, i.e.,

$$f : V(G) \rightarrow V(H)$$  \hspace{1cm} (6.1)
such that any two vertices \( u \) and \( v \) of \( G \) are adjacent in \( G \) if and only if \( f(u) \) and \( f(v) \) are adjacent in \( H \). This kind of bijection is commonly called “edge-preserving bijection”, in accordance with the general notion of isomorphism being a structure-preserving bijection.

### 6.1.2 Network Motifs

Network motifs are patterns (sub-graphs) that recur within network much more often than expected at random. Most networks have been found to show a small set of network motifs. Each type of network seems to display its own set of characteristic motifs. These small circuits can be considered as simple building blocks from which the network is constructed (Shen-Orr et al., 2002). This idea was first presented by Alon et al. (2002) when network motifs were discovered in the gene regulation networks of E. Coli bacteria.

### 6.1.3 Degree Centrality Measurement

Within graph theory and network analysis, there are many measures of the degree centrality of a node within a graph that determines the relative important of a node within a graph.

The degree centrality of a node in a network is the number of connections or edges the node has to other nodes in the network. Degree centrality is based on the idea that important nodes are defined by the number of links incident upon the node. The degree centrality of node \( i \) is defined as (Niemin et al., 1974):
\[ C_i^D = \frac{k_i}{N - 1} = \frac{\sum_{j \in G} a_{ij}}{N - 1} \]  \hspace{1cm} (14)

where \( k_i \) is the degree of node \( i \), \( a_{ij} \) is adjacency matrix, and \( N \) is the number of nodes.

6.2 Experiments

6.2.1 Similarity patterns of sensor data using isomorphism

In the section, we show how to recognize similarity patterns of sensor data using isomorphism within the network. Firstly, we generate random sensor data and transform them into complex networks using recurrence network as described in Chapter 3. Secondly, we use one of local properties in network analysis called Isomorphism to classify sensor data signals within the network.

Figs. 6.1(a) and 6.1(b) illustrate that using amplitude values of sampled process signals are insufficient to Fig. out if the two process signals are the same. For instance, the dotted signal in Fig. 6.1(a) is a time-shifted version of the original signal, but the amplitudes of the two signals at any sampling time instance differ significantly. Similarly, two signals with different frequencies can have approximately equal amplitudes as shown in Fig. 6.1(b). A comparison of such signals based on amplitude values alone would therefore tend to erroneously classify the signals in Fig. 6.1(a) as different and the signals in Fig. 6.1(b) as equal.
Fig. 6.1. (a) identical subsignals but shifted in time. They have completely different amplitudes at any time instance of sampling, (b) Subsignals with different forms but which are approximately equal in amplitude, and (c) Transformation of sampled signals into complex network using Recurrence Networks.

For the purpose of the experiment, we generate sampled subsignals with length $N = 50$, which is assumed to be collected at a sampling rate of one time unit as shown in Fig. 6.1(a) but dotted signals are time-shifted version of the original subsignals with 20
time periods. Furthermore, we transform these signals into complex network using recurrence networks as shown in Fig. 6.1(c). We found that similar subsignals are completely symmetric using Isomorphism. As can be seen from the test pair detected, there can be redundancies in the output of the proposed method, but these do not lead to misleading results.

In order to examine sampled subsignals with different frequencies but equal amplitudes, we generate sampled subsignals with length $N = 50$, which is assumed to be collected at a sampling rate of one time unit as shown in Fig 6.1(b). Also we generate dotted subsignals by time-inverting the original sampled subsignals. We have discovered that similar subsignals are structurally identical or isomorphic.

6.2.2 Superfamily phenomenon of sensor data using network motifs

Although we could recognize similar patterns in sensor data using isomorphism, we still can’t clearly classify patterns of chaotic signals and periodic signals including additive noise. Recently, Small et al. (2008) found that different types of sensor data belong to different superfamilies (that is, the set of networks with the same relative abundance of the different subgraphs), and the fine local structures in the complex network domain reflect the state recurrence properties of this sensor data. In the section, we transform sensor data into recurrence networks and classify superfamily of each sensor data using network motifs. In addition, each pattern is contaminated with an additive white noise. Nonetheless, we choose four of nearest neighbors for network motifs.
(a) Periodic Signals including Additive White Noise

In this section, we attempt to recognize the patterns of sinusoidal signals with additive white noise using network motifs. First of all, we have generated 100 sampled sinusoidal signals with different phase, amplitude, and frequency similar to the one shown in Fig. 6.2(a). Also we have transformed them into complex networks using recurrence as described in Chapter 3. We have found that all sampled signals belong to same superfamilies or subgraphs within the networks as shown in Fig. 6.2(b).

Fig. 6.2. (a) Sampled sinusoidal signal and (b) The pattern of sinusoidal signal using network motifs
We have also found that noisy periodic signals belong to the same subgraphs within the networks. We examine 100 sampled noisy sinusoidal signals with different additive white noise as shown in Fig. 6.3(a). Consequently, we demonstrate that the same type of sensor data belong to the same superfamilies or subgraphs within the networks shown in Fig. 6.3(b).

Fig. 6.3. (a) Sampled noisy sinusoidal signal and (b) The pattern of noisy sinusoidal signal using network motifs
In this experiment, we have found original sampled sinusoidal signals and noisy sampled sinusoidal signals belong to same superfamilies of network motifs. In contrast, we would find new subgraphs if signal-to-noise ratio is higher than 1:1, indicating more signal than noise.

(b) Nonlinear Signals

In the last several decades, many researchers have attempted to recognize the pattern of nonlinear signals. In this section, we introduce a novel methodology of recognizing the pattern of nonlinear signals using network motifs. We examine lung sound signals as nonlinear signals, see Fig. 6.4(a). We transform lung sound signals into complex networks using recurrence networks, and then detect the pattern of subgraphs with size of four nearest neighbors. We have found this chaotic signal belong to same superfamilies within networks as shown in Fig 6.4(b).
6.3 Summary

A method for classifying the sensor data using recurrence network analysis has been proposed. The proposed system is considered general-purpose in the sense of being able to classify arbitrary patterns of sensor data using isomorphism and network motifs. This greatly assists in recognizing the pattern of sensor data from recurrence networks.

Consequently, these methods are useful as a tool for data analysis. For example, we might study characteristic and behavior of each sensor data signal from complex networks perspective. One benefit of recurrence network is phase space reconstruction that allows for high-dimensional systems. It is rare and difficult to get information from one-dimension.

Finally, we have demonstrated that these methods could classify sensor data using recurrence network analysis. At present these algorithms convert sensor data into recurrence networks. In contrast, we classified sensor data using network motifs. Moreover, a natural bridge between network theory and sensor data has now been built using the properties of complex networks that allows us to analyze sensor data.
Chapter 7

7. Applications

The high potential for the analysis and classification using recurrence based techniques arise with their applicability. Hundreds of applications of nonlinear sensor data, especially physiological signals and machining signals, are waiting us to solve their mysteries. In this chapter selected applications of the physiological signals and acoustic emission signals are presented. Methods of linear data analysis mostly fail in these applications because of the rather short length of sensor data and their nonstationary. Except for the example in the section 7.4 all results of these applications are already published (Sultornsanee et al., 2011a; Sultornsanee et al., 2011b; Sultornsanee et al. 2011c).

7.1 Lung Sounds Classification via Recurrence Quantification Analysis

Respiratory diseases are a major cause of sickness throughout the world. Therefore, the study and classification of lung sounds have attracted attention over the years. There exist multiple classification methods such as multivariate linear autoregressive model (Hernandez et al., 2005), wavelet coefficients (Kandaswamy et al., 2004), and combined neural network and genetic algorithm (Guler et al., 2005). However, many of these current methods of time series analysis do not rely on the characteristics of lung sound signals because of their complexity, nonlinearity, and non-stationarity. We
introduce a novel method of analysis of lung sound signals using recurrence quantification analysis. Lung sound signals are transformed into recurrence plots and a set of statistical features are extracted using recurrence quantification analysis. Examining the acoustic patterns in lung sounds, they are classified into one of the five categories: normal sounds of inspiratory, normal sounds of expiratory, crackles, rhonchus, and wheezes. The results show that the proposed method is successful with 100% classification accuracy. The accurate results indicate that the proposed lung sounds classification method is very effective for real world lung sound classification applications.

7.1.1 Review of lung sounds classification

Respiratory sounds contain significant information on physiology and pathology of the lungs and the airflow. The characteristic of lung sounds without adventitious sound components (crackles, rhonchus, and wheezes) may reflect airway dimension and pathologic changes in the pulmonary tissue. Characteristics of crackles are significantly different in pulmonary disorders. Also a wheeze may have acoustic features indicating not only the presence of abnormality in the respiratory system but also the severity of airway obstructions most frequently found in asthma and large-airways stenosis. Rhonchus can be found in patients with secretions or narrowing in large airways and with abnormal airway collapsibility.

Early diagnosis of respiratory disorders is important for medical treatment. Basic diagnostic methods of respiratory diseases are chest X-rays and computerized tomography (CT). While these methods provide a clear picture of lungs and airways, they
are expensive and expose patients to potentially harmful doses of X-ray radiation. Also radiation put dialysis patient at increased risk of cancer.

There exist various classification methods such as multivariate linear autoregressive model (Hernandez et al., 2005), wavelet coefficients (Kandaswamy et al., 2004), and combined neural network and genetic algorithm (Guler et al., 2005). They can capture the patterns of lung sounds. However, many of these current methods do not detect the characteristics of lung sound signals because of their complexity, nonlinearity, and non-stationarity.

In our study, the signals are sampled at 2 kHz for a length of 1800 data points. Each signal belongs to one of five different classes. These classes are inspiratory phase of bronchial sound (class 1), expiratory phase of bronchial sound (class 2), sound with crackles (class 3), sound with rhonchus (class 4), and sound with wheezes (class 5) as shown in Fig. 7.1. The first two of these classes are normal lung sounds. The other three classes illustrate abnormal lung sounds.

(a) Normal inspiratory phase of bronchial sound signal
(b) Normal expiratory phase of bronchial sound signal

(c) Crackles signal

(d) Rhonchi signal
7.1.2 A recurrence based feature extraction method for lung sounds classification

The details of a recurrence based feature extraction method were described in Chapter 4. The proposed feature extraction method for lung sounds classification has three main steps as shown in Fig. 7.2. The first step is to transform the lung sounds into phase space trajectories and then create the corresponding recurrence plot. Each data set is fed to a separate recurrence preprocessing which generates the recurrence plot based on that individual data set. This step can be viewed as recurrence preprocessing activity prior to feature extraction.

The second step computes the statistics of the recurrence plot through recurrence quantification analysis. The statistics of recurrence plot contain the characteristic behavior of the lung sounds. In other words, a set of recurrence statistics is extracted to form the feature vector. In the last step, the feature vector is processed through a
machine learning technique such as regression trees, multivariate adaptive regression splines or neural networks to classify the lung sounds.

Fig. 7.2. The steps of the recurrence based feature extraction method for lung sounds
**Recurrence Preprocessing for Lung Sounds (Step 1)**

We transform 90 lung sound signals into 90 sets of phase space vectors by using the embedded dimension $m = 1$ and the time delay $T = 1$. Each set has 1,800 phase space vectors. From each set of 1,800 phase space vectors, we create a corresponding recurrence plot by using the threshold distance $\varepsilon = 0.1$ as shown in Fig. 7.3. We use the embedded dimension $m = 1$, the time delay $T = 1$ and the threshold distance $\varepsilon = 0.1$ for testing the performance of classification technique. If the classification accuracy is unacceptable, one can reconstruct the phase space trajectories by using estimated parameters as shown in Equations (2.2) and (2.3).

For all recurrence plots, we conclude that lung sound signals obey a certain nonlinear and non-stationary behavior (see Fig. 7.3) because different categories of lung sounds show the recurrence points in different positions and states.

**Feature Extraction from Lung Sounds (Step 2)**

We transform a set of 90 lung sound signals into 90 sets of recurrence matrices. We use these matrices to calculate the eight recurrence statistics given by Equations (2.6) – (2.13) and form the feature vectors of size eight. However, we tried to reduce the dimensional feature vectors to five by using box plots for each feature as shown in Fig. 7.4. The dimension reduction might cause some information loss but hopefully retain the most important variance. The comparison results are shown in Table 7.1.
Fig. 7.3. Examples of the calculated recurrence plots; (a) normal inspiratory phase of bronchial sound, (b) normal expiratory phase of bronchial sound, (c) crackle, (d) rhonchi, and (e) wheeze
The feature selection criterion is essentially based on the distribution of the features across five classes of lung sounds. Each box plot illustrates the location and variation changes between different groups of data. The reduced dimensional features vector has five recurrence statistics: determinism ($DET$), mean diagonal line length ($L$), longest diagonal line ($L_{max}$), Entropy ($ENTR$), and longest vertical line ($V_{max}$).

![Box plots of different recurrence statistics](image)

(a) Recurrence Rate (RR)  
(b) Determinism (DET)  
(c) Laminarity (LAM)  
(d) Mean Diagonal Line

**Fig. 7.4.** The distribution of training data based on five categories of lung sounds
Fig. 7.4. The distribution of training data based on five categories of lung sounds

Classification Techniques (Step 3)

We use the features vectors calculated in Step 2 to classify lung sounds. The classification accuracy of our proposed method is very high (see Table 7.1). We tried different classifiers, namely, multilayer perceptron (MLP), support vector machine (SVM), interaction trees (CART), and multivariate adaptive regression splines (MARS) to classify lung sounds. We test our feature vectors with different classifiers because we prove that our feature extraction method can work well with every classifier.
7.1.3 Experiments and results

In our experiments, the data is randomly split into two sets. The data from Set 1 are used as the training set. The data from Set 2 are used to study the classification accuracy. The training data set (Set 1) has 90 different signals from different classes. The class breakdown of the signals is as follows: 12 signals each from class 1, 12 from class 2, 26 from class 3, 18 from class 4, and 22 from class 5. The testing data set (set 2) has 40 different signals from different classes. The class breakdown of the signals is as follows: 6 signals each from class 1, 6 from class 2, 13 from class 3, 8 from class 4, and 7 from class 5.

Table 7.1 shows the comparison of all experimental results. As shown, the features vector of size eight gave the best classification results. It indicates that a feature vector of size eight is more accurate and consistent than the reduced dimensional feature vector of size five. The results of the proposed method are compared to benchmark methods that are known to give the best performance on this lung sound signal application (Hernandez et al., 2005; Kandaswamy et al., 2004; Guler et al., 2005) (see Fig. 7.5).

**TABLE 7.1. COMPARISON OF ALL EXPERIMENTAL RESULTS**

| Classification Techniques | Feature Vector of size 8 | Feature Vector of size 5  
(The reduced dimensional feature vector) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Multilayer perceptron (MLP)</td>
<td>99.67%</td>
<td>93.07%</td>
</tr>
<tr>
<td>Support Vector Machine (SVM)</td>
<td>100.00%</td>
<td>99.23%</td>
</tr>
<tr>
<td>Interaction Trees (CART)</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
<tr>
<td>Multivariate Adaptive Regression Splines (MARS)</td>
<td>100%</td>
<td>99.23%</td>
</tr>
</tbody>
</table>
The proposed feature extraction method is verified by the classification accuracy of lung sounds. The comparison of the performance of different classification techniques for lung sounds is shown in Fig. 7.5.

![Graph showing comparison of accuracy performance of the classification of lung sounds](image)

**Fig. 7.5.** Comparison of accuracy performance of the classification of lung sounds

The results indicate that the feature vector of size eight is reasonable for lung sounds application. We only use the common parameters to construct the recurrence plots. These common parameters are reasonable for recognizing the patterns of lung sounds. In other words, the proposed method is very effective with the pattern recognition. The features vector contains significant information to categorize the five different classes of lung sounds.
7.2 Classification of Electromyogram using Recurrence Quantification Analysis

Clinical analysis of the electromyogram is a powerful tool for diagnosis of neuromuscular diseases. Therefore, the classification of electromyogram signals has attracted much attention over the years. Several classification methods based on techniques such as neuro-fuzzy systems (Sabri et al., 2010), wavelet coefficients (Subasi et al., 2006), and artificial neural networks (Nihal et al., 2005) have been investigated for electromyogram signal classification. However, many of these time series analysis methods are not highly successful in classification of electromyography signals due to their complexity and non-stationarity. We introduce a novel approach for the diagnosis of neuromuscular disorders using recurrence quantification analysis and support vector machines.

Electromyogram signals are transformed into recurrence plots and a set of statistical features are extracted using recurrence quantification analysis. Support vector machine employing radial basis functions is used for classifying the normal and abnormal of neuromuscular disorders. Examining the acoustic patterns in electromyogram, we classify the signals into one of the three categories: healthy, neuropathy, and myopathy. The results show that the proposed method classifies these signals with 98.28% accuracy; it is a significantly better accuracy than what has been reported in the literature thus far. The accurate results indicate that proposed diagnosis method of neuromuscular disorders is very effective.
7.2.1 Review of electromyogram classification

There are more than 100 neuromuscular disorders that affect the brain, spinal cord, nerves, and muscles (Anand et al., 2007). Neuromuscular disorders are related to pathological changes in the structure of motor units and can be divided into muscular (myopathy) and neuronal disorders (neuropathy). Myopathy is neuromuscular disorders in which the primary symptom is muscle weakness due to dysfunction of muscle fibres (Anand et al., 2007). Neuropathy describes damage to the peripheral nervous system which transmits information from the brain and spinal cord to every other part of the body. Myopathy is neuromuscular disorders in which the primary symptom is muscle weakness due to dysfunction of muscle fibres (Anand et al., 2007).

Motor unit morphology can be studied by recording its electrical activity which is electromyogram (EMG). In clinical EMG, motor unit potential (MUPs) are recorded using a needle electrode at slight voluntary contraction (Stalberg et al., 1996). A MUP reflects the electrical activity of a single anatomical motor unit. Moreover, they are used to detect and describe different neuromuscular diseases. Therefore, intramuscular electromyogram is commonly used as a diagnostic tool in clinical practice.

Modern electromyogram analysis has been used for some decades. It is focused on signal processing and graphical representation of the signals (Nihal et al., 2005; Subasi et al., 2006). In the last decade, many applications were introduced. Nihal et al. (2005) implemented an artificial neural network to diagnose neuromuscular disorders. His results showed 91.6 - 94.4% classification accuracy. Subasi et al. (2006) introduced a wavelet coefficient method to classify the EMG signals. Sabri et al. (2010) introduced a
neuro-fuzzy system to classify EMG signals. Its performance showed 83.3 – 90% classification accuracy.

Analyzing nonlinear time series through recurrence quantification analysis has been investigated for many years (Marwan et al., 2002; Marwan et al., 2007). To test whether electromyogram is a nonlinear signal or just random noise, we analyze EMG signals of neuromuscular diseases using recurrence plot (see Fig. 7.6). From all the results, we conclude that EMG obeys a certain nonlinear deterministic law and non-stationarity is significant within these signals. It is therefore possible to study and analyze these signals as a non-linear system using recurrence quantification analysis.

Fig. 7.6. Electromyogram signals of neuromuscular diseases
In this section, we introduce a novel diagnosis and classification method for neuromuscular disorders; this method uses recurrence quantification analysis and a support vector machine. Our proposed method is evaluated using electromyogram signals corresponding to three different neuromuscular disorders: healthy, neuropathy, and myopathy. In order to extract their features, we apply recurrence quantification analysis as a means for feature extraction. The classification of neuromuscular disorders is obtained from support vector machine learning. Furthermore, electromyogram signals of myopathy, neuropathy, and healthy subjects are used for analyzing the diagnostic performance of neuromuscular disorders.

7.2.2 A recurrence based feature extraction method for electromyogram classification

We introduce the feature extraction based on recurrence quantification analysis in combination with the support vector machine classifier. The proposed Electromyogram classification method is shown in Fig. 7.7. This approach allows us to extract the features of electromyogram signals from recurrence quantification analysis. Also we use these features to classify the type of neuromuscular disorders via support vector machines.

Recurrence preprocessing (step 1)

In proposed method, recurrence preprocessing plays an important role to convert the electromyogram signal into a recurrence plot in order to extract their features using recurrence quantification analysis. The electromyogram signals are preprocessed by transforming it into phase space trajectories and then creating the recurrence plots as described in Chapter 4. Usually, a phase space does not have low enough dimensions to
be visualized in three-dimensional space. Higher-dimensional phase space can only be visualized by projecting it into two- or three-dimensional sub-spaces. However, the recurrence plot enables us to investigate the $m$-dimensional phase space trajectory through a two dimensional representation of its recurrences. According to recurrence plots, we investigate the characteristics of EMG signals including their features and patterns. This preprocessing improves the classification accuracy of neuromuscular disorders.

Fig. 7.7. The steps in the proposed method for classification of EMG signals
Feature Extraction (step 2)

In this step, the three features (Recurrence Rate (RR), Determinism (DET), and Laminarity (LAM)) of the recurrence plots are extracted to form an input vector. These extracted features have been chosen by examining their ability to differentiate patterns of different neuromuscular disorders using the box plots for each feature in every class as shown in Fig. 7.8.

![Box plots for RR, DET, and LAM](image)

(a) Recurrence Rate (RR)  (b) Determinism (DET)  (c) Laminarity (LAM)

**Fig. 7.8.** The distribution of EMG based on three classes of neuromuscular disorder

Support Vector Machine Classifier (step 3)

Support vector machine (SVM) plays an important role as classifier of EMG signals. It learns the relationship between features and the actual target from a set of representative training samples. Once the support vector machine learns the mapping between recurrence measurements and the actual target, it can classify neuromuscular disorders through feature extractions.
We create a support vector machine by employing a radial basis function (RBF) as a kernel function. In the proposed classification scheme, the data is fed to a separate recurrence preprocessing which provides the classification based on that individual data measurement. It provides a highly accurate and robust classification performance.

### 7.2.3 Experiments and results

We conduct two sets of experiments to test the accuracy of the proposed method. The data from Set 1 experiments is used for training the support vector machine by using the extracted features from the recurrence quantification analysis. The data from Set 2 experiments is used to study the classification performance of the support vector machine at the same conditions that were used to train the support vector machine.

Goldberger et al. (2000) collected the data with a Medelec Synergy N2 EMG Monitoring System. A 25mm concentric needle electrode was placed into the tibialis anterior muscle of each subject. The signals are recorded at 50 KHz and then downsampled to 4 KHz. During the recording process two analog filters are used: a 20 Hz high-pass filter and a 5K Hz low-pass filter. Each of these signals belongs to one of three different classes: healthy (class 1), neuropathy (class 2), and myopathy (class 3). The training data set (set 1) has 192 EMG signals (each signal is defined by 512 points) including 29 healthy, 88 neuropathy, and 75 myopathy; and the testing data set (set 2) has 100 EMG signals including 15 healthy, 50 neuropathy, and 35 myopathy. For each data set, we extract the three features from the electromyogram.

We use cross-validation to find the best parameters for the support vector machines. The results confirm that the classification accuracy of the SVM with radial
basis function kernel function is 98.28%. The classification efficiency which is defined as the percentage ratio of the number of electromyogram correctly classified to the total number of electromyogram observed for classification is shown in Table 7.2.

**Table 7.2.** The classification accuracy of neuromuscular disorders in our methodology

<table>
<thead>
<tr>
<th>Class</th>
<th>Training set (Set 1)</th>
<th>Test set (Set 2)</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Healthy</td>
<td>100.00%</td>
<td>93.33%</td>
<td>97.72%</td>
</tr>
<tr>
<td>Neuropathy</td>
<td>98.86%</td>
<td>94.00%</td>
<td>97.10%</td>
</tr>
<tr>
<td>Myopathy</td>
<td>100.00%</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
<tr>
<td>Overall</td>
<td>99.47%</td>
<td>96.00%</td>
<td>98.28%</td>
</tr>
</tbody>
</table>

Fig. 7.9 shows the position of the neuromuscular phase in the feature space (DET, LAM, RR). As one can notice, the clusters relative to all classes are separated, and the healthy class is positioned in the central part of the diagram. It is worth noticing that, from the clinical point of view, neuropathy and myopathy are separated.

The results show that the proposed method offers very high classification accuracy. It should be noted that our data may contain healthy or unhealthy acoustic signals but altogether are completely classified by our framework. Consequently, the proposed method performs the classification task ideally even though the data might contain turbulence noise.
Fig. 7.9. Position of the neuromuscular phases in the \((DET, LAM, RR)\) feature space
While other approaches (Sabri et al., 2010; Subasi et al., 2006; Nihal et al., 2005) have produced classification accuracy of 86 to 94.4%, our method shows better classification accuracy. The comparison of accuracy performance of the classification of neuromuscular disorders in testing data set is shown in Fig. 7.10.

**Fig. 7.10.** Comparison of accuracy performance of the classification of neuromuscular disorders
7.3 Phase Synchronization Approach to Stock Correlation Network

Stock correlation network, a subset of financial network, is built on the stock price correlation. It is used for observing, analyzing and predicting the stock market dynamics. We present a new method, which we call phase synchronization (PS) for constructing and analyzing the stock correlation network. The PS method captures the dynamic behavior of the time series of stocks and mitigates the information loss. To test the proposed PS method we use the weekly closing stock prices of the S&P index (439 stocks) from 2000-2009. The PS method provides valuable insights into the behavior of highly correlated stocks which can be useful for making trading decisions. The network exhibits a scale free degree distribution for both chaotic and non-chaotic periods.

7.3.1 Review of stock correlation network

In the last decade, financial networks have attracted more attention from the research community. The stock correlation network is a subset of financial network which constructs a stock network based on correlation between stock entities (prices, trading volume and so on). The application of financial networks include examples such as company ownership based network, board of directors network that showed a power law distribution, i.e., majority of companies were controlled by small number of people and small number of board members (Mandere et al., 2011).

Several studies have proposed network based models for studying the stock correlation network (Mantegna et al., 1999; Vandewalle et al., 2001; Bonnanno et al.,
Stock correlation network has been successful in predicting market movements. A study showed that the average distance between the stocks can be a significant indicator of market dynamics (Chakrabortia et al., 2003). Andrew et al. (2007) analyzed the hedge funds based network (before the August 2007 stock market turbulence).

**Existing methods for the construction of stock correlation network**

The popular approach for constructing a stock correlation network involves the following steps:

1. The time series related to the stock is selected (e.g., daily prices, weekly prices, and trading volumes)
2. The cross correlation for each pair of stocks is computed and the cross correlation matrix \([ C_{ij} ]\) is constructed.
3. Minimum spanning tree (MST) and planar maximally filtered graph (PMFG) method uses a metric distance \(d_y\) to establish links between the stocks. The \(d_y\) metric is defined as

\[
d_y = \sqrt{2(1 - C_{ij})} \tag{7.1}
\]

The minimum spanning tree and planar maximally filtered graph methods leads to loss of information, i.e., some high correlated nodes are discarded and low correlated nodes are retained because of the topological reduction criteria (Tse et al., 2010). The winner-take-all connection criterion overcomes the drawback of minimum spanning tree and planar...
maximally filtered graph. In WTA method, the link between two stocks is established based on a threshold \( \lambda \). Let the graph \( G = (V, E) \) represent the correlation network, where \( V \) and \( E \) are the set of vertices edges respectively. \( E \) is defined as:

\[
E = \begin{cases} 
    e_{i,j} = 1 & \text{and } C_{ij} > \lambda \\
    e_{i,j} = 0 & \text{otherwise}
\end{cases}
\] (7.2)

Tse et al. (2010) showed that for large values of threshold (e.g., 0.7, 0.8, or 0.9) the stock correlation networks exhibit a scale free behavior. For small values of threshold, the network tends to be fully connected and does not exhibit scale free distribution.

The cross correlation computation, when used for a non-linear system such as stock prices, fails to capture the dynamic behavior embedded in the time series. We present a new method, namely, phase synchronization (PS) method for constructing and analyzing the stock correlation network. We transform stock prices to a recurrence plot and then compute the cross correlation coefficient between two phase trajectories to quantify the phase synchronization. The cross correlation coefficient computed is then subjected to a cutoff threshold value (\( \varepsilon_{\text{cutoff}} \)) in order to establish links between pairs of stocks. The PS method captures the dynamic behavior of the time series and mitigates the information loss.

### 7.3.2 The phase synchronization method based stock correlation network

Two systems are phase synchronized when the difference of their respective phase is bounded. The approach to detect phase synchronization is based on the recurrence of the
trajectories of nonlinear system in phase space. The main step in the phase
synchronization method is the calculation of the \( N \times N \) matrix:

\[
R_{i,j} (\varepsilon) = \theta(\varepsilon - \bar{x}_i - \bar{x}_j), \quad i,j = 1,\ldots,N
\]

(7.3)

where \( N \) is the number of measured points \( \bar{x} \); \( \varepsilon \) is a threshold distance; \( \theta \) is the
Heaviside step function; and \( \| \| \) denotes a suitable norm in the phase space considered.

\( P(\tau) \) is the probability that the system returns to the \( \varepsilon \)-neighborhood of a former point
\( x_i \) of the trajectory after \( \tau \) time interval. One can detect and quantify the phase
synchronization by comparing \( P(\tau) \) for both the systems which are defined (Romano et
al., 2005) as follows:

\[
P(\tau) = \frac{\sum_{i=1}^{N-\tau} \theta(\varepsilon - \bar{x}_i - \bar{x}_{i+\tau})}{N - \tau}
\]

(7.4)

The above equation is an autocorrelation function which can define higher-order
correlations between the points of the trajectory in time interval (Romano et al., 2005).

One can detect and identify the phase synchronization by using the coincidence of the
positions of the maxima of \( P(\tau) \) for two systems. Therefore, the cross correlation
coefficient between \( P_l(\tau) \) and \( P_m(\tau) \) is defined as follows (Romano et al., 2005):

\[
CPR_{lm} = \frac{P_l(\tau)P_m(\tau)}{\sigma_l \sigma_m}
\]

(7.5)

where \( l \) and \( m \) represents two stocks (daily prices, trading volumes, or weekly prices)
under consideration, \( P_{l,m}(\tau) \) means that the mean value has been subtracted and \( \sigma_l \) and
\( \sigma_m \) are the standard deviations of \( P_l(\tau) \) and \( P_m(\tau) \) respectively. If both systems are phase synchronized, the probability of recurrence is maximal at the same time and CPR \( \approx 1 \). The CPR value is subjected to a cutoff threshold value in order to establish a link between the two nodes.

\[
A_{lm} = \theta(\varepsilon_{\text{cutoff}} - CPR_{lm})
\]

(7.6)

where \( A_{lm} \) is the adjacency matrix; \( \theta \) is the Heaviside step function; and \( \varepsilon_{\text{cutoff}} \) is the cutoff threshold. The links are established between stock \( l \) and stock \( m \) (prices or trading volumes) if the value of \( CPR_{lm} \) is greater than or equal to the cutoff threshold value.

### 7.3.3 Experiments and results

To test the proposed PS method we use the weekly closing stock prices of the S&P index (439 stocks) from 2000-2009. For the recurrence plot, we use a threshold value \( \varepsilon = 0.1 \). We construct the network with a cutoff threshold value \( \varepsilon_{\text{cutoff}} \) between 0.5 and 1. The stock prices data were grouped into five groups of two years (2000, 2001; 2002, 2003; 2004, 2005; 2006, 2007; and 2008, 2009). The years 2002, 2003 and 2008, 2009 witnessed a chaotic market condition. All other groups witnessed a non chaotic market condition.

Fig. 7.11 shows the network parameters of the stock correlation network constructed using the phase synchronization method. The number of links in case of the chaotic period is lower than the non chaotic period. This observation indicates that many stocks in the chaotic period did not move in the same direction.
Fig. 7.11. Network parameters of the stock correlation network constructed using Phase Synchronization method for years 2000-2009
A power law degree distribution is observed for both chaotic and non-chaotic periods. The power law degree distribution was disrupted at cutoff threshold value of 0.5 only in case of a chaotic period. Hence the stock correlation network constructed using phase synchronization method exhibits a scale free network structure for cutoff threshold value from 0.6 to 0.9 for both chaotic and non-chaotic market conditions. However the exponent value of the power law degree distribution is observed to be lower in case of chaotic period compared to non-chaotic period. The exponent values obtained are negative. For the sake of discussion we do not consider the negative sign and only focus on the absolute value. The average degree in case of a chaotic period is observed to be lower than a non-chaotic period indicating that many stocks did not move in the same direction during the chaotic period. The average shortest path is observed to be longer in case of a chaotic period. The average clustering coefficient is observed to be smaller in case of a chaotic period than in case of a non-chaotic period.

7.4 Recurrence based Approach to Estimate Surface Roughness in turning processes

Surface roughness is one of the important factors in all areas of tribology and in evaluation of the quality of machining operations such as turning, milling, or grinding. This research describes a method for in-process estimation of surface roughness of the workpiece in a turning process from acoustic emission signals generated by sliding friction between a graphite probe and the workpiece. Acoustic emission signals are transformed into recurrence plots and a set of recurrence statistics are computed using the recurrence quantification analysis. The surface roughness parameters are estimated
using an artificial neural network, taking the recurrence statistics of the acoustic
emission signals as inputs.

This method is verified by conducting an extensive set of experiments on AISI
1054 steel workpiece and K420 grade uncoated carbon inserts. We consider three
surface roughness parameters for estimation, namely arithmetic mean ($R_a$), maximum
peak-to-valley roughness ($R_{\text{max}}$), and mean roughness depth ($R_z$). The estimation
accuracy of the proposed method is in the range of 90.13% to 91.26%. The merit of the
proposed method lies in its ability to overcome the limitations of the existing optical
and induction-based techniques which are severely affected by the rotation of the
workpiece, presence of coolants, and interference of chips. Furthermore, these accurate
results indicate that the proposed surface roughness estimation method is very effective
and amenable for practical implementation.

7.4.1 Review of surface roughness estimation

The major interest in finish-machining processes, such as turning, milling, or grinding,
aside from dimensional accuracy and tool life is surface finish. Surface roughness is a
measure of the topographic relief of a surface, which is measured for determining the
quality of finished products. The changes of workpiece surface roughness caused by the
cutting process and the increases of tool wear are measured. In the early 1930s, surface
roughness measurement was made quantitative by the appearance of the stylus
instruments that traversed a surface to be measured with a diamond stylus (Dong et al.,
1992; Dong et al., 1993; Dong et al., 1994).
Surface texture is usually measured by a profilometer which draws the diamond stylus over a sample length of the surface of the component. The currently existing stylus profilometer as well as optical and ultrasonic techniques are useful for surface inspection at a post-process stage only. However, such an inspection does not guard against the production of non-conforming parts which have to be scrapped. In-process assessment of workpiece surface roughness is an essential requirement for implementing adaptive control schemes to maintain consistency in the surface finish of machined parts. Such an adaptive control scheme for a turning process would adjust the cutting speed and feed or change the cutting tool on-line to maintain the consistency of surface finish at a desired level.

In the past few years, the generally accepted surface roughness measures, namely arithmetic mean ($R_a$), maximum peak-to-valley roughness ($R_{\text{max}}$), and mean roughness depth ($R_z$) have been studied (Susic et al., 1995; Diniz et al., 1992; Bonifacio et al., 1994; Jang et al., 1996). The arithmetic mean ($R_a$) roughness value is obtained by measuring the mean deviation of the peaks from the center line of a surface trace, the center line being established as a line above and below which there is an equal area between the center line and the surface trace. The maximum peak-to-valley roughness ($R_{\text{max}}$) is the distance between highest peak and the deepest valley within the sampling length. The mean roughness depth ($R_z$) is the mean of 5 maximum peak-to-valley roughness depths in 5 successive sampling lengths.

These three parameters are the important surface roughness characteristics and profiles. In Great Britain, the center line average (CLA) value of the surface is used in
place of the RMS value. This is now also the standard in the USA and is called arithmetic average (AA) or arithmetic mean ($R_a$) roughness. In Europe and Japan, a similar development took place but the characteristic maximum peak-to-valley roughness ($R_{\text{max}}$) is used in place of an RMS value.

Some previous research has investigated the use of acoustic emission signals to measure the surface roughness of turned parts. Susic et al. (1995) observed that acoustic emission signals generated in friction processes are correlated with the workpiece surface roughness. Diniz et al. (1992) studied acoustic emission signals in the context of turning. They showed that there exists a relationship between certain statistical parameters of acoustic emission signals and the surface roughness parameters. It has been shown in turning processes that root mean square values of vibration signals and the surface roughness parameters ($R_a$ and $R_{\text{max}}$) show similar behavior, particularly when the cutting tool is worn out (Bonifacio et al., 1994). Jang et al. (1996) developed an algorithm for the estimation of $R_{\text{max}}$ on a turned workpiece utilizing relative cutting vibrations between the tool and the workpiece. This algorithm computed values of $R_{\text{max}}$ for experimental measurements fairly accurately. However, the technique is limited to ferrous materials, and the loose metal particles that get into the space between the inductance pickup and the workpiece surface could cause unexpected errors in measurements.

Surface roughness estimation is also required to monitor the quality of the finished parts. In commercial precise instruments for surface roughness measurements, tactile sensors composed of a hard stylus and a displacement transducer are usually applied for recording the surface roughness of a workpiece. However, surface roughness
is then commonly measured off-line by using a standard stylus type of instrument. This procedure is not practical for on-line or in-process measurement because the workpiece has to be removed from the lathe and measurements are to be obtained after the machining is finished. To affect some time saving, portable instruments are used which permit the inspection of the workpiece without altering the setup. Still the machine has to be stopped and the tolls have to be cleared before any measurements are carried out. Therefore, the idea of in-process estimation of surface roughness is worth investigating.

There are numbers of different conditions that lead to the installation of a monitoring system in an industrial manufacturing process. Modern manufacturing equipment has to be more and more flexible and has to run automatically. The machine tools have to work free of errors in order to insure economic saving. However, the improvement of the process control is required to save costs by reducing the non-productive time of expensive machine tool and workpiece.

Furthermore, in-process and on-line surface roughness measuring instruments have to be rapid, non-destructive, and simple to operate in a shop or manufacturing environment. A sensor is one of the most important measuring instruments for in-process monitoring. The sensor output should be compatible with accepted metal roughness over an area of the surface. The measurements should occur in real time and not possibly be affected by surface contaminants (such as debris, coolants, and lubricants) and machine vibration. The best sensing measurements come from sensors that contact directly, as much as possible, the surface without any disturbances.

In this research, we use the acoustic emission signal generated by the sliding friction at the point of contact between a graphite probe (sensor) and the rotating
workpiece to estimate the surface roughness parameters. Acoustic emission is the transient elastic energy spontaneously released in materials undergoing deformation, fracture, or both. Acoustic emission refers to the elastic stress waves generated as a result of the rapid release of strain energy within a material due to a rearrangement of its internal structure.

We investigate a novel feature extraction method assessing the workpiece surface quality in real-time. We introduce the recurrence based feature extraction method for surface roughness estimation in finish turning processes. We study and analyze acoustic emission signals generated by a complex friction process using recurrence plots. In our experiments, we acquire acoustic emission signals at several time instances during the turning process through an acoustic emission sensor, which is mounted without obstructing the cutting process. When the turning process is stopped, we measure three parameters, $R_a$, $R_{max}$, and $R_z$, which characterize the surface roughness of the workpiece using a Federal Pocket Surf portable surface roughness gage. We use those measurements to determine the accuracy of our method as shown in Section 7.4.3.

The proposed method yields accurate surface roughness parameters (90.13% - 91.26% estimation accuracy). It can be used for in-site surface roughness estimation in turning processes and can be integrated with control functions of a CNC (Computer Numerical Control) lathe.

7.4.2 Recurrence based feature extraction

We capture the dynamic behavior of the acoustic emission signals, which are generated by the friction between the workpiece surface and a graphite probe, and form the feature
vectors to estimate the three parameters: arithmetic mean ($R_a$), maximum peak-to-valley roughness ($R_{\text{max}}$), and mean roughness depth ($R_z$) which characterize the surface roughness of the workpiece.

Fig. 7.12. Steps of the proposed method for estimation of surface roughness parameters
The proposed method for estimation of surface roughness parameters has three main steps as shown in Fig. 7.12. In the first step we represent the acoustic emission signals as recurrence plots. Each signal is fed to a separate recurrence preprocessing (step 1) to generate the corresponding recurrence plot individually. This step can be viewed as recurrence preprocessing activity prior to feature extraction. In the second step we compute the statistics of the recurrence plots through recurrence quantification analysis by using Equations (2.7), (2.8), (2.12) and (2.13). The statistics of the recurrence plots contain the characteristic behavior of the acoustic emission signals. In other words, a set of recurrence statistics is extracted to form the feature vectors. In the last step, we estimate the surface roughness parameters by feeding the feature vectors through a multilayer neural network.

**Recurrence preprocessing (Step 1)**

The recurrence preprocessing plays an important role in transforming the acoustic emission signals into recurrence plots in order to eventually extract the features using recurrence quantification analysis. The recurrence plots are reconstructed with a threshold value of $\varepsilon = 1.50$. This step allows us to capture the characteristics of acoustic emission signals which are influenced by the surface roughness level on the workpiece. It improves the estimation accuracy of surface roughness parameters. In Fig. 7.13 we show an example acoustic emission signal as a time series and the corresponding recurrence plot. We conclude that the acoustic emission signal is well known to be non-stationary (Bukkapatnam et al., 1995). It is therefore possible to study and analyze these signals as a non-linear system using recurrence quantification analysis.
Fig. 7.13. An example of an acoustic emission signal and its recurrence plot
**Feature Extraction (Step 2)**

After the acoustic emission signals are transformed into recurrence plots in step 1, the recurrence statistics embedded in these recurrence plots are extracted through recurrence quantification analysis. Due to the complexity and non-stationarity of acoustic emission signals, we analyze the diagonal structures and vertical structures in the recurrence plots. We extract four recurrence statistics, namely the *Entropy (ENTR)*, *Determinism (DET)*, *Laminarity (LAM)* and *Longest vertical line (VMax)*. These four statistics are used to form the feature vectors.

The details of this step are described below:

1. Calculate the recurrence statistics of all recurrence plots using Equations (2.7), (2.8), (2.12), and (2.13).
2. Construct the feature vectors \( F_i \) defined as:

\[
F_i = (ENTR, DET, LAM, V_{max}), \quad i = 1, 2, ..., N
\]

\[
\text{where } N \text{ is the total number of all recurrence plots. In other words, each feature vector contains four recurrence statistics.}
\]

**Artificial neural network estimator (step 3)**

In Fig. 7.14 we use a multilayer neural network as an estimator of surface roughness parameters. This neural network has the ability to “understand” the information captured in feature vectors (inputs) and to estimate a process state or a continuous process variable (output), as long as there exists a continuous relationship between the features vectors and the response variable and a sufficiently rich set of training examples is available.
From a practical point of view, however, the interpretation of sensor signals is made easier by their transformation to a set of feature vectors using recurrence plots and recurrence quantification analysis. Once the neural network learns the mapping between a set of the feature vectors and the surface roughness values, it computes the estimation of surface roughness parameters for any acoustic emission reading. To solve our surface roughness estimation problem at hand, a multilayer feed forward neural network is trained using back propagation algorithm.

**Fig. 7.14.** The Neural Network Architecture for Surface Roughness Estimation

### 7.4.3 Experiments and results

The proposed method is verified by conducting experiments on a turning machine (lathe). We use a 5 HP American Pacemaker lathe. Fig. 7.15 shows the experimental setup. The
acoustic emission sensor collects the acoustic emission signals via a graphite pin that is in contact with the workpiece surface being machined. The pin has been placed normal to the rotating workpiece. The cross section of the pin is a square with 4 mm side. The pin rubs the surface of the workpiece. The acoustic emission sensor, which is in contact with the pin, senses acoustic emission signals generated by the friction process.

Fig. 7.15. Setup for acquiring AE signals from a rotating workpiece

All the experiments are conducted with a constant depth of cut of 0.015 inches. K420 grade uncoated carbon inserts are used for cutting the workpiece made of AISI 1054 steel. The experiments are divided into two sets: Set 1 and Set 2. Both Set 1 and Set 2 consist of data collected at unique combinations of cutting speed (570, 730, 925, 1200, 1500 rpm) and feed rate (0.002, 0.004, 0.006, 0.008, 0.012 ipr). For each of the 25
combinations of cutting speed and feed rate, acoustic emission signals are collected at 5 different locations as the cutting progresses. Each time three acoustic emission signal samples (15 seconds apart) are collected. Each sample of length 2048 points is collected at a frequency of 2 MHz. It results in $5 \times 3 \times 25 = 375$ AE signal records in Set 1. Another 375 signal records are collected to constitute the data in Set 2. The operating conditions occurring in Set 1 and Set 2 are exactly the same.

The data in Set 1 and Set 2 experiments are used to determine the feature vectors in the proposed method. The four features of recurrence statistics for each signal are calculated. In addition, the actual cutting speed and the cutting feed rate of the turning operation are appended as additional features. The data from Set 1 experiments are used for training the neural network for surface roughness estimation. The data in Set 2 experiments are used to study the performance and accuracy of the neural network.

To carry out the analysis of the surface roughness estimation results, the four statistical parameters are computed: mean, standard deviation, mean square error (MSE), and percent of the paired differences of the measured and the estimated surface roughness values:

$$d_i = \hat{w}_i - w_i, \quad \text{for } i = 1, 2, \ldots, n$$  \hspace{1cm} (7.8)

where $\hat{w}_i$ and $w_i$ respectively are the estimated and the measured surface roughness values, and $n$ is the total number of data records in Set 2. The paired differences are also referred to as surface roughness estimation errors or simply estimation errors.
Table 7.3 shows the mean, standard deviation, MSE, and percentage of estimation errors from the neural network. For training the neural network, we used 375 pairs of measured and estimated surface roughness values over 25 tools in Set 1 experiments. To test the performance, we used another 375 pairs of measured and estimated surface roughness values over 25 tools in Set 2 experiments. Fig. 7.16 shows the comparison between the actual and estimated surface roughness values for the test set, i.e., Set 2. The R-Squared or the coefficient of determination ($R^2$) value is very close to 1. This indicates that the surface roughness estimates are accurate and consistent.
Fig. 7.16. Measured surface roughness values versus estimated values for test data
Chapter 8

8. Conclusion

This research shows that recurrence quantification analysis and recurrence network analysis offer different application possibilities. Recurrence based techniques can be modified and adapted to specific problems. Different aspects of the application of recurrence network analysis to sensor data have been studied. Methodical developments have been introduced for the nonlinear sensor data analysis. Applications to model sensor data have illustrated the applicability of these methodical developments, especially where other methods fail.

8.1 Methodical Developments

8.1.1 Recurrence based classification and estimation analysis

The recurrence based feature extraction method is investigated. The key idea in our method is the transformation of the sensor into phase space trajectories and subsequent construction of the recurrence plot. The recurrence plot is transformed into the corresponding recurrence network. The recurrence quantification analysis is used to capture the recurrence statistics of recurrence of states in the recurrence plot and the recurrence network. In other words, these statistics are the characteristics of the dynamical systems that generated the sensor data. Thus the feature vector is formed by the recurrence statistics.
The proposed method is verified on many applications such as lung sounds classification, Electromyogram classification and surface roughness estimation. These sensor data are classified and estimated using four different types of classification techniques: multilayer perceptron, support vector machine, interaction trees, and multivariate adaptive regression splines. Regardless of the classification technique, the proposed method results provide better classification and estimation accuracies when compared to those of other methods.

We have used the common parameters to construct the phase space trajectories and the recurrence plots. The results show that the common parameters of embedded dimension $m = 1$, the time delay $T = 1$, and the threshold distance $\varepsilon = 0.1$ give excellent classification accuracy in some applications. However, other sensor data may require to adjustment of these parameters.

The proposed method is an effective feature extraction method for sensor data because the proposed features vector resembles closely the behavior of dynamical systems. We can apply the proposed method to multivariate time series or even the multi-channel time series.

8.1.2 Correlation network using phase synchronization method

The phase synchronization based correlation network method has been investigated. The proposed method can be considered general-purpose in the sense of being able to construct the correlation network of sensor data using phase synchronization method. This greatly assists to analyze the patterns of sensor data from the network.
We applied the proposed method to the financial data such as stock indices. The results show that the topology of the network for the real and for the considered artificial markets is able to provide information about the collective behavior of different stocks. The correlation networks have been constructed for the weekly closing stock prices of the S&P index (439 stocks) from 2000 to 2009. The construction procedure is based on connecting any two stocks whose weekly price time series are similar in terms of phase synchronization method evaluated over a period of time. It has been found that the network formed by connecting stocks of highly correlated closing prices is a scale-free network. Some of the network parameters indicate that the market is actually heavily influenced by a select set of stocks in the financial sector.

Finally, we have proved that this proposed method could construct the correlation network of stocks using phase synchronization method. The phase synchronization method captures the dynamic behavior of the stocks and mitigates the information loss. This proposed method provides valuable insights into the behavior of highly correlated stocks which can be useful for making trading decisions.

**8.2 Applications**

The applications presented are from different scientific disciplines. Using recurrence based techniques we have achieved important results, which otherwise not feasible with alternative methods.
8.2.1 Lung sounds classification

Lung disease is a major cause of ill-health throughout the world. The diagnosis of these diseases is facilitated by electronic auscultation using a stethoscope. Auscultation with a stethoscope has many limitations. It is not easy to produce quantitative measurements or make a permanent record. Thus the classification of lung sounds becomes an essential goal for solving this problem. Moreover, many methods of sensor data analysis are not highly successful in characteristic classification of lung sounds because they are complexity, nonlinearity, and non-stationary. In this research, a novel framework of the recurrence based feature extraction method is proposed.

The lung sound signals are transformed into recurrence plots and a set of statistical features are calculated using recurrence quantification analysis. Our results show that these features play an important role as feature extraction in our framework. The lung sounds are classified using four different types of classification techniques: multilayer perceptron, support vector machine, interaction trees, and multivariate adaptive regression splines. Regardless of the classification technique, the proposed method provides better classification accuracies when compared to those of other methods.

8.2.2 Electromyogram classification

Electromyography plays an important role in clinical neurological diagnosis; it can confirm or dismiss clinical diagnoses, indicate the location and type of an abnormality or expose disorders that are clinically uncertain. The classification of neuromuscular
disorders is an essential for correct diagnosis. Moreover, many methods of time series analysis do not rely on the characteristics of electromyogram because of their complexity, nonlinearity, and non-stationarity. In this paper, a novel framework of diagnostic classification of neuromuscular disorders is proposed.

The electromyogram signals are transformed into recurrence plots and a set of statistical features are calculated using recurrence quantification analysis. Our results show that specified features play an important role as feature extraction in our method. We simulate the classification of neuromuscular disorders using these features in support vector machine. Support vector machine employing radial basis function is found to be effective for diagnosis and classification of neuromuscular disorders. The results show that our method gives 98.28% classification accuracy.

8.2.3 Stock correlation network using phase synchronization method

We used the phase synchronization method for constructing and analyzing the stock correlation network. The stock correlation network constructed using the phase synchronization method exhibits a scale free degree distribution for higher cutoff threshold values. This essentially means that the stock market is influenced only by a few major stocks. The phase synchronization method captures the dynamic behavior of the time series and reduces information loss and hence can be used for analyzing non-linear system such as stock prices.
8.2.4 Surface roughness estimation in turning processes

This research tries to estimate the surface roughness in finish turning machining operations through AE signals generated by the sliding friction between the workpiece and a graphite pin. The proposed surface roughness estimation method uses the recurrence theory and has been implemented. The proposed method accurately estimates the surface roughness in the turning processes. The feature extraction plays an important role in our approach. The experimental results confirm that recurrence statistics of AE signals contain the characteristic information of surface roughness. The outcome shows a clear correlation between the estimated and measured roughness measures for three parameters, $R_a$, $R_{\text{max}}$, and $R_z$. One possibility for improving the accuracy of the estimation is to improve the geometry of the probe (graphite pin) to minimize the variation in the contact area. Nevertheless, the stated method is general and accurate to address AE sensor-based monitoring problems and allows for possible improvements to develop an effective geometric adaptive control scheme. The results show that the proposed method is successful with 90.13% to 91.26% estimation accuracy. The accurate results indicate that the proposed surface roughness estimation method is very effective for real world manufacturing applications.
Reference


