DICTIONARY AND DEEP LEARNING ALGORITHMS WITH APPLICATIONS TO REMOTE HEALTH MONITORING SYSTEMS

by

Sherin Mary Mathews

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical and Computer Engineering

Winter 2017

DICTIONARY AND DEEP LEARNING ALGORITHMS WITH APPLICATIONS TO REMOTE HEALTH MONITORING SYSTEMS

by

Sherin Mary Mathews

Approved: ______________________________

Kenneth E. Barner, Ph.D.
Chair of the Department of Electrical and Computer Engineering

Approved: ______________________________

Babatunde Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: ______________________________

Ann L. Ardis, Ph.D.
Senior Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy .

Signed: ______________________________

Kenneth E. Barner, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy .

Signed: ______________________________

Charles G. Boncelet,Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy .

Signed: ______________________________

Leonard J. Cimini, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy .

Signed: ______________________________

Chandra Kambhamettu, Ph.D.
Member of dissertation committee

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor Dr. Kenneth Barner for his continuous support for my Ph.D. research, for his patience, motivation, and immense knowledge. It has been such an honor and pleasure to work with him. I appreciate all his contribution of time and ideas to make my Ph.D. experience stimulating and productive. I would also like to thank Dr. Blake Meyers for his guidance and support during my Ph.D.

My sincere thanks also go to Dr. Chandra Kambhamettu and Dr. Daniel Lees for their encouragement and insightful comments on my dissertation. I am also grateful to have such a great thesis committee. Thank you, Dr. Charles Boncelet, Dr. Leonard J. Cimini and Dr. Chandra Kambhamettu for evaluating my work and providing valuable comments and suggestions. I am grateful for the administrative support received from the ECE office at the University of Delaware, including Ms. Karen DiStefano, Ms. Gwen Looby.

I have been lucky to get the opportunity to work with well-known researchers from the industry: Dr. Mark Yarvis and Dr. Hakim Nagib at Intel Labs, Dr. Francisco Imai and Dr. Sandra Skaff at Canon Research Labs. They motivated and taught me to use machine learning to solve real world problems and have been instrumental in my choice of career path, and I am grateful for their guidance and support.

The members of our lab group have also contributed immensely to my personal and professional time. Our weekly meetings and discussions helped me not only with my work directly, but also with becoming a better researcher in general. I would like to thank Dr. Luisa Polania with whom I had a very active work collaboration and from whom I also learned a lot. The group has been a source of friendship as well as

good advice. Special thanks go to Xin Guo who has always been a great support and friend. Thank you to all my friends and colleagues in the lab: Dr. Jinglun Gao, Dr. Yin Zhou, Dr. Luisa Polania, Dr. Rui Hu, Ms. Xin Guo, Mr. Sergio Matiz Romero, Mr. Xinjie Lan.

I could not complete my work without invaluable love and assistance from my parents. Thank you for supporting me throughout my life, and my success is a testament to their sacrifices. I also would like to thank my parents for being so supportive of everything, every decision, every step, every journey. No words would be enough to express thanks for whatever they have done for me throughout my life. I would also like to mention my parents-in-laws and my extended family for their love and support throughout.

I would like to thank my husband for being so understanding and patient over the course of my Ph.D. His unconditional support has helped me go through the tough times and kept me balanced. Having him by my side gave me strength and it is wonderful to be able to share my life with him. A special mention for my eight-month-old baby who has always been a source of joy for me. She brings lots of fun times and sweet memories, and it is worth sleepless nights, restless days and mountains of laundry!

And last but not least, I would like to thank all my friends for everything. I am very fortunate to have friends to count on, laugh with and learn together. Thank you to everyone who was a part of this journey. Your support made this possible.

TABLE OF CONTENTS

LIST OF TABLES . . . . **x**
LIST OF FIGURES . . . . **xiv**
ABSTRACT . . . . **xvi**

Chapter

1 INTRODUCTION **1**

1.1 Applications Addressed in this dissertation 2

1.1.1 Activity Recognition task 3
1.1.2 Intensity Estimation Task 4
1.1.3 ECG Recognition Task 5

1.2 Prior Work Reviewed 6
1.3 Motivation 8
1.4 Research Contributions 11
1.5 Publications 14

1.5.1 Conferences 14
1.5.2 Journals 15

1.6 Thesis Organization 15

2 SPARSE SIGNAL REPRESENTATION AND DICTIONARY LEARNING **17**

2.1 Sparse Representation 17
2.2 Compressed Sensing 18
2.3 Dictionary Learning 19
2.4 Review of Dictionary Learning algorithms 21
2.5 Dictionary Pair Learning 25
2.6 Conclusion 28

3 CLASS SPECIFIC CENTRALIZED DICTIONARY PAIR LEARNING FOR ACTIVITY RECOGNITION USING WEARABLE SENSORS 29

3.1 Introduction 29
3.2 Proposed Methodology 30

3.2.1 Proposed DPL model with class specific centralized regularizer term 32

3.3 Experimental results and discussion 35

3.3.1 PAMAP2 Database 35

3.3.1.1 PAMAP2 -Results on Intensity Estimation Task . . . 36
3.3.1.2 PAMAP2 - Results on Activity Recognition Task . . 38

3.3.2 SBHAR 40
3.3.3 WISDM 41

3.4 Conclusion 43

4 MAXIMUM CORRENTROPY BASED DICTIONARY LEARNING FRAMEWORK FOR PHYSICAL ACTIVITY RECOGNITION USING WEARABLE SENSORS 44

4.1 Introduction 44
4.2 Proposed Methodology: Correntropy based Dictionary Learning . . . 45

4.2.1 Maximum Correntropy Criterion (MCC) 45
4.2.2 Correntropy based Dictionary Pair Learning Framework . . . 46

4.3 Experimental results and discussion 50

4.3.1 Results with PAMAP2 Intensity Estimation and Activity Recognition Tasks 51

4.3.1.1 Results on PAMAP2 Intensity Estimation Task . . . 52
4.3.1.2 Results on PAMAP2 Activity Recognition Task . . . 53

4.3.2 SBHAR 55

4.3.3 WISDM 56

4.4 Conclusion 58

5 A HIERARCHICAL DICTIONARY LEARNING FRAMEWORK FOR PHYSICAL ACTIVITY RECOGNITION USING WEARABLE SENSORS 59

5.1 Introduction 60
5.2 Related Work 62
5.3 Proposed Methodology 63

5.3.1 CFU Tree based Clustering 64
5.3.2 Creating the hierarchy using similarity metric 66
5.3.3 Class Specific Centralized Dictionary Pair Learning Classifier 67

5.4 Evaluation Techniques for the Proposed Framework with experimental results and discussions 69

5.4.1 Database 69

5.5 Evaluation Metrics 69
5.6 Experimental Results and Discussion 71

5.6.1 Results from PAMAP2 dataset 71
5.6.2 Results from the WISDM dataset 75

5.7 Conclusion 77

6 A DEEP LEARNING FRAMEWORK FOR SENSOR-BASED ECG CLASSIFICATION 79

6.1 Introduction 79
6.2 Proposed Methodology 82

6.2.1 Preprocesing 82
6.2.2 Processing 83
6.2.3 Feature Extraction 84

6.2.3.1 Feature Set 1 84

6.2.3.2 Feature Set 2 86

6.3 Deep Learning Framework 87

6.3.1 Restricted Boltzmann Machine 87

6.3.2 Deep Belief Networks 89

6.4 Evaluation of Proposed Methodology with results and discussion . . . 90

6.4.1 The MIT-BIH Database 90

6.4.2 AAMI Standard 91

6.4.3 Evaluation Metrics 91

6.4.4 Experimental Results and Discussion 92

6.5 Conclusion 97

7 CONCLUSION AND FUTURE WORK 99

7.1 Summary of research outcomes 99

7.2 Future Work 101

BIBLIOGRAPHY 105

LIST OF TABLES

3.1 Overall confusion matrix using class specific centralized dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes. 37

3.2 Overall confusion matrix in % using class specific centralized dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes. 37

3.3 Recognition results on PAMAP2-IE Dataset. 37

3.4 Computation time for classifying one query activity on PAMAP2-IE Dataset. 37

3.5 Confusion matrix using proposed framework on PAMAP2 -AR dataset. 38

3.6 Confusion matrix in % using proposed framework on PAMAP2 -AR dataset. 38

3.7 Recognition results on PAMAP2-AR Dataset. 39

3.8 Computation time for classifying one query activity on PAMAP2-AR Dataset. 39

3.9 Performance of the proposed algorithm on the PAMAP2 - AR dataset with varying training data 39

3.10 Overall confusion matrix using proposed framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes. 40

3.11 Overall confusion matrix in % using proposed framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes. 40

3.12 Recognition results on SBHAR Dataset. 41

3.13 Performance of the proposed algorithm on the SBHAR dataset with varying training data 41

3.14 Overall confusion matrix using proposed class specific centralized dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes. 42

3.15 Overall confusion matrix in % using proposed class specific centralized dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes. 42

3.16 Recognition results on WISDM Dataset. 42

3.17 Performance of the proposed algorithm on the WISDM dataset with varying training data 43

4.1 Overall confusion matrix using maximum correntropy criterion based dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes. 52

4.2 Overall confusion matrix in % using maximum correntropy criterion based dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes [1]. 52

4.3 Comparison of proposed approach on PAMAP2-IE dataset to state-of-the-art methods in terms of accuracy (calculated in %) . . 53

4.4 Confusion matrix using proposed framework on PAMAP2 -AR dataset. 53

4.5 Confusion matrix in % using proposed framework on PAMAP2 -AR dataset. 54

4.6 Comparison of proposed approach on PAMAP2-AR dataset to state-of-the-art methods in terms of accuracy (calculated in %) . . 54

4.7 Overall confusion matrix using correntropy based dictionary pair learning framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes. . . 55

4.8 Overall confusion matrix in % using correntropy based dictionary pair learning framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes. . . 56

4.9 Recognition results on SBHAR Dataset. 56

4.10 Performance of the proposed algorithm on the SBHAR dataset with varying training data 56

4.11 Overall confusion matrix using correntropy based dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes. . . 57

4.12 Overall confusion matrix in % using correntropy based dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes. . . 57

4.13 Recognition results on WISDM Dataset. 57

4.14 Performance of the proposed algorithm on the WISDM dataset with varying training data 58

5.1 Comparison of recognition accuracy of all non-hierarchical (NH-prefix) models (calculated in %) 74

5.2 Comparison of recognition accuracy of all hierarchical(H- prefix) models(calculated in %) 74

5.3 Comparison of Internal evaluation scores of all hierarchical approaches 75

5.4 Comparison of recognition accuracy of all non-hierarchical (NH-prefix) models (calculated in %) on WISDM Dataset. 76

5.5 Comparison of recognition accuracy of all hierarchical (H- prefix) models (calculated in %) on WISDM Dataset. 77

6.1 Comparison of classification results at sampling rate of 360 Hz . . 93

6.2 Comparsion of classification results at sampling rate of 114 Hz . . 93

6.3 Comparison of FS1 + Deep Learning Framework at Sampling Rate of 360 Hz 94

6.4 Classification results in % for FS1 + Deep Learning Framework at Sampling Rate of 360 Hz 94

6.5 Comparison of FS2 + Deep Learning Framework at Sampling Rate of 360 Hz 94

6.6 Classification results in % for FS2 + Deep Learning Framework at Sampling Rate of 360 Hz 95

6.7 Comparison of FS1 + Deep Learning Framework at Sampling Rate of 114 Hz 95

6.8 Classification results in % for FS1 + Deep Learning Framework at Sampling Rate of 114 Hz 95

6.9 Comparison of FS2 + Deep Learning Framework at Sampling Rate of 114 Hz 95

6.10 Classification results in % for FS2 + Deep Learning Framework at Sampling Rate of 114 Hz 96

LIST OF FIGURES

1.1 Illustration of different wearable sensors: My Intelligent Communication Accessory (MICA), Myo (Muscle Activity Tracker), Smart Wristband, Tracking Watch, Sensoria (Smart Socks) FootLogger (Shoe Sole for Fitness Tracking) and FitBit [2]. 1

1.2 Illustration of remote health monitoring system. Motion and heart rate data are gathered via wearable sensors and information is transmitted to caregiver. Caregiver could use this information to implement interventions as and when required [2]. 3

1.3 Aerobic Activity Recognition Monitoring system. A large number of similar activities are considered for the classification task (walking, running, nordic walking and cycling) [3] 3

1.4 Aerobic Activity Recognition Monitoring system. A large number of high level (composite) activities are also considered for the classification task [3]. 4

1.5 Aerobic Intensity Estimation Monitoring system. The activities are distinguished based on MET values into different intensity levels i.e high , moderate and light effort [3]. 4

1.6 Illustration of ECG Recognition System 5

2.1 Compressive Sensing Measurement Process [4]. 19

3.1 Proposed Class Specific Centralized Dictionary Learning Framework 30

4.1 Contour Maps of Correntropy Induced Metric 46

4.2 Proposed Correntropy based Dictionary Pair Learning Framework . 46

5.1 Example of clustering where C_i is the new input object [5]. 65

5.2 Structure of the human intuition based hierarchical classification framework 72

5.3 Dimensionality Reduction Results for determining feature Correlation. 72

5.4 Structure of the feature correlation based hierarchical classification framework 73

5.5 Structure of CFu-tree based hierarchical classification framework on PAMAP2 dataset 73

5.6 Structure of the CFu-tree based hierarchical classification framework on WISDM Dataset 75

5.7 Structure of the feature correlation based hierarchical classification framework on WISDM Dataset 76

6.1 Applications of Deep Learning in object recognition and tracking . 80

6.2 Block diagram of the proposed methodology 82

6.3 ECG Cardiac Trace [6] 84

6.4 Segmented Morphology Intervals Features [7] 86

6.5 Schematic of a restricted Boltzmann machine 88

6.6 Schematic of a deep belief network of three layers. 90

ABSTRACT

Dictionary and deep learning algorithms facilitate efficient signal representations, thereby offering tremendous representational power along with achieving good recognition rates in real-world machine learning problems. In this dissertation, we present three dictionary learning approaches and a deep learning framework for classification tasks related to remote health monitoring systems.

This dissertation presents a more robust class specific centralized dictionary learning method to solve the wearable sensor-based physical activity classification problem. Inspired by experiments that achieved high recognition rates using a few representative samples on high dimensional data, we explore the physical activity recognition signals from wearable sensors and propose a dictionary pair learning-based framework for human physical activity monitoring and recognition. The essential strategy involves integrating the class specific centralized regularizer term into the dictionary pair learning objective function and efficiently optimizing the objective function by combining the alternating direction method of multipliers and the $l_1 - l_s$ minimization method. Specifically, the class specific regularizer term ensures that the sparse codes belonging to the same class will be concentrated thereby enhancing the classification performance. Experimental results show that the classifiers built in this framework achieve higher recognition rate over four activity recognition tasks and outperforms state-of-the-art methods.

Physical activity recognition involves variations in different walking styles and human body movements which result in the erroneous classification of similar activities. To address this issue, we present a correntropy induced dictionary pair learning framework to achieve improved recognition. In particular, the dictionary pair learning algorithm developed based on the maximum correntropy criterion is much more insensitive to outliers. A combination of alternating direction method of multipliers and an iteratively reweighted method is employed to approximately minimize the objective function. Evaluations are conducted using four activity recognition tasks and results show that the proposed classifier framework achieve enhanced performance compared to the state-of-the-art recognition systems.

Although classification accuracy is enhanced using state-of-art classifiers, actual recognition performance tends to fall off when distinguishing a large number of similar activities. To this end, we propose and evaluate methods for analyzing hierarchical and sequentially structured human activities, designed to scale activity recognition by creating a hierarchical cluster of activity labels. Instead of using a single classifier to distinguish between large numbers of activities, we propose a hierarchy of classifiers, each of which distinguishes between child nodes at a particular location in the hierarchy. We hypothesize that building such a hierarchy of activity will improve recognition performance over that of the flat classifier model. We validate the effectiveness of our proposed model by employing it on two standard activity recognition datasets, which include a large set of similar physical activities. The results of hierarchical structure modeling furnish evidence that decomposing the problem leads to more accurate specialized classifiers.

This dissertation also applies deep learning methodology to the classification of single-lead electrocardiogram (ECG) signals. State of-the-art automatic ECG recognition systems often rely on a pattern-matching framework thereby requiring high sampling rates and burdensome computational times to classify arrhythmias. Deep learning networks represent a high level of abstraction showcasing its tremendous representational power. Consequently, to enable implementation in real time, we develop

a deep learning framework that includes Restricted Boltzmann Machine and Deep Belief Networks for ECG classification with lower computational time, making it a highly practical option in a clinical setting.

Chapter 1

INTRODUCTION

Recent advances in ubiquitous computing technologies have contributed to the broad scale adoption of pervasive sensor-based devices. Traditionally, unobtrusively monitoring the physical activity of a patient once he/she is released from the hospital is a critical but difficult task. Although patients fill out questionnaires and surveys on their physical activity, this data collection tends to be imprecise. With recent progress in pervasive healthcare and wearable sensing, providers have resorted to a new approach of having patients wear sensors all day, thus acquiring a more precise measurement of long-term activity. Wearable sensors (or inertial sensors) finds applications in healthcare, entertainment, sports, security, and commercial fields as they can provide accurate and reliable information on a person's activity and behavior, thereby ensuring a safe and sound living environment.

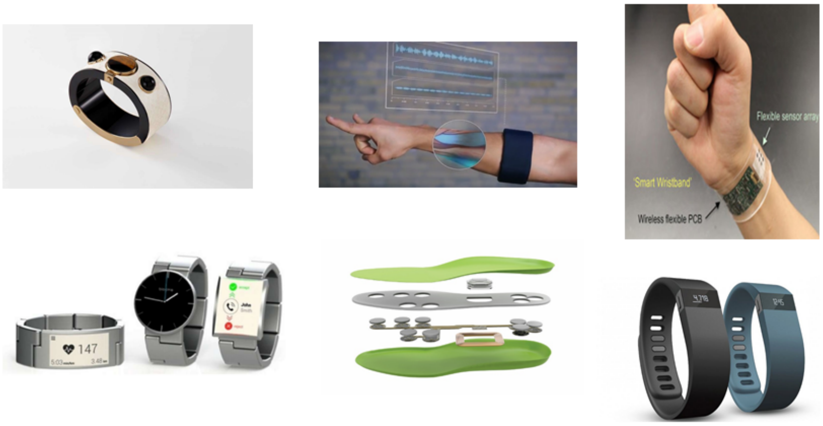

Figure 1.1: Illustration of different wearable sensors: My Intelligent Communication Accessory (MICA), Myo (Muscle Activity Tracker), Smart Wristband, Tracking Watch, Sensoria (Smart Socks) FootLogger (Shoe Sole for Fitness Tracking) and FitBit [2].

Activity recognition using sensor signals is an emerging area of research interest in pervasive computing and has been investigated in health care [8], [9], smart homes [10], [11], [12], situated support domains [13] and the monitoring of mental and physical well-being [14], [15], [16]. Wearable sensor-based activity recognition enables the fine-grained estimation of a person's activities over extended periods of time. Having good applications in the field of building a pervasive and smart environment for fall detection of elderly people and providing personalized support in smart home environments, it has become a perfect platform to deliver long-term personalized healthcare support anywhere and anytime.

1.1 Applications Addressed in this dissertation

Ubiquitous Sensing is an active research area with the main purpose of extracting knowledge from the data acquired by pervasive sensors [17]. Particularly, the recognition of human activities has become a task of high interest within the assistive living field, especially for home rehabilitation, assessment of treatment efficiency, medical and security applications. Therefore, recognizing activities such as walking, running, or cycling becomes quite useful to the caregiver to provide feedback about the patients behavior and to detect abnormal activities and prevent undesirable consequences [18]. A conceptual representation of a remote monitoring system is illustrated in Figure 1.

In this dissertation, we address three recognition tasks using accelerometers and heart rate data obtained from sensors:

- activity classification task
- intensity estimation task
- ECG recognition task

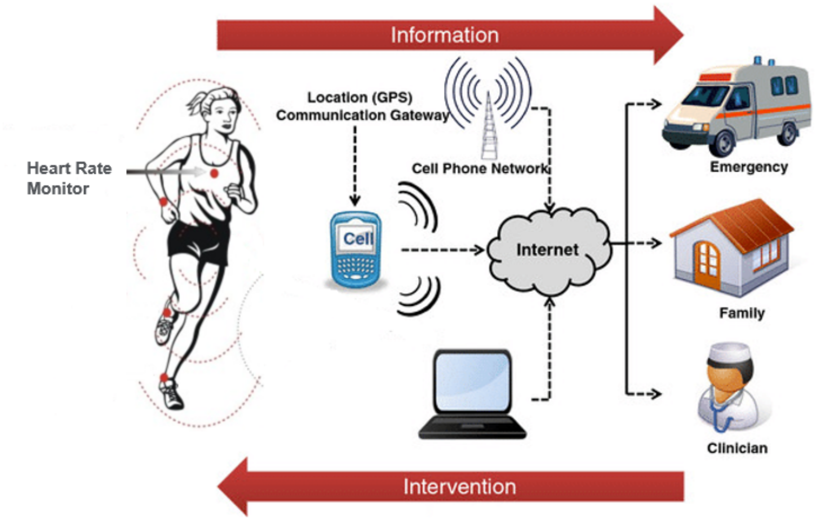

Figure 1.2: Illustration of remote health monitoring system. Motion and heart rate data are gathered via wearable sensors and information is transmitted to caregiver. Caregiver could use this information to implement interventions as and when required [2].

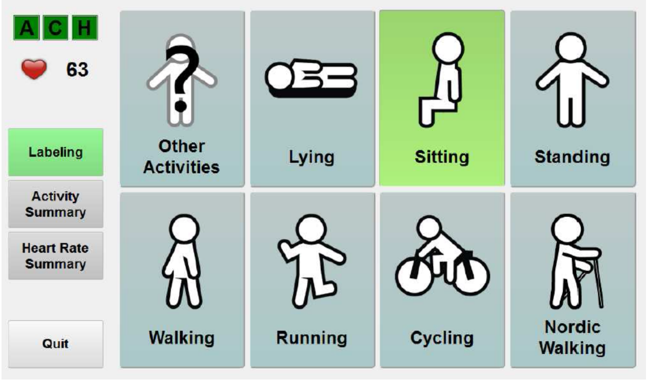

Figure 1.3: Aerobic Activity Recognition Monitoring system. A large number of similar activities are considered for the classification task (walking, running, nordic walking and cycling) [3]

1.1.1 Activity Recognition task

For activity recognition, the major goal of current research in context-aware computing is to distinguish varied number of activities with a high recognition rate along with lower computational testing time (Figure 1.2 and Figure 1.3). To this end, we develop two robust dictionary learning activity recognition frameworks to distinguish closely similar activities with reduced misclassification rate. The proposed methodology is tested on three activity recognition datasets and extensive experimental results on benchmark datasets demonstrate the superior performance of our proposed approaches compared with conventional approaches.

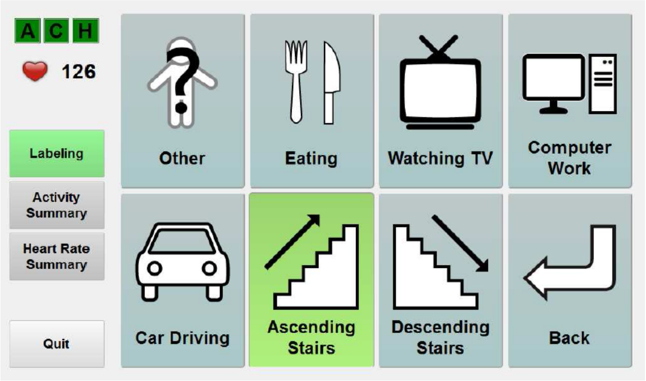

Figure 1.4: Aerobic Activity Recognition Monitoring system. A large number of high level (composite) activities are also considered for the classification task [3].

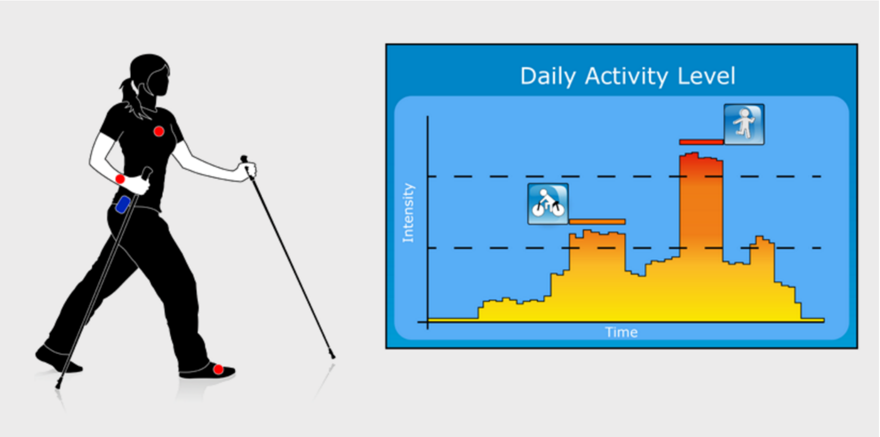

Figure 1.5: Aerobic Intensity Estimation Monitoring system. The activities are distinguished based on MET values into different intensity levels i.e high , moderate and light effort [3].

1.1.2 Intensity Estimation Task

Recent research on physical activity monitoring have also focused on estimating the intensity of performed activities. Evaluating the intensity of activity (i.e., light, moderate or vigorous) is addressed in [19] which illustrates the metabolic equivalent (MET) - a parameter that refers to the energy expenditure of physical activity for different physical activities [20] (Figure 1.4). Thus the ground truth for the intensity estimation task is based on the metabolic equivalent (MET) of the different physical activities, provided by [19]. Overall, the intensity estimation task is considered as a 3-class classification problem on PAMAP2 dataset and can be defined as follows:

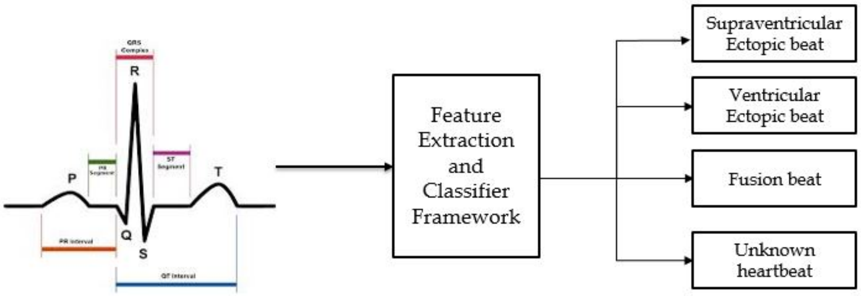

Figure 1.6: Illustration of ECG Recognition System

- lie, sit, stand, watch TV, computer work, drive a car, iron, fold laundry, and clean house are regarded as activities of light effort (< 3.0 METs);
- walk, cycle, Nordic walk, descend stairs and vacuum clean as activities of moderate effort (3.0-6.0 METs);
- run, ascend stairs, play soccer and rope jump as activities of vigorous effort (> 6.0 METs).

1.1.3 ECG Recognition Task

Wearable sensors can also gather physiological data and monitor vital signs (heart rate and blood pressure) thereby, enabling overall health monitoring. Physiological monitoring can help diagnose and treat individuals with cardiovascular, hypertension and neurological diseases. Clinical studies are currently carried out to evaluate the performance of wearable sensor platform to monitor physiological data over a long period of time and improve clinical management of patients with congestive heart failure. In this dissertation, we address the sensor based ECG classification task by applying of Restricted Boltzmann Machine and Deep Belief Networks. Experimental results demonstrate that proposed framework achieves a high classification accuracy at a lower sampling rate using the proposed framework.

1.2 Prior Work Reviewed

Classification of basic physical activities (such as walk, run, cycle) and basic postures (i.e. lie, sit, stand) have been well researched [21], [22], [8], [23], and good recognition performance can be achieved with a single 3D-accelerometer and simple classifiers. Moreover, recent studies have focused on estimating the intensity of these basic activities (e.g., in [24], [25]).

Over the years, many studies have analyzed both simple and complex human activities reported by wearable sensors. A large number of these have focused on determining which features in this activity data are the most informative, and how these data can be most effectively employed to classify the activities [8], [23], [26], [27].

Other research endeavors have investigated which computational model would be the most appropriate to represent human activity data [28], [29], [30]. Huang and Schneider [31] proposed spectral learning algorithms for hidden Markov models (HMMs) that incorporate static data to demonstrate the performance of their new algorithms on real (not synthetic) data. Moreover, Clifton et al. [32] introduced an extreme function theory for novelty detection, and illustrated their proposed method on wearable sensors based activity recognition dataset. Despite such research efforts, the scalability of handling large intra-class variations and the robustness of many existing human activity recognition techniques to the model parameters remains limited.

To make physical activity monitoring feasible in everyday life scenarios, an activity recognition framework must be robust, i.e., it must handle a broad range of everyday, household, or sports activities and must manage a variety of potential users. Two challenges facing these frameworks are dataset size and simultaneously occurring background activities. First, using smaller activity datasets consisting of merely a few basic recorded activities reduces the scope of the framework since these methods would then only apply to specific scenarios. Although current research has focused on increasing the number of activities that are recognized, including examples of everyday, household, or sports activities, each increase in the number of activities causes the classification performance to fall off. Second, recording and using only a small set of a few activities in basic activity recognition, without having simultaneous background activities, limits the applicability and versatility of the developed algorithms. Real-time scenarios might include some activity switching, thus requiring testing on a wider range of activities than were used for training.

In recent years, sparse models have been widely used in a variety of applications in computer vision and pattern recognition, e.g., image analysis [33], image denoising [34], image restoration [35], [36], [37] image processing and recognition [38], [39], [33]. The underlying principle of sparse modeling is to represent objects using as few variables as possible [37], [40], and the success of sparse coding is ascribable to the fact that high-dimensional data of particular types often lie on some low-dimensional manifolds. A sparse signal can be summarily expressed as a linear combination of a few signal items (called atoms or bases) from an over-complete dictionary. Recently, much attention has been focused on applying dictionary learning techniques to problems of computer vision and image processing, such as image denoising [34], image inpainting [41], and compressive sensing [42], [43], [44]. Infilling of missing pixels and image and speech classification problems have been successfully addressed by dictionary learning mainly because of its robustness towards missing data and noise. Inspired by experiments that achieved high recognition rates using a few representative samples on high-dimensional data, we explored the physical activity recognition from wearable sensors and proposed a dictionary pair learning-based framework for human physical activity monitoring and recognition. To achieve this, we are motivated to seek efficient methods to generate informative data representations along with obtaining robust classification rates.

Our work aims to develop dictionary learning frameworks for physical activity recognition and intensity estimation, thereby extending the applicability of such systems. To our best knowledge, dictionary learning and specifically dictionary pair learning have never been used in wearable sensor-based applications. Consequently, our novel dictionary learning-based framework algorithm will promote future research on this method's potential applicability for accurate sensor-based data classification and other physiological-signal classification.

1.3 Motivation

One facet of our motivation thrives on machine learning studies which have been extensively explored over the last three decades for its capability to learn to map of functions from patterns. Classical machine learning frameworks typically require domain knowledge and a large amount of training data to accurately establish the relationship between the data input and class labels for classification. The design of a machine learning framework consists of two major phases: training phase and test phase. In the training phase, the algorithm adjusts the internal parameters to build a mapping model that approximates the implicit relationship between the input and output training samples. The developed model is evaluated to test the generalization ability of the algorithm in the testing phase.

Recently, a significant research effort has been devoted to finding compact or sparse representation of signals, and to enhance the processing ability for large-scale data. Sparse coding provides a class of algorithms for finding succinct representations of data; by learning basis functions that capture higher-level features in the data. Within the machine learning domain, we extend the dictionary and deep learning models as their primary strength lies in succinct representation, which essentially abstracts the dominant information within the data. An added boon to the involvement of this deep learning approach is that it can also help reveal unknown feature coherences of input signals, an important capability for learning tasks that involve complicated models [45], [46], [47], [48]. In this work, we present four frameworks wherein we extend and modify the deep and dictionary learning models as needed to be more effective for each classification task.

It is also worth noting that our classification task has its own challenges, such as intra-class variability, inter-class similarity, asides from the diversity and complexity of the physical activities themselves. Thus, a successful activity-recognition technique requires classification algorithms that are sufficiently robust for classification even with limited training data. Unlike other time series signals (like speech signals and financial signals), human activity-based signals have few parts of a continuous signal stream that are relevant to the concept of interest (i.e. human activities), and the dominant irrelevant part mostly corresponds to the Null activity. Furthermore, considering human activity, in reality, we learn that every activity is a combination of several basic continuous movements. Typically, a human activity could last a few seconds in practice, and within one second a few basic movements could be involved. We require the feature extraction method to be effective enough to capture the nature of basic continuous movements as well as the salience of the combination of basic movements. All these challenges make it highly desirable to develop a systematic feature representation approach to effectively characterize the nature of signals related to the activity recognition task.

Dictionary learning has been successfully applied to many problems namely including image and speech classification problems particularly due to its robustness to missing data and noise. The primary strength of this technique lies in the compact representation, which essentially allows capturing the significant information within the data. This led to the first motivation of this dissertation - exploiting dictionary learning for activity recognition, as dictionary learning seeks a compact set of bases to best represent each signal in training set under some sparsity constraints.

We first present a novel dictionary learning technique for a sensor platform that improvises physical activity recognition rate by leveraging a class specific regularizer term into the dictionary pair learning objective function. Secondly, we present an effective maximum correntropy criterion-based dictionary pair learning framework to evaluate the robustness of activity recognition and intensity estimation of aerobic activities using data from wearable sensors. Inorder to further reduce misclassification rate, we introduce a hierarchical class specific dictionary learning algorithm which divides the multi- class activity recognition problem into smaller sub- problems.

ECG signal classification, also being sensor based data constitute a prime factor in recognition of life-threatening cardiac arrhythmias. Computerized recognition of ECGs has become a well-established practice, assisting cardiologists in the task of classifying long-term ECG recordings. However, most of the previous work often rely on a pattern-matching framework that represents an ECG signal as a sequence of stochastic patterns, and hence they require high sampling rates and thus burdensome computational times to classify arrhythmias. Consequently, to enable implementation in real time, these systems must enlarge their classification criteria by using a set of simple features and a lower sampling rate. This fact leads to the next motivation of this dissertation: the need for developing a single lead ECG recognition framework using a set of simple features and lower sampling rate.

Deep learning networks, implemented using stacked autoencoders, are capable of representing highly expressive abstractions, thereby compactly yielding much larger sets of functions than shallow networks can [49]. Through the tremendous representational power of hierarchical feature learning, these networks can help discover unknown feature coherences of input signals, a characteristic that is crucial for learning tasks involving complicated models [50]. Inspired by recent progress in the area of deep learning (especially its application to speech recognition, natural language processing, and object recognition), we developed a deep learning framework that yielded competitive ECG classification performance at a lower computational time.

1.4 Research Contributions

This thesis aims to address the activity recognition problem in wearable sensors using novel dictionary and deep learning frameworks. Within the machine learning techniques, we extend and modify the dictionary and the deep learning models as they focus on concise signal representation, thereby resulting in better feature characterization. We demonstrate the proposed frameworks on applications related to human health monitoring. Fundamental questions addressed include:

- How to obtain a robust classification framework such that it reduces dependency on complicated hand-crafted features by being subject- and trained-activity independent? How to improve the underlying dictionary learning algorithm to accurately classify the misclassified examples? How to optimize the learning model to improve the classification performance?

  The dissertation leverages dictionary pair learning in order to obtain a subject-independent activity recognition with higher classification accuracy as well as with reduced computational time. The classification accuracy of the underlying algorithm is further improved by adding a discriminative regularizer term to the objective function which ensures that sparse codes belonging to the same class are concentrated. Also, a correntropy-based robust algorithm which is insensitive to large outliers is introduced, thereby resulting in higher recognition rates.

- How to supplement and further remediate the classifier algorithms by analyzing the errors made in the testing phase? How can the feature vectors be changed to improve the classification? This problem is addressed by proposing a combination of hierarchical classifier recognition scheme along with feature selection algorithm. The classes to be recognized (defined as macro-classes) are merged at each node in the hierarchy, and this procedure is continued at subsequent nodes and levels. The proposed modular learning system consists of a hierarchy of classifiers, each solving different multi-class activity recognition problem. This proposed hierarchical classifier shows to give better classification results than single complex classifiers.

- How to develop intrinsic patterns and learn complex functions from input data automatically without using complex human-engineered features to perform classification?

  This is done by developing a deep learning based approach which improves classification accuracy of single-lead ECG signal at lower sampling rate by using simple features, thereby eliminating the need for complex feature extraction processes. This was done by employing a Restricted Boltzmann Machines and Deep Belief Networks framework. We also introduce an efficient feature learning to automatically learn intrinsic high-level features using deep belief networks.

This dissertation presents frameworks for automatic human activity recognition using wearable sensor-based data. It proposes two dictionary learning based framework to evaluate the robustness of activity recognition and intensity estimation of aerobic activities using data from wearable sensors. To achieve an even better feature learning result by capturing intrinsic properties of the features, we also present a deep learning framework for single-lead ECG sensor data classification, leading to robust classification performance at lower sampling rates.The work presented in this dissertation is one of the first attempts on the application of dictionary learning framework on wearable sensor-based human activity recognition. This work has led to publications mentioned in section 1.4 and our key contributions include:

- Proposing a novel dictionary pair learning framework to achieve physical activity recognition by leveraging a class specific regularizer term into the objective function. The proposed algorithm jointly learning a synthesis dictionary and an analysis dictionary wherein the class-specific regularizer term ensures that the sparse codes belonging to the same class will be concentrated thereby proving beneficial for the classification stage. A combination of an alternating direction method of multipliers and a $l_1 - l_s$ minimization method is employed to approximately minimize the objective function.

- Presenting a correntropy induced dictionary pair learning framework for an activity recognition and an intensity estimation problem. In particular, the dictionary pair learning algorithm is developed based on the maximum correntropy criterion, which is much more insensitive to outliers. In order to develop a more tractable and practical approach, we employ a combination of alternating direction method of multipliers and an iteratively reweighted method to approximately minimize the objective function.

- Designing a conceptual clustering based hierarchy of classifiers, each of which distinguishes between child nodes at a particular location in the hierarchy. By injecting the automated incremental clustering methodology established on similarity between class nodes, a semantic attribute representation, and a multi-layer classifier is designed.

- Validating the effectiveness of the proposed models by employing it on activity recognition problems and an intensity estimation problem, which includes a large number of physical activities from four datasets.

- Exploring and designing a deep learning methodology to the classification of single-lead electrocardiogram (ECG) signals. Specifically, the work demonstrates the application of the Restricted Boltzmann Machine (RBM) and deep belief networks (DBN) for ECG classification following the detection of ventricular and supraventricular heartbeats using single-lead ECG.

- Evaluating the performance of the proposed methods and comparing them with the state-of-the-art algorithms for ECG classification. .

1.5 Publications

The research undertaken in this thesis has resulted in the following publications:

1.5.1 Conferences

- Mathews, Sherin M., Luisa F. Polania, and Kenneth E. Barner. ”Leveraging a discriminative dictionary learning algorithm for single-lead ECG classification.” 41st Annual Northeast Biomedical Engineering Conference (NEBEC) IEEE, 2015.

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”Am I your sibling? Inferring kinship cues from facial image pairs.” 49th Annual Conference on Information Sciences and Systems (CISS), IEEE 2015.

- Sherin M., and Kenneth E. Barner ”A Deep Learning Framework for Single-lead ECG classification ” Grace Hopper ACM Research Conference 2015

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”Maximum Correntropy based Dictionary Learning framework for physical activity recognition using wearable sensors.” International Symposium on Visual Computing (ISVC) Springer International Publishing, 2016.

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”Centralized Class Specific Dictionary Pair Learning for Wearable Sensors based Activity Recognition.”(Accepted at Information Sciences and Systems (CISS), 2017)

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”A Hierarchical Dictionary Learning framework for physical activity recognition using wearable sensors.” (In Preparation to IEEE International Conference on Image Processing (ICIP 2017))

1.5.2 Journals

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”A Deep Learning Framework for Single-lead ECG classification” (Submitted for publication at Journal of Multimedia Tools and Applications (MTA))

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”Maximum Correntropy based Dictionary Learning framework for physical activity recognition using wearable sensors” (Submitted for publication at Image Vision Computing (IVC))

- Mathews, Sherin M., Chandra Kambhamettu, and Kenneth E. Barner. ”Hierarchical Class Specific Centralized Dictionary Pair Learning for Wearable Sensors based Activity Recognition” (Submitted for publication at Pattern Recognition (PR))

1.6 Thesis Organization

These contributions are organized into six chapters with the current chapter presenting the introduction. Below are the general outline and structure of the remainder of the dissertation.

Chapter 2- Gives a comprehensive review of sparse representation and dictionary learning concepts. We review efficient dictionary learning algorithms that can be used to learn sparse representations from the input data, which can also help the supervised classification tasks.

Chapter 3 - Introduces the centralized class specific dictionary pair learning algorithm for wearable sensors based activity recognition and demonstrates superior classification performance by making sparse codes in the same class concentrated.

Chapter 4 - Presents a maximum correntropy-based dictionary learning framework for physical activity recognition using wearable sensors.

Chapter 5 - Proposes an ensemble based framework comprising of hierarchical dictionary pair learning classifiers for the activity recognition tasks. To model the sequential structure of the hierarchy, an agglomerative clustering model is considered to define the similarity of attributes of activities.

Chapter 6 - Explains a deep learning based model for automatic classification of sensor based ECG signals. The approach holds promise as a scalable algorithm for learning hierarchical representations from high-dimensional, complex data.

Chapter 7 - Summarizes the research contributions and provides suggestions for future research directions.

Chapter 2

SPARSE SIGNAL REPRESENTATION AND DICTIONARY LEARNING

In this chapter, Section 2.1 and Section 2.2 focus on the theory of sparse signal representation and compressive sensing. Thereafter, we introduce dictionary learning for classification in Section 2.3 which is followed by a review of existing dictionary learning algorithms for classification in Section 2.4. Section 2.5 presents the dictionary pair learning for classification, followed by the summary of the chapter in Section 2.6.

2.1 Sparse Representation

Sparse representation has proven its credibility in machine learning, computer vision and image processing. It has been widely studied and has paved the way for state-of-the-art implementations with astounding results in numerous applications such as in the field of computer vision, audio and image classification, image denoising [34], texture synthesis and image restoration [36], [35]. A sparse signal can be comprehensively expressed as a linear combination of a few signal atoms over an over-complete dictionary. The learned atoms resemble the neurons in the visual cortex [51] [52], thus making sparse coding a plausible model characterizing the visual cortex [52], [53]. Signals of interest are not necessarily sparse in the canonical basis; however, they can have a concise representation when expressed in a convenient basis. The traditional signal acquisition processing divides the sampling and compression into two separate processes which samples a lot of unnecessary information and results in inefficiency for sparse signals. Compressed sensing overcomes these inefficiencies by directly acquiring the compressed signal representation from fewer samples or measurements than traditional methods use.

2.2 Compressed Sensing

Traditional reconstruction approaches follow the Shannon sampling theorem [54], which states that the sampling rate must be twice the highest frequency [55]. The theory of compressive sensing (CS) also known as compressed sensing, compressive sampling or sparse recovery provides a fundamentally new approach to data acquisition. CS relies on the empirical observation that many types of signals or images can be well-approximated by a sparse expansion in terms of a suitable basis, that is, by only a small number of non-zero coefficients. This forms the key to the efficiency of many lossy compression techniques such as JPEG, MP3 [55]. It predicts that instead of acquiring an entire Nyquist ensemble of signal samples, CS can reconstruct sparse signals from a small number of (random or deterministic) linear measurements via linear regression, convex optimization or greedy recovery algorithms.

Compressed Sensing (CS) was first proposed for image compression based on the assumption that most fields on images can be sparsely represented using wavelets. CS theory can sample sparse high-bandwidth signals at a sub-Nyquist rate and exploit the sparsity of high dimensional signals to enable the reconstruction of signals from non-uniform measurements. The number of required measurements depends on the sparsity of the signals rather than the bandwidth. With the compressed measurements, it is possible to reconstruct the original signal by solving a convex optimization problem.

A discrete-time signal x can be represented in terms of an orthonormal basis of

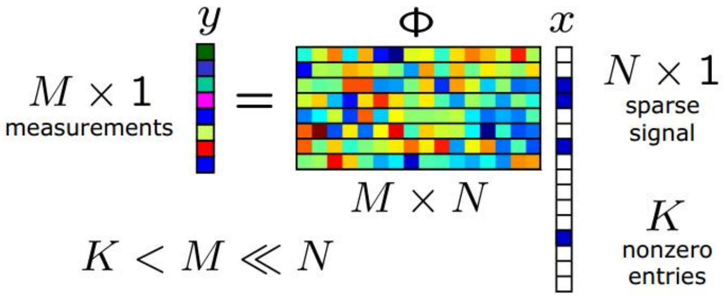

Figure 2.1: Compressive Sensing Measurement Process [4].

$N \times 1$ vectors $\Psi_{i=1}^{N}$ as follows:

$$x = \sum_{i=1}^{N} s_i \Psi_i \tag{2.1}$$

where s_i is the coefficient sequence of x. This can be further simplified as $x = \Psi s$ where s is $N \times 1$ vector and Ψ is $N \times N$ matrix with Ψ_i as columns. Signal x has a K-sparse expansion if only K of the entries in are non-zero and (N-K) are zero. Assuming that the signal $x \in \mathbb{R}^N$ is K-sparse and the sensing system acquires $M < N$ linear and non-adaptive measurements, the process can be mathematically represented as

$$y = \Phi x = \Phi \Psi s = \Theta s \tag{2.2}$$

where $\Theta = \Phi\Psi$ is $M \times N$ matrix, Ψ is $M \times N$ measurement matrix with Ψ_j^T as rows, $\Phi \in \mathbb{R}^{M \times N}$ is fixed represents a dimensionality reduction and $y \in \mathbb{R}^M$ represents the measurement vector.

2.3 Dictionary Learning

The dictionary learning problem is closely related to the CS but arises in a different context, where the main goal is to find compact and meaningful signal representations and correspondingly use them in signal and image processing tasks, such as denoising and classification. Dictionary learning aims to adapt the dictionary to better fit the task-specific model [56]. It needs a succinct set of atoms to best represent each signal in training set with defined sparsity constraints and it can be defined as follows:

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{y}_N] \in \mathbb{R}^{n \times N}$ be a set of n-dimensional N input signals. The dictionary $\mathbf{D}$ which sparsely represents $\mathbf{X}$ and corresponding sparse codes $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$ can be learned by solving the optimization problem

$$< \mathbf{D}, \mathbf{A} >= \arg\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_2^2\,, \quad \text{s.t.}\forall_i, \|a_i\|_0 \leq T \tag{2.3}$$

where $\mathbf{T}$ is sparsity constraint factor. $||.||_2$ denotes the l_2 norm of a vector. (The l_2 norm for for a vector y of n dimensions can be defined as $\sqrt{\sum_{i=1}^{n} y_i^2})$ $||.||_0$ is the l_0 norm of a vector which counts the non zero elements in a vector. Solving a l_0 problem is a NP hard problem and algorithms for determining approximating solutions have been extensively investigated and greedy algorithms such as Matching Pursuit algorithms [57], FOCUSS [58] and l_1 minimization were subsequently introduced as practical alternatives.

The l_0 norm can be relaxed to the l_1 norm, which for a n dimensional vector x can be defined as $\sum_{i=1}^{n} |x_i|$. The optimization problem can be defined as:

$$< \mathbf{D}, \mathbf{A} >= \arg\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_2^2\,, \quad \text{s.t.}\forall_i, \|a_i\|_1 \leq T \tag{2.4}$$

The constraint in equation 2.4 can be further relaxed using the Lagrangian method and thus can be re-written as

$$< \mathbf{D}, \mathbf{A} >= \arg\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_2^2\,, + \lambda \|a_i\|_1 \tag{2.5}$$

where λ is the Lagrangian multiplier and equation 2.5 is called LASSO (Least

Angle Shrinkage and Selection Operator) [59]. The model parameters **W** and dictionary **D** can be jointly learned by solving the following:

$$< \mathbf{D}, \mathbf{W}, \mathbf{A} >= \arg \min_{\mathbf{D},\mathbf{W},\mathbf{A}} (\|\mathbf{X} - \mathbf{DA}\|_2^2 + \sum_i \mathcal{L}\{h_i, f(x_i, W)\}$$

$$+\lambda_1 \|\mathbf{W}\|_F^2), \text{ s.t.} \forall_i, \|x_i\|_0 \leq T \tag{2.6}$$

where h_i is label of y_i, $\mathcal{L}$ is a classification loss function and λ_1 is a regularization parameter.

There are two approaches to decide on the sparsifying basis, i.e., the D matrix. One method is to exploit the mathematical model of the signal and accordingly choose off the shelf dictionaries like wavelets, contour-lets and the like or the other way is to learn a dictionary that does best on the training data for the given task at hand. Considering the latter methodology using dictionary, a typical technique to minimize the above objective is by iteratively solving for sparse representations based on the dictionary and updating the dictionary given the sparse codes, until the constraint is met. A dictionary learned from the training data when compared to a predetermined fixed dictionary generally leads to a more compact representation, thereby achieving enhanced results in many practical computer vision applications.

2.4 Review of Dictionary Learning algorithms

Several compact dictionary learning approaches have been developed in [39], [56], [60], [61], [62], [63], [64], for the task of learning a dictionary from data samples. In [39], the dictionary is constructed by manual selection of training samples and in [60], k-means clustering is used to group features from training samples. A few of the most prevalent dictionary learning classifiers are the SRC classifier (Sparse Reconstruction

Classifier), K-SVD, D-KSVD (Discriminative KSVD) and the label consistent version (LC- KSVD) and Fisher Discrimination Dictionary Learning (FDDL).

In SRC [65] procedure, the training data is used to form a dictionary and the classification of test data is achieved through finding its sparse coefficients with respect to this dictionary. Thus, the dictionary is critical for the performance and the manual selection of dictionary atoms would be impractical for large datasets. In [56], K-SVD algorithm is introduced that aims to find a dictionary and a sparse matrix that minimize the representation error. The KSVD algorithm is an iterative method that alternates between sparse coding and dictionary update steps to better fit the data and is solved by alternating between the following two steps

- Sparse Coding Step: Use any pursuit algorithm to compute the representation vector for each input signal x_i by approximating the solution of

$$\arg\min_{\mathbf{a_i}} \|\mathbf{x_i} - \mathbf{D}\mathbf{a_i}\|_2^2 \tag{2.7}$$

- Codebook Update Stage: For each column $k = 1.....K$ in dictionary D^{J-1}, update the dictionary column by computing the overall representation error E_k as

$$E_k = X - \sum_{j \neq k} d_j a^j \tag{2.8}$$

The method of optimal directions [61] is also another iterative training algorithm inspired by the Generalized Lloyd Algorithm (GLA) [66] which follows the k-means outline and updates the dictionary efficiently during the learning stage. In [67], a tree-structured sparse regularization is employed to exploit the semantic relationships between dictionary elements so as to learn structured dictionaries in a hierarchy.

These methods have proven to provide state-of-the-art results in image processing applications, including infilling missing pixels, image compression and reconstruction.

The design of supervised discriminative dictionaries has also gained significant attention in recent years. Construction of such discriminative dictionaries encompasses modification of the function so as to enforce sparsity and maintain discrimination. This is usually done by introducing a discriminative cost function [68], [69], [70], [62], linear predictive classification error [63], [64], fisher discrimination criterion [71], [40], [72], and logistic loss function [69], [73] which essentially enforces separability among dictionary atoms belonging in different classes.

With the aim of promoting discrimination between classes, a classification error is incorporated into the objective function in [63]. However, it does not guarantee the discriminability of the resultant sparse codes while using a small-size dictionary. In [74], [62], [68], [75], a dictionary is learned for each specific class and classification is done based on the corresponding reconstruction error but not by leveraging the sparse codes. However, the drawback here is that class-wise sparse coding during testing stage becomes time-consuming for a large number of classes.

A popular discriminative dictionary learning procedure is the D-KSVD wherein a classification regularization term is added to the reconstruction term. The resulting optimization problem can be defined as :

$$\min_{\mathbf{D},\mathbf{A},\mathbf{W}} \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \|\mathbf{H}-\mathbf{W}\mathbf{A}\|_F^2 + \lambda||\mathbf{W}||_F \tag{2.9}$$

Here $||H - WA||_F^2$ represents the classification error term, W denotes the classifier parameter and H represents the class labels of input signal where each column of H has a one at the i^{th} position if the sample X_i belongs to class i.

In LC-KSVD, a discriminative term is added to the classification error term. At each iteration, the algorithm tries to find the discriminative sparse code with a small classification error thereby reducing the reconstruction error at the same time. The LC-KSVD formulation can be written as:

$$\min_{\mathbf{D},\mathbf{A},\mathbf{Z},\mathbf{W}} \|\mathbf{X}-\mathbf{DA}\|_F^2 + \|\mathbf{Q}-\mathbf{ZA}\|_F^2 + \|\mathbf{H}-\mathbf{WA}\|_F^2 \tag{2.10}$$

where Z is linear transformation matrix and Q represents the discriminative sparse codes of input signal. $Q = [q_1.....q_N]$ will be considered as a discriminative sparse code corresponding to an input signal X_i, if the nonzero values of q_i occur at same indices where the input signal and the dictionary item have the same label.

In another work [76], the dictionary learning process is wrapped inside a boosting procedure for learning multiple dictionaries. In [77], Ramirez et al. demonstrate learning of class-specific dictionaries using an incoherence promoting term, which encourages class-specific dictionaries to be independent. The inter-related dictionary learning algorithm proposed in [71] learns multiple dictionaries for visually correlated object categories. The common properties of the group are symbolized by a common shared dictionary, and category- specific properties are symbolized by multiple category-specific dictionaries.

Non-Linear kernel-based dictionary learning algorithms have also been proposed in the literature [78], [79]. By means of a predetermined kernel function, these algorithms essentially map the input data onto a higher dimensional feature space. Sparse codes and dictionaries are later trained on these feature space for better representation and discrimination.

Traditional Dictionary learning algorithms aimed only to minimize the reconstruction error, whereas the Dictionary Pair Learning (DPL) model here targets at classification by learning class-specific dictionaries for data representation for the activity classes. A regularizer term is introduced in the DPL objective function which enhances the discriminative capability of learned dictionaries by generating the minimum reconstruction error for the accurate class. The similarity-constrained term in DPL model projects each descriptor into its local coordinate system which captures the correlations between similar descriptors by sharing bases and the test sample is classified into the class whose dictionary generates the minimum reconstruction error, thereby making it remarkably more discriminative.

2.5 Dictionary Pair Learning

The dictionary pair learning objective is formulated such that it enhances the class-discrimination capabilities of individual atoms rather than that of the subspaces they generate. This renders the designed dictionaries especially suitable for fast classification of query images with very sparse approximations. The DPL-framework jointly learns a synthesis dictionary and an analysis dictionary, i.e., this pair of dictionaries works together to perform representation and discrimination simultaneously. The rationale behind this strategy is that using linear projection instead of nonlinear sparse coding not only improves the recognition rate but also makes it computationally more efficient when compared to using l_1 and l_0 norm for regularizing representation coefficients.

Prior dictionary learning algorithms have made use of a predefined analytical dictionary (e.g., wavelet dictionary, Gabor dictionary) to represent a signal, thereby producing the representation coefficients by simple inner product operations. Such fast, explicit coding makes an analytical dictionary attractive in image representation but less effective in modeling the complex local structures of natural images. Traditionally either l_0-norm or l_1-norm have been used to regularize the representation coefficients since sparser coefficients are more likely to produce better classification results. However using these sparsity regularizations is still a computational burden, making training and testing inefficient. In contrast, DPL classification algorithm obtains the representation coefficients by linear projection instead of nonlinear sparse coding. Thus, DPL framework is a promising approach for learning a synthesis dictionary and an analysis dictionary jointly for pattern classification. The analysis dictionary is trained to generate discriminative codes by efficient linear projection, while the synthesis dictionary is trained to achieve class-specific discriminative reconstruction.

To define the discriminative dictionary learning, we denote a set of p-dimensional training samples from K classes by $X = [X_1, ..., X_k, ..., X_K]$, where $X_k \in \mathbb{R}^{p \times n}$ is the training sample set of class k, and n is the number of samples of each class. Discriminative dictionary learning (DL) methods aim to learn an effective data representation model from X for classification tasks by exploiting the class label information of training data, and can be formulated under the following framework:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_F^2 + \lambda||\mathbf{A}||_p + \Psi(\mathbf{D}, \mathbf{A}, \mathbf{Y}) \tag{2.11}$$

Here $\lambda \geqslant 0$ is a scalar constant; $||.||_F$ denotes the Frobenius norm of a matrix. The Frobenius norm of a matrix A can be defined as $Tr\sqrt{AA^T}$, where Tr denotes the trace of a matrix. Also, Y represents the class label matrix of samples in X; D is the synthesis dictionary to be learned; and A is the coding coefficient matrix of X over D. In the training model given in equation (11), the data fidelity term $\|\mathbf{X} - \mathbf{DA}\|_F^2$ ensures the representation ability of D; $||\mathbf{A}||_p$ is the l_p-norm regularizer on A; and $\Psi(\mathbf{D}, \mathbf{A}, \mathbf{Y})$ stands for some discrimination promotion function that ensures the discrimination power of D and A.

If when using an analysis dictionary denoted by $P \in \mathbb{R}^{mK \times p}$, the code A can be analytically obtained as A = PX, then the representation of X becomes efficient. Based on this idea, the DPL model learns an analysis dictionary P together with the synthesis dictionary D, leading to the following DPL model:

$$P^*, D^* = \arg\min_{\mathbf{P},\mathbf{D}} \|\mathbf{X} - \mathbf{DPX}\|_F^2 + \Psi(\mathbf{D}, \mathbf{P}, \mathbf{X}, \mathbf{Y}) \tag{2.12}$$

Here Ψ (D, P, X, Y) is the discrimination function; and D and P form a dictionary pair where the analysis dictionary P is used to code X analytically, and the synthesis dictionary D is used to reconstruct X.

The learned structured synthesis dictionary $D = [D_1, ..D_k, ., D_K]$ and the structured analysis dictionary $P = [P_1, ..., P_k......P_K]$ form a sub-dictionary pair corresponding to class k. Recent studies on sparse subspace clustering [80] have shown that a sample can be represented by its corresponding dictionary if the signals satisfy certain incoherence conditions. Thus, by using the structured analysis dictionary P, we want the sub-dictionary P_k to project the samples from class i,$i \neq k$ to a nearly null space:

$$\mathbf{P}_k, \mathbf{X}_i = 0, \forall \;\; k \neq i \tag{2.13}$$

Then by means of equation (1.5) the coefficient matrix PX will be nearly block diagonal. Using the structured synthesis dictionary D, we expect the sub-dictionary D_k can properly reconstruct the data matrix X_k from its projective code matrix $P_k X_k$; that is, the dictionary pair should minimize the reconstruction error:

$$\min_{\mathbf{P},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k \mathbf{P}_k \mathbf{X}_k\|_F^2 \tag{2.14}$$

Based on the above analysis, we readily have the following DPL model:

$$P^*, D^* = \arg\min_{\mathbf{P},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k \mathbf{P}_k \mathbf{X}_k\|_F^2 + \lambda ||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2 \tag{2.15}$$

Here $\overline{X_k}$ denotes the complementary data matrix of X_k in the whole training set X, $\lambda > 0$ is a scalar constant, and d_i denotes the i^{th} atom of synthesis dictionary D.

In the DPL model, the analysis sub-dictionary P_k is trained to produce small coefficients for samples from classes other than k, and thus can only generate significant coding coefficients for samples from class k. Meanwhile, the synthesis sub-dictionary D_k is trained to reconstruct the samples of class k from their projective coefficients P_kX_k, so the residual a_i will be small. Conversely, since P_kX_i will be small and D_k is not trained to reconstruct X_i, the residual a_i will be much larger. Thus in the testing phase, if the query sample y is from class k, its projective coding vector P_k will more likely be large, while its projective coding vectors P_i will be small. Consequently, any reconstruction residual with respect to P_k will be much smaller than the residuals with respect to P_i. A class-specific reconstruction residual can be used to identify the class label of y, and we can naturally have the resulting classifier associated with the DPL model:

$$identity(y) = \arg\min_{\mathbf{i}} \|\mathbf{y} - \mathbf{D}_i\mathbf{P}_i\mathbf{y}\|_2 \qquad (2.16)$$

2.6 Conclusion

In this chapter, we have briefly reviewed the theory of the sparse representation , compressive sensing and dictionary learning. An overview of existing state-of-the-art dictionary learning algorithms in literature has been discussed. In addition, the dictionary pair learning algorithm which aims to minimize the reconstruction loss by introducing a fidelity term involving both analysis and synthesis dictionary has been discussed briefly. One promising property of the DPL model is that it is appropriate for balancing the representation and discrimination to boost recognition performance. The next chapter introduces a centralized class specific dictionary pair learning for sensor-based activity recognition which makes the sparse codes belonging to the same class concentrated thereby resulting in improved recognition.

Chapter 3

CLASS SPECIFIC CENTRALIZED DICTIONARY PAIR LEARNING FOR ACTIVITY RECOGNITION USING WEARABLE SENSORS

3.1 Introduction

In this chapter, we present a unified dictionary pair learning framework by incorporating a centralized class specific regularizer term to solve the wearable sensor-based classification problem. In the dictionary pair learning framework, the training samples of each class contribute identically to the dictionary, thus generating a dictionary having training samples corresponding to each class. This can result in instability and high residual error, which is detrimental to the recognition performance. Using traditional class specific dictionary learning approaches [81] does allow learning a dictionary for each class, but might result in interdependence in sparse codes and erroneous discrimination. The main contribution is to explicitly incorporate centralized class specific sparse codes to the dictionary pair learning objective function to obtain superior classification performance by making sparse codes in the same class concentrated [82]. A new alternate minimization algorithm incorporated with a $l_1 - l_s$ minimization method is developed to facilitate convergence of the non-convex objective function. Experiments on the standard sensor-based activity recognition datasets demonstrate the effectiveness of the proposed method. To the best of our knowledge, dictionary learning frameworks, and specifically centralized class dependent dictionary pair learning frameworks, have not to date been used in wearable sensor-based applications. Consequently, our novel dictionary learning-based framework algorithm will instigate future research on this method's potential applicability for accurate sensor-based data classifications and other physiological-signal classifications.

3.2 Proposed Methodology

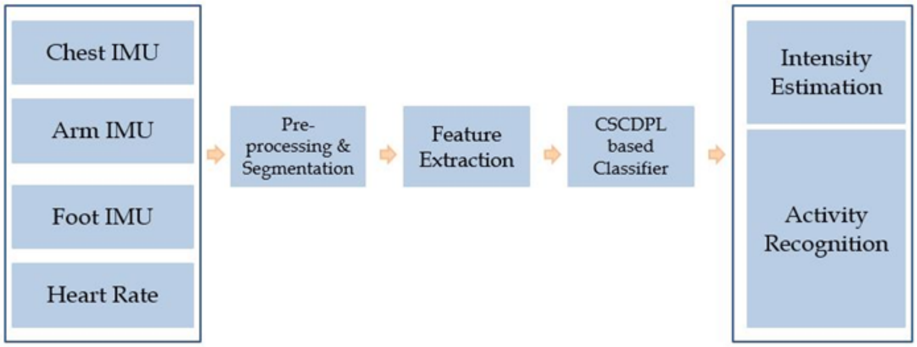

Figure 3.1: Proposed Class Specific Centralized Dictionary Learning Framework

The proposed framework involves two steps: data processing and recognition. Data processing incorporates preprocessing, segmentation and feature extraction stages. In the preprocessing stage, raw sensory data is synchronized, timestamped, and labeled. During segmentation, this collected data is segmented with a sliding window, using a defined window size and signal features extracted from the segmented data in the feature extraction stage. A dictionary pair learning classification algorithm jointly learns a synthesis dictionary and an analysis dictionary to attain the objective of signal representation and discrimination [83]. To explain discriminative dictionary learning, a set of p-dimensional training samples from K classes can be defined as $X = [X_1, ..., X_k, ..., X_K]$, where $X_k \in \mathbb{R}^{p \times n}$ is the training sample set of class k, and n is the number of samples of each class. Discriminative dictionary learning (DL) methods learn an efficient data representation model from X for classification by employing the class label information of training data, and can be formulated as:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{DA}\|_F^2 + \lambda||\mathbf{A}||_p + \Psi(\mathbf{D}, \mathbf{A}, \mathbf{Y}) \tag{3.1}$$

Here $\lambda \geqslant 0$ is a scalar constant; Y denotes the class label matrix of samples in X; D is the synthesis dictionary to be learned; and A is the coding coefficient matrix of X over D. The data fidelity term $\|\mathbf{X} - \mathbf{DA}\|_F^2$ in the training model (equation (3.1)) ensures the representation ability of D; $||\mathbf{A}||_p$ is the l_p-norm regularizer on A; and $\Psi(\mathbf{D}, \mathbf{A}, \mathbf{Y})$

represents discrimination promotion function that ensures discrimination power of D and A [83].

An analysis dictionary generates discriminative codes by efficient linear projection and can be defined by $P \in \mathbb{R}^{mK \times p}$. Using the analysis dictionary, the code A can be analytically represented as A = PX, making the representation of X efficient. Based on this concept, the DPL model learns an analysis dictionary P together with the synthesis dictionary D, leading to the following DPL model [83]:

$$P^*, D^* = \arg\min_{\mathbf{P},\mathbf{D}} \|\mathbf{X} - \mathbf{DPX}\|_F^2 + \Psi(\mathbf{D}, \mathbf{P}, \mathbf{X}, \mathbf{Y}) \tag{3.2}$$

Here Ψ (D, P, X, Y) represents the discrimination function; and D and P form a dictionary pair where the analysis dictionary P is used to analytically code X, and the synthesis dictionary D is used to reconstruct X. The learned structured synthesis dictionary $D = [D_1, ..D_k, ., D_K]$ and the structured analysis dictionary $P = [P_1, ..., P_k ......P_K]$ form a sub-dictionary pair corresponding to class k. Thus incorporating the structured analysis dictionary P, we ensure that the sub-dictionary P_k projects the samples from class i,$i \neq k$ to a nearly null space thereby making the coefficient matrix PX nearly block diagonal. By adopting variable matrix A to relax the non-convex problem, the following DPL model can be defined as:

$$\begin{aligned} P^*, A^*, D^* = \arg\min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2 \\ + \tau \|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2 + \lambda || \mathbf{P}_k, \overline{\mathbf{X_i}} ||_F^2 \end{aligned} \tag{3.3}$$

Here $\overline{X_k}$ represents the complementary data matrix of X_k in the training set X, $\lambda > 0$ denotes a scalar constant, and d_i is the ith atom of synthesis dictionary D.

3.2.1 Proposed DPL model with class specific centralized regularizer term

The objective is to incorporate a centralized class specific regularizer term [84] to obtain a unified dictionary pair learning framework. In DPL model, training samples of each class contribute equivalently to the dictionary, thus generating a dictionary consisting of training samples in corresponding class, resulting in instability and higher residual error. The main contribution here is to explicitly incorporate centralized class specific sparse codes [84] to the dictionary pair learning objective function, thereby, making sparse codes in the same class concentrated.

To attain this objective, we denote the mean of each row of sparse code A as E(A). The regularizer term can be formulated as :

$$R(A_k) = \eta \sum_{n=1}^{N} \|(\mathbf{A}_k).n - E(\mathbf{A}_k)\|_2^2 \tag{3.4}$$

where η is the tradeoff parameter between the reconstruction error and the degree of deviation from the sparse code to their centers and $A_{\cdot n}$ represent the nth column of a sparse matrix A. Incorporating the class specific regularizer term to the DPL objective function, equation (3.3) can be formulated as:

$$\begin{aligned} P^*, A^*, D^* = \arg\min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k\mathbf{A}_k\|_F^2 + \tau\|\mathbf{P}_k\mathbf{X}_k - \mathbf{A}_k\|_F^2 \\ +\lambda||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2 + \eta \sum_{n=1}^{N} \|(\mathbf{A}_k).n - E(\mathbf{A}_k)\|_2^2 \end{aligned} \tag{3.5}$$

In order to solve this equation, we integrate the alternating direction method of multipliers with the $l_1 - l_s$ method to facilitate convergence. The alternating direction method of multipliers (ADMM) solves convex optimization problems by fixing some variables and solving for the other variable, thereby decomposing the problem into smaller sub-problems making each of them easier to handle [83]. The minimization can be alternated between the following steps:

1:**Fix A, update P and D**

$$P^* = \arg\min_{\mathbf{P}} \sum_{k=1}^{K} \tau \|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2 + \lambda ||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2 \tag{3.6}$$

$$D^* = \arg\min_{\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2 \tag{3.7}$$

The closed form solution for P can be obtained as:

$$P^* = \tau A_k X_k^T (\tau X_k X_k^T + \lambda \bar{X}_k \bar{X}_k^T + YI)^{-1} \tag{3.8}$$

The closed form solution for D can be obtained by introducing a variable S as in DPL [83]

$$D^{r+1} = \arg\min_{\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2 + \rho \|\mathbf{D}_k - \mathbf{S}_k^r + \mathbf{T}_k^r\|_F^2 \tag{3.9}$$

The solution for D and P are the same as in the DPL framework using ADMM algorithm.

2. **Fix D and P, update A**

$$\begin{aligned} A^* = \arg\min_{\mathbf{A}} \sum_{k=1}^{K} & tr[(\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k)^\top (\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k)] \\ & + tr[\tau(\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k)^\top (\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k)] \\ & + \eta \sum_{n=1}^{N} \|(\mathbf{A}_k)_{\cdot n} - E(\mathbf{A}_k)\|_2^2 \end{aligned} \tag{3.10}$$

Here, tr represents the trace of a matrix and using trace properties $tr(A) = tr(A^T)$ and $tr(A'A) = ||A||_F^2$. We use $l_1 - l_s$ optimization method to solve for A. Ignoring the constant terms, the function can be simplified using $l1 - ls$ optimization method.

$$A^* = \arg\min_{\mathbf{A}} \sum_{k=1}^{K} tr[-2(\mathbf{X_k}^\top \mathbf{D_k} + \tau \mathbf{X_k}^\top \mathbf{P_k}^\top)\mathbf{A_k}$$
$$+\mathbf{A_k}^\top(\mathbf{D_k}^\top \mathbf{D_k} + \tau I)\mathbf{A_k}] + \eta \sum_{n=1}^{N} \|(\mathbf{A}_k).n - E(\mathbf{A}_k)\|_2^2 \tag{3.11}$$

Defining $t = (\mathbf{X_k}^\top \mathbf{D_k} + \tau \mathbf{X_k}^\top \mathbf{P_k}^\top)$ and $S = (\mathbf{D_k}^\top \mathbf{D_k} + \tau I)$ and using property $tr(A^\top BA) = \sum_{n=1}^{N} A_{\cdot n}^\top B(A_{\cdot n})$, equation 3.11 can be rewritten as

$$A^* = -2\sum_{n=1}^{N} \mathbf{t_n}_{\cdot}(A_k)_{\cdot n} + \sum_{n=1}^{N} (\mathbf{A_k})_{\cdot \mathbf{n}}^\top \mathbf{S}(A_k)_{\cdot n} +$$
$$\eta \sum_{n=1}^{N} [(\frac{N-1}{N})^2 (\mathbf{A}_k)_{\cdot n}^\top (\mathbf{A}_k)_{\cdot n} - 2\frac{N-1}{N^2}(\mathbf{A}_k)_{\cdot n}^\top \sum_{m=1,m\neq n}^{N} (\mathbf{A}_k)_{\cdot m}] \tag{3.12}$$

The objective function now reduces to equation (3.13)

$$f(A_k)_{ln} = (A_k)_{ln}^2 (S_{ll} + \eta[\frac{N-1}{N}]^2) + 2(A_k)_{ln}(\sum_{q=1q\neq l}^{R} S_{lq}(A_k)_{qn})$$
$$-2\eta(A_k)_{ln}[\frac{N-1}{N^2}\sum_{m=1,m\neq n}^{N}(A_k)_{mn}] - 2(A_k)_{ln} t_{ln}^T \tag{3.13}$$

Here $(A_k)_{ln}$ denotes all elements in A_k except the element in the 'l' th row and the 'n' th column. Defining

$$d_{ln} = t_{ln}^T + \eta[\frac{N-1}{N^2}\sum_{m=1,m\neq n}^{N}(A_k)_{lm}] \tag{3.14}$$

$f(A_k)_{ln}$ becomes a piece-wise parabolic function as in [84]. Adopting the convexity and monotonic property of the parabolic function and the problem transformation defined in [84] , we have the closed-form solution as function reaches the minimum at the unique point.

$$(A_k)_{ln} = \frac{d_{ln} - [S(\hat{A_k}^{ln})]_{ln}}{S_{ll} + \eta[\frac{N-1}{N}]^2} \tag{3.15}$$

In the testing phase, the analysis sub-dictionary P_k is trained to produce small coefficients for samples from classes other than k, and thus can only generate significant coding coefficients for samples from class k. Meanwhile, the synthesis sub-dictionary D_k is trained to reconstruct the samples of class k from their projective coefficients $P_k X_k$, i.e.,reconstruction residual will be small. Conversely, since $P_k X_i$ will be small and D_k is not trained to reconstruct X_i, the reconstruction residual a_i will be much larger. Thus, if the query sample y is from class k, its projective coding vector P_k will more likely be large, while its projective coding vectors P_i will be small. Therefore, the class-specific reconstruction residual is used to identify the class label of testing sample.

3.3 Experimental results and discussion

We evaluate our approach on four realistic activity recognition tasks from following databases: the PAMAP2 -Intensity Estimation Task, PAMAP2 Activity Recognition Task, Smartphone-Based Human Activity Recognition (SBHAR) and Wireless Sensor Data Mining (WISDM). We compare our approaches with Adaboost [3], C4.5 decision tree [3] and other state-of-the-art algorithms.

3.3.1 PAMAP2 Database

The proposed algorithm is evaluated over the activity recognition and intensity estimation classification problems defined on the recently released PAMAP2 Physical Activity Monitoring Data Set. Briefly, this dataset captures 18 physical activities performed by 9 subjects wearing 3 IMUs (Inertial measurement unit) and a HR (heart rate) monitor. The raw sensory data is first synchronized, timestamped, and labeled in the preprocessing stage and 3D-acceleration and heart rate data are acquired. During segmentation, this collected data is segmented with a sliding window, using a window size of 512 samples. During the feature extraction stage, signal features extracted from the segmented 3D-acceleration data are calculated for each of the three axis separately and for the 3 axes together. The inclusion of Heart Rate (HR) monitor data with the commonly used inertial sensors proved specifically useful for physical activity intensity estimation [3]. Mean and gradient are calculated on both the raw and normalized heart rate signals from the HR data. Overall, a total of 137 basic features are computed: 133 features from Inertial Measurement Unit (IMU) acceleration data and 4 features from HR data [3].

The activity classification task consists of 15 different activity classes represented as lie, sit, stand, walk, run, cycle, Nordic walk, drive car, ascend, descend stairs, vacuum, iron, fold laundry, clean house, play soccer and jump rope. This classification task is referred to as the PAMAP2 Activity Recognition (PAMAP2-AR) task. The intensity-estimation classification task aims to distinguish activities of light, moderate, and vigorous effort based on the MET of the various physical activities, as provided by [19] and is referred to as the PAMAP2 Intensity Estimation (PAMAP2 - IE) task. Therefore, intensity classes are defined as activities of light effort ($<$ 3.0 METs) (lie, sit, stand, drive a car, iron, fold laundry, clean house, watch TV, work at a computer), moderate effort (3.0-6.0 METs) (walk, cycle, descend stairs, vacuum and Nordic walk), or vigorous effort ($>$ 6.0 METs) (run, ascend stairs, jump rope, play soccer).

Using these two defined classification tasks, the proposed method is compared with C4.5 Decision Tree and Adaboost algorithms [3]. For the evaluation procedure, we randomly selected 75% of the data for training and 25% for testing and averaged recognition results over 10 repetitions. In addition to providing classification accuracy, we also report on the average computation time for classifying one query activity of competing algorithms in the experiments.

3.3.1.1 PAMAP2 -Results on Intensity Estimation Task

First presented are classification results over an intensity estimation task defined on the PAMAP2 dataset. In [85], the C4.5 decision tree algorithm and Adaboost classifier were tested on the PAMAP2 dataset and demonstrated an accuracy of 70.07%

Reference/Algorithm	Class 1	Class 2	Class 3
Class 1	11655	1338	331
Class 2	621	6539	402
Class 3	5	741	1368

Table 3.1: Overall confusion matrix using class specific centralized dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3
Class 1	87.47%	10.04%	2.48%
Class 2	8.21%	86.47%	5.31%
Class 3	0.33%	9.31%	90.35%

Table 3.2: Overall confusion matrix in % using class specific centralized dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes.

Method	Proposed	Adaboost [3]	C4.5 Decision Tree [3]
Accuracy	**85.89%**	73.93%	70.07%

Table 3.3: Recognition results on PAMAP2-IE Dataset.

Method	Proposed	Adaboost	C4.5 Decision Tree
Computation time	0.24s	10.82s	2.39s

Table 3.4: Computation time for classifying one query activity on PAMAP2-IE Dataset.

and 73.93% respectively. The overall confusion matrix using the proposed framework for the three intensity estimation tasks (*i.e.*, light (Class 1), moderate (Class 2) and vigorous (Class 3) tasks) is given in Table 3.1. An independent performance assessment of the proposed framework results in an accuracy of 85.89% on the AR-IE task, demonstrating that our framework outperforms the C4.5 decision tree and AdaBoost classifiers (Table 3.3). As shown in Table 3.3, the proposed framework achieves the highest recognition rate of 85.89%. As shown in Table 3.4, the proposed algorithm is approximately more than 5 times faster than Adaboost. Clearly, the proposed approach outperforms other methods.

Reference/ Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Class 11	Class 12	Class 13	Class 14	Class 15
Class 1	1660	2	3	19	12	0	0	0	1	0	0	0	30	0	1
Class 2	1	1257	134	73	18	4	0	4	0	3	1	0	130	25	5
Class 3	0	83	1262	126	33	1	4	0	6	2	2	0	156	20	8
Class 4	1	29	128	1379	103	0	0	0	0	0	3	0	506	40	0
Class 5	0	0	10	170	693	5	0	5	0	2	8	0	634	11	19
Class 6	0	0	0	0	0	409	49	199	101	0	0	0	15	0	6
Class 7	0	0	0	0	0	73	358	49	102	7	0	1	41	0	0
Class 8	0	0	0	0	0	77	35	1825	224	28	0	0	4	0	0
Class 9	0	0	0	0	0	4	21	111	1568	0	0	0	2	0	3
Class 10	0	4	1	3	10	8	7	25	78	1312	0	0	19	0	5
Class 11	0	0	0	0	14	0	0	0	6	0	791	0	18	0	6
Class 12	0	0	0	0	30	0	33	0	35	0	0	2668	2	0	0
Class 13	0	9	43	208	350	12	6	1	9	10	2	0	1056	34	8
Class 14	0	1	21	322	145	0	0	0	0	0	2	0	353	57	0
Class 15	0	0	0	0	18	43	2	16	46	0	8	3	17	0	242

Table 3.5: Confusion matrix using proposed framework on PAMAP2 -AR dataset.

Reference/ Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Class 11	Class 12	Class 13	Class 14	Class 15
Class 1	96.06%	0.11%	0.17%	1.09%	0.69%	0%	0%	0%	0.05%	0%	0%	0%	1.73%	0%	0.05%
Class 2	0.06%	75.95%	8.09%	4.4%	1.08%	0.24%	0%	0.24%	0%	0.18%	0.06%	0%	7.85%	1.51%	0.3%
Class 3	0%	4.87%	74.10%	7.39 %	1.93%	0.058%	0.23%	0%	0.35%	0.11%	0.11%	0%	9.16%	1.17%	0.46%
Class 4	0.04%	1.32%	5.84%	62.99%	4.7%	0%	0%	0%	0%	0%	0.13%	0%	23.11%	1.82%	0%
Class 5	0%	0%	0.64%	10.91%	44.55%	0.32%	0%	0.32%	0%	0.12%	0.51%	0%	40.71%	0.70%	1.22%
Class 6	0%	0%	0%	0%	0%	52.5%	6.29%	25.54%	12.96%	0 %	0%	0%	1.92%	0%	0.77%
Class 7	0%	0%	0%	0%	0%	11.56%	56.73%	7.76%	16.16%	1.1%	0%	0.15%	6.49%	0%	0%
Class 8	0%	0%	0%	0%	0%	3.51%	1.59%	83.21%	10.21%	28%	0%	0%	0.18%	0%	0%
Class 9	0%	0%	0%	0%	0%	0.23%	1.22%	6.49%	91.74%	0%	0%	0%	0.11%	0%	0.17%
Class 10	0%	0.27%	0.06%	0.20%	0.67%	0.54%	0.47%	1.69%	5.29%	89.13%	0%	0%	1.29%	0%	0.33%
Class 11	0%	0%	0%	0%	1.67%	0%	0%	0%	0.71%	0%	94.73%	0%	2.15%	0%	0.72%
Class 12	0%	0%	0%	0%	1.08%	0%	1.19%	0%	1.26%	0%	0%	96.38%	0.07%	0%	0%
Class 13	0%	0.51%	2.45%	11.89%	20.02%	0.68%	0.34%	0.05%	0.51%	0.57%	0.11%	0%	60.41%	1.94%	0.45%
Class 14	0%	0.11%	2.33%	35.73%	16.09%	0%	0%	0%	0%	0%	0.22%	0%	39.17%	6.32%	0%
Class 15	0%	0%	0%	0%	4.55%	10.88%	0.5%	4.05%	11.6%	0%	2.02%	0.75%	4.30%	0%	61.26%

Table 3.6: Confusion matrix in % using proposed framework on PAMAP2 -AR dataset.

3.3.1.2 PAMAP2 - Results on Activity Recognition Task

To investigate the proposed framework's performance on PAMAP2− AR task, we computed its performance on 15 classes based activity-recognition tasks. We find that our framework outperforms the C4.5 decision tree with an accuracy of 73.17% versus 71.59%, and it gives competitive results when compared to an AdaBoost classifier on the PAMAP2 AR task (Table 3.7).

In addition to competitive accuracy, our framework provides the additional advantages of lower computation time for classifying a query activity. Further examination of our results indicate that averaged over 10 test runs, the confusion matrix of the best-performing classifier on the PAMAP2-AR task yields an overall accuracy of

73.17%, showing that some activities are recognized with high accuracies, such as lying, walking, or even distinguishing between ascending and descending. Overall, misclassifications, where activities belonging to one class are mistakenly classified as belonging to its neighboring classes, are lower.

In [85] and [3], the evaluation technique was leave-one-activity-out (LOAO) where an activity monitoring system is used on a previously unknown activity. Our framework also takes in data randomly. Evaluation is based on a completely unknown activity from an unknown user, while training is performed using a different activity with a different user in our random 75%-25% validation approach. These types of subject-independent and activity-independent validation techniques are preferred for physical activity monitoring since they provide results with more practical meaning. Using our framework, we can not only achieve good classifier performance but also eliminate the need of pre-training a particular activity for classification. Thus, our proposed method makes it possible to design a robust physical activity monitoring system that has the desired generalization characteristics.

Method	Proposed	Adaboost [3]	C4.5 Decision Tree [3]
Accuracy	**73.17%**	71.78%	71.59%

Table 3.7: Recognition results on PAMAP2-AR Dataset.

Method	Proposed	Adaboost	C4.5 Decision Tree
Computation time	1.52s	11.25s	9.39s

Table 3.8: Computation time for classifying one query activity on PAMAP2-AR Dataset.

Varying Training Data %	20	40	60	80	100
Accuracy in %	70.71%	72.56%	73.21%	73.48%	73.81%

Table 3.9: Performance of the proposed algorithm on the PAMAP2 - AR dataset with varying training data

3.3.2 SBHAR

This dataset is built from the recordings of 30 subjects performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors [86], [87]. This dataset includes six activities (walking straight, walking upstairs, walking downstairs, sitting, standing and laying) which are performed while wearing a smartphone (Samsung Galaxy S II) on the waist. The 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz were captured using the phone's embedded accelerometer and gyroscope. The sensor signals are pre-processed by applying noise filters and then sampled in fixed-width sliding windows of 2.56 sec and 50% overlap (128 readings/window). From each window, a vector of features was obtained by calculating variables from the time and frequency domain (e.g. mean, standard deviation, signal magnitude area, entropy, signal-pair correlation, etc.)

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	1877	29	0	0	0	0
Class 2	572	1190	15	0	0	0
Class 3	4	17	1918	0	0	5
Class 4	1	2	1	1588	51	79
Class 5	0	8	0	10	1320	68
Class 6	0	12	0	18	19	1495

Table 3.10: Overall confusion matrix using proposed framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	98.47%	1.52%	0%	0%	0%	0%
Class 2	32.18%	66.96%	0.84%	0%	0%	0%
Class 3	0.20%	0.87%	98.66%	0%	0%	0.25%
Class 4	0.05%	0.11%	0.05%	92.21%	2.96%	4.58%
Class 5	0%	0.56%	0%	0.71%	93.88%	4.83%
Class 6	0%	0.77%	0%	1.16%	1.23%	96.82%

Table 3.11: Overall confusion matrix in % using proposed framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes.

Method	Proposed	Multi-Class SVM [88]	Multi-Class HF SVM [88]	Naives Bayes [89]	Decision Tree [89]
Accuracy	**91.31%**	89.0%	89.3%	82.5%	86.8%

Table 3.12: Recognition results on SBHAR Dataset.

Varying Training Data %	20	40	60	80	100
Accuracy in %	91.2%	92.16%	92.35%	92.38%	92.59%

Table 3.13: Performance of the proposed algorithm on the SBHAR dataset with varying training data

We evaluate our proposed approach and compare with Naive's Bayes Classifier [88], Decision Tree [88], Multiclass HF SVM and other state-of-the-art approaches [88]. Our approaches outperform the state-of-the-art approaches as illustrated in Table 3.12. The confusion matrices for the CSCDPL approach is shown in Table 3.10. Our proposed methodology consistently outperform all the competing approaches and the basic reason for the good recognition performance, even with only a few training examples (as illustrated in Table 3.13), is that the new regularizer constraint encourages the input signals from the same class to have similar sparse codes and those from different classes to have dissimilar sparse codes thereby maintaining a high classification accuracy even when using a smaller training set.

3.3.3 WISDM

The Wireless Sensor Data Mining (WISDM) dataset consists of time series data belonging to 29 volunteers performing daily activities such as walking, jogging, climbing stairs, sitting, and standing. This aggregated time series data summarized the user activity over 10-second intervals. WISDM project [90] aims to explore the use of accelerometer sensor from powerful mobile devices in order to identify the activity that a user is performing. The time series data was divided into 10-second segments and forty-three features were generated from the accelerometer values contained in each 10-second interval. These forty-three features are variations of just the six basic features(i.e. average acceleration, standard deviation, average resultant acceleration, binned distribution). The resulting training data aims to induce a predictive model for activity recognition, thereby acquiring useful knowledge about the habits of millions of users by having them carry cell phones in their pockets.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	1591	19	57	22	3	0
Class 2	19	1154	13	8	6	0
Class 3	287	28	137	78	0	2
Class 4	271	5	156	47	5	4
Class 5	3	5	4	9	253	2
Class 6	2	2	0	4	22	132

Table 3.14: Overall confusion matrix using proposed class specific centralized dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	94.03%	1.12%	3.36%	1.3%	0.177%	0%
Class 2	1.58%	96.16%	1.08	0.66%	0.5%	0%
Class 3	53.94%	5.26%	25.75%	14.66%	0%	0.375%
Class 4	55.53%	1.02%	31.96%	9.63%	1.02%	0.81%
Class 5	1.08%	1.81%	1.44%	3.26	91.66%	0.72%
Class 6	1.23%	1.23%	0%	2.46%	13.5%	81.48%

Table 3.15: Overall confusion matrix in % using proposed class specific centralized dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes.

Method	Proposed	Neural Networks [91]	SVM [91]	J48 [90]	RBFN [90]
Accuracy	**77.1%**	69.5%	70.17%	72.2%	73%

Table 3.16: Recognition results on WISDM Dataset.

Following the defined experimental settings, we evaluate our approach using tenfold cross-validation as in [90], [91], where one fold is used for testing and the remaining nine folds are used for training. The result is averaged over ten runs. The

Varying Training Data %	20	40	60	80	100
Accuracy in %	75.14%	75.25%	76.48%	77.07%	78.64%

Table 3.17: Performance of the proposed algorithm on the WISDM dataset with varying training data

detailed comparison results are shown in Table 3.16. Table 3.14 shows the confusion matrices using the proposed algorithm. We also compared our approach with varying training data. For both evaluation schemes including varying ptrain and tenfold cross-validation, our results are better than other state-of-the-art approaches and boosting learning approaches. Table 3.14 and table 3.16 demonstrates that the proposed algorithm maintain a high classification accuracy and outperform the other three competing approaches significantly, even when using a smaller size training data.

3.4 Conclusion

In this chapter, we presented a novel dictionary learning framework to evaluate the robustness of activity recognition and intensity estimation of aerobic activities using data from wearable sensors. The main contribution is integrating class specific centralized regularizer term into the objective function for dictionary pair learning. It was shown that this results in a discriminative dictionary learning formulation for recognition. The proposed objective function is efficiently optimized by a combination of alternating direction method of multipliers and $l_1 - l_s$ minimization method. Experimental results show that the classifiers built in this framework achieves impressive classification performance over four activity recognition tasks and outperforms state-of-the-art methods along with being trained activity and subject independent. Both of these are important considerations for developing systems that must be robust, scalable and must perform well in real world settings.

Chapter 4

MAXIMUM CORRENTROPY BASED DICTIONARY LEARNING FRAMEWORK FOR PHYSICAL ACTIVITY RECOGNITION USING WEARABLE SENSORS

4.1 Introduction

In this chapter, we present a dictionary pair learning framework based on the maximum correntropy criterion [92] to solve the wearable sensor-based classification problem. Correntropy has demonstrated to obtain robust inferences in information theory learning (ITL) [93] and effectively handle non-Gaussian noise and large outliers [92]. Inspired by dictionary learning experiments that achieved highly successful recognition rates using a few representative samples on high-dimensional data, we propose a unified dictionary pair learning-based framework based on maximum correntropy for human physical activity monitoring and recognition [1], [94]. To optimize the non-convex correntropy objective function, a new alternate minimization algorithm incorporated with an iteratively reweighted method is developed to facilitate convergence. We validate the effectiveness of our proposed model by adopting it on three recognition problem and an intensity estimation problem, each of which includes a large number of physical activities from the recently released datasets [1]. Experimental results indicate that classifiers built using this correntropy induced dictionary learning based framework provide state-of-the-art performance using simple features, and that this approach gives results competitive with classical systems built upon features with prior knowledge [1].

4.2 Proposed Methodology: Correntropy based Dictionary Learning

4.2.1 Maximum Correntropy Criterion (MCC)

Recognition against outliers and noise is critically challenging, mainly due to the unpredictable nature of the errors (bias) caused by noise and outliers. The concept of correntropy was introduced in ITL [93] to process non-Gaussian noise. Correntropy is directly relevant to Renyis quadratic entropy [92] wherein the Parzen windowing method is employed to estimate the data distribution [1]. Maximization of correntropy criterion cost function (MCC) is defined by maximizing

$$V(X,Y) = \frac{1}{N}\sum_{i=1}^{N} k_\sigma(x_i - y_i) = \frac{1}{N}\sum_{i=1}^{N} k_\sigma(e_i) \tag{4.1}$$

where $X = [x_1, x_2, ..x_N]$ is the desired signal, $Y = [y_1, y_2, ..y_N]$ is the system output, $E = [e_1, e_2, ..e_N]$ is the error signal, each of them being N-dimensional vectors, where N is the training data size and $k_\sigma(x)$ is the Gaussian kernel with bandwidth σ given by :

$$k_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp^{-x^2/2\sigma^2} \tag{4.2}$$

M-estimators are a broad class of estimators, which are obtained as minima of sums of functions of the data [1]. In a general framework of M-estimation, MCC can be defined as

$$\rho(e) = (1 - \exp^{-e^2/2\sigma^2})/\sqrt{2\pi}\sigma \tag{4.3}$$

MCC cost function has proved to satisfy the properties of non-negativity, translation invariant, triangle inequality and symmetry and thus is a well-defined metric [92]. Adoption of MCC to train adaptive systems actually makes the output signal close to the desired signal. By analyzing the contour maps [92], it can be inferred that when the error vector is close to zero, it acts like l_2 distance; when the error gets larger, it is equivalent to l_1 distance; and for cases when the error is large the cost metric levels off and is very insensitive to the large-value of error vector, thereby intuitively explaining the robustness of MCC [1].

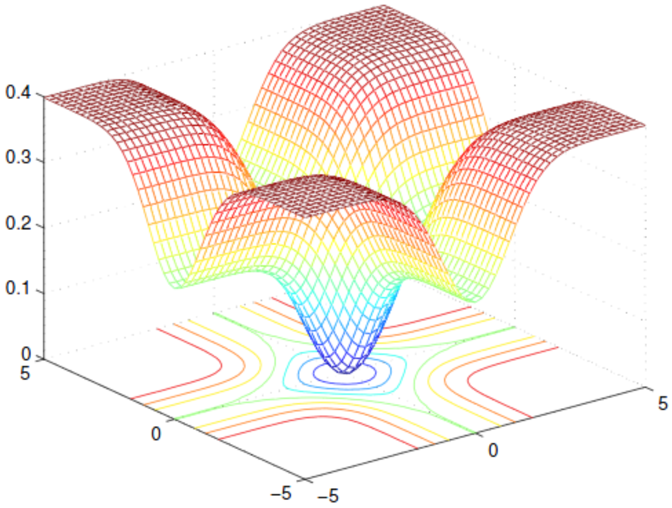

Figure 4.1: Contour Maps of Correntropy Induced Metric

4.2.2 Correntropy based Dictionary Pair Learning Framework

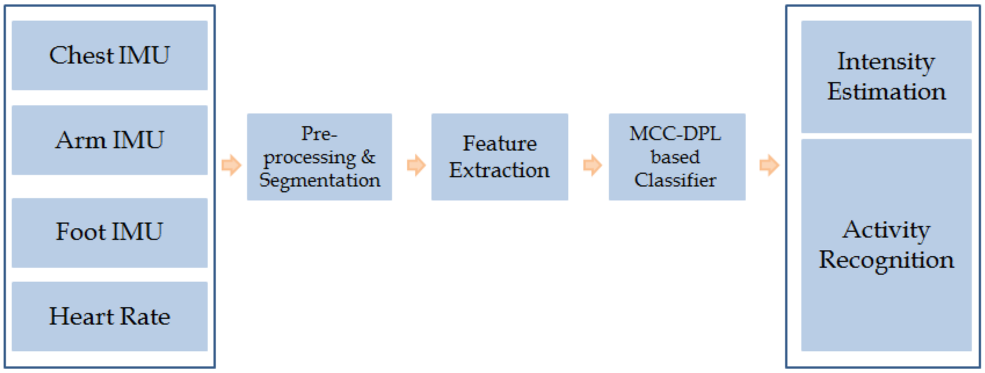

Figure 4.2: Proposed Correntropy based Dictionary Pair Learning Framework

The dictionary pair learning classification algorithm initially jointly learns an analysis dictionary and a synthesis dictionary to achieve the goal of signal representation and discrimination [1]. To define discriminative dictionary learning, we define a set of p-dimensional training samples from K classes by $X = [X_1, ..., X_k, ..., X_K]$, where $X_k \in \mathbb{R}^{p \times n}$ is the training sample set of class k, and n is the number of samples of each class. Discriminative dictionary learning (DL) methods aim to learn an effective data representation model from X for classification tasks by exploiting the class label information of training data, and can be formulated under the following framework:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \lambda||\mathbf{A}||_p + \Psi(\mathbf{D},\mathbf{A},\mathbf{Y}) \tag{4.4}$$

Here $\lambda \geqslant 0$ is a scalar constant; Y represents the class label matrix of samples in X; D is the synthesis dictionary to be learned; and A is the coding coefficient matrix of X over D. In the training model (equation 4.4), the data fidelity term $\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2$ ensures the representation ability of D; $||\mathbf{A}||_p$ is the l_p-norm regularizer on A; and $\Psi(\mathbf{D},\mathbf{A},\mathbf{Y})$ denotes the discrimination promotion function that ensures the discrimination power of D and A [83].

If when using an analysis dictionary denoted by $P \in \mathbb{R}^{mK\times p}$, the code A can be analytically obtained as A = PX, then the representation of X becomes efficient. Based on this idea, the DPL model learns an analysis dictionary P together with the synthesis dictionary D, leading to the following DPL model [83]:

$$P^*, D^* = \arg\min_{\mathbf{P},\mathbf{D}} \|\mathbf{X}-\mathbf{D}\mathbf{P}\mathbf{X}\|_F^2 + \Psi(\mathbf{D},\mathbf{P},\mathbf{X},\mathbf{Y}) \tag{4.5}$$

Here Ψ (D, P, X, Y) is the discrimination function; and D and P form a dictionary pair where the analysis dictionary P is used to analytically code X, and the synthesis dictionary D is used to reconstruct X. The learned structured synthesis dictionary $D = [D_1, ..D_k, ., D_K]$ and the structured analysis dictionary $P = [P_1, ..., P_k......P_K]$ form a sub-dictionary pair corresponding to class k [1]. Thus using the structured analysis dictionary P, we want the sub-dictionary P_k to project the samples from class i,$i \neq k$ to a nearly null space thereby making the coefficient matrix PX nearly block diagonal. Using variable matrix A to relax the non-convex problem, we readily have the following DPL model [1]:

$$P^*, A^*, D^* = \arg\min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k\mathbf{A}_k\|_F^2 + \tau\|\mathbf{P}_k\mathbf{X}_k - \mathbf{A}_k\|_F^2 + \lambda||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2 \tag{4.6}$$

Here $\overline{X_k}$ denotes the complementary data matrix of X_k in the whole training set X, $\lambda > 0$ is a scalar constant, and d_i denotes the ith atom of synthesis dictionary D. Incorporating correntropy based criteria (equation 4.3), equation 4.6 can be rewritten as:

$$P^*, A^*, D^* = \arg\min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=1}^{K} 1 + \tau - \lambda \exp^{\|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2/\sigma^2} - \tau \exp^{\|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2/\sigma^2} - \lambda \exp^{||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2/\sigma^2} \tag{4.7}$$

In order to solve this equation, we adopt alternating direction method of multipliers with iteratively reweighted method to facilitate convergence. A general maximization problem solved by an iteratively reweighted method can be described as follows: Consider a general equivalent maximization problem

$$\max f(x) + \sum_i h_i(g_i(x)) \tag{4.8}$$

where $f(x)$ and $g_i(x)$ are arbitrary functions, x denotes an arbitrary constant and $h_i(x)$ is an arbitrary convex function in domain of $g_i(x)$. The details to solve general maximization problem (equation 4.8) using iteratively reweighted optimization is described in Algorithm 1 [95], where $h_i(g_i(x))$ denotes any supergradient of the concave function h_i at point $g_i(x)$.

ProblemTransformation $h_i(g_i(x)) \rightarrow Tr(D_i^T g_i(x))$

```
Algorithm 1 Optimization algorithm for a general maximization problem
0: Initialize $D_i = I$
1: Update x by optimal solution to the problem
$\max f(x) + \sum_i Tr(D_i^T g_i(x))$
2: Calculate $D_i = h_i'(g_i(x))$ for each i
3: Iteratively perform 1-2 until convergence
```

Now consider maximum correntropy criterion problem:

$$\min f(x) + \sum_i -exp^{-l_i^2(x)/2\sigma^2} \tag{4.9}$$

where $f(x)$ and $l_i(x)$ are arbitrary functions and x indicates an arbitrary constant. Comparing with equation (8), in $h_i(g_i(x))$, let $h_i(z) = \exp^{-z/2\sigma^2}(z > 0)$ and $g_i(x) = l_i^2(x)(z > 0)$, then $h_i(g_i(x)) = \exp^{-l_i^2(x)/2\sigma^2}$ where $h_i(z) = 1 - \exp^{-z/2\sigma^2}$ $(z > 0)$ is concave function. Applying the iteratively reweighted method (Algorithm 1) [95], the problem transformation and steps to determine the maximum correntropy criterion problem can be described as:

$$\textbf{ProblemTransformation}: \ 1 - \exp^{-l_i^2(x)/2\sigma^2} \rightarrow d_i l_i^2(x)(d_i = \tfrac{1}{2\sigma^2} exp^{-l_i^2(x_t)/2\sigma^2})$$

Algorithm 2 Optimization algorithm for maximum correntropy criterion

```
0: Initialize $d_i = 1$
1: Update x by optimal solution to the problem
$\min f(x) + \sum_i d_i l_i^2(x)$
2: Calculate $d_i = \frac{1}{2\sigma^2} exp^{-l_i^2(x_t)/2\sigma^2}$ for each i
3: Iteratively perform 1-2 until convergence
```

The original objective function (equation 4.7) can be easily solved by a combination of ADMM and an iteratively re-weighted algorithm. The alternating direction method of multipliers (ADMM) solves convex optimization problems by fixing some variables and solving for the other variable, thereby breaking the problem into smaller pieces making each of them easier to handle. The minimization can be alternated between the two steps as enumerated:

1: Update A

$$A^* = \arg\min_{\mathbf{A}} \sum_{k=1}^{K} -\lambda \exp^{\|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2/\sigma^2} - \tau \exp^{\|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2/\sigma^2} \tag{4.10}$$

Applying the problem transformation defined in algorithm 2, we have the closed-form solution:

$$A^* = (d_1 D_k^T D_k + d_2 \tau I)^{-1}(d_1 D_k^T X_k + \tau d_2 P_k X_k) \tag{4.11}$$

$$where \ \ d_1 = \frac{1}{2\sigma^2} \exp^{-\|\mathbf{X}_k - \mathbf{D}_k \mathbf{A}_k\|_F^2/2\sigma^2} \ and \ d_2 = \frac{1}{2\sigma^2} \exp^{-\|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2/2\sigma^2} \tag{4.12}$$

2: For updating P

$$P^* = \arg\min_{\mathbf{P}} \sum_{k=1}^{K} -\tau \exp^{\|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2/\sigma^2} -\lambda \exp^{||\mathbf{P}_k, \overline{\mathbf{X}_\mathbf{i}}||_F^2/\sigma^2} \tag{4.13}$$

The closed-form solutions of P can be obtained as:

$$P^* = d_2 \tau A_k X_k^T (d_2 \tau X_k X_k^T + d_3 \lambda \bar{X}_k \bar{X}_k^T + Y I)^{-1} \tag{4.14}$$

where

$$d_2 = \frac{1}{2\sigma^2} \exp^{-\|\mathbf{P}_k \mathbf{X}_k - \mathbf{A}_k\|_F^2/2\sigma^2} \quad d_3 = \frac{1}{2\sigma^2} \exp^{-\|\mathbf{P}_k \bar{\mathbf{X}}_\mathbf{k}\|_F^2/2\sigma^2} \tag{4.15}$$

Iterate between the steps until convergence.

In the testing phase, the analysis sub-dictionary P_k is trained to produce small coefficients for samples from classes other than k, and thus can only generate significant coding coefficients for samples belonging to class k. Meanwhile, the synthesis sub-dictionary D_k is trained to reconstruct the samples of class k from their projective coefficients $P_k X_k$, i.e., residual will be small. Conversely, since $P_k X_i$ will be small and D_k is not trained to reconstruct X_i, the residual a_i will be much larger. Thus, if the query sample y is from class k, its projective coding vector P_k will more likely be large, while its projective coding vectors P_i will be small. Therefore, class-specific reconstruction residual is used to identify the class label of testing samples [1].

4.3 Experimental results and discussion

In this section, we present experimental results to verify the effectiveness of our proposed approach on four recognition tasks. The four recognition tasks: PAMAP2 Activity Recognition, PAMAP2 Intensity Estimation task, Smartphone-Based Human Activities Recognition dataset and Wireless Sensor based Activity Recognition dataset are available at the UCI machine learning repository [96]. For comparative analysis, we implemented the same experimental setup along with the cross-validation scheme as in the referred papers for each dataset [1]. Additionally, we have added classification results for varying percentages (%) of training data and associated computation times.

4.3.1 Results with PAMAP2 Intensity Estimation and Activity Recognition Tasks

Here we consider two classification task provided within the PAMAP2 Physical Activity Monitoring Data Set available at the UCI machine learning repository [96]. Briefly, this database incorporates wide range of everyday, household, and fitness activities involving 18 physical activities performed by 9 subjects wearing 3 IMUs (Inertial measurement unit) and a HR (heart rate) monitor [1]. Each subject adopted the predefined data collection protocol of 12 activities (lie, sit, stand, walk, run, cycle, nordic walk, iron, vacuum, jump rope, ascend and descend stairs), and optionally completed 6 other activities (watch TV, work at a computer, drive a car, fold laundry, clean house, play soccer) to enrich the dataset range [1].

To evaluate the proposed framework, we used both the activity recognition and intensity estimation classification problems defined on the PAMAP2 dataset. In the PAMAP2 dataset, the activity classification task has 15 activities including 3 additional activities from the optional activity list (fold laundry, clean house, play soccer) [1]. The complete activity recognition task consisting of these 15 different activity classes is referred to as the PAMAP2 Activity Recognition (PAMAP2-AR) task. The goal of the intensity-estimation classification task is to distinguish activities of light, moderate, and vigorous effort, and is referred to as the PAMAP2 Intensity Estimation (PAMAP2 - IE) task. These levels of effort are differentiated based on the MET of the various physical activities, as provided by [19]. Therefore, intensity classes are defined as activities of light effort ($<$ 3.0 METs) (lie, sit, stand, drive a car, iron, fold laundry, clean house, watch TV, work at a computer), moderate effort (3.0-6.0 METs) (walk, cycle, descend stairs, vacuum and nordic walk), or vigorous effort ($>$ 6.0 METs) (run, ascend stairs, jump rope, play soccer).

Thus two classification tasks were defined: (1) activity recognition and (2) intensity estimation. Considering these two defined classification tasks, we evaluated different boosting methods and compared them to our proposed correntropy-based dictionary pair learning-based approach [1]. For the evaluation procedure, we randomly selected 75% of the data for training and 25% for testing as in [3]. The final result is the averaged value over all ten runs.

4.3.1.1 Results on PAMAP2 Intensity Estimation Task

In [3], base-level classifiers were used for activity monitoring classification tasks. The C4.5 decision tree algorithm was tested on the PAMAP2 dataset and demonstrated an accuracy of 70.07%. Also, the results obtained in [3] indicated that further improvement in classification accuracy is attainable since even the best result employing Adaboost classifier was only 73.93% accuracy. The overall confusion matrix using the proposed framework for the three intensity estimation tasks (*i.e.*, light (Class 1), moderate (Class 2) and vigorous (Class 3) tasks) is given in Table 4.1. An independent performance assessment of the proposed framework results in an accuracy of 87.6% on the AR-IE task, demonstrating that our framework [1] outperforms the C4.5 decision tree and AdaBoost classifiers (Table 4.3).

Reference/Algorithm	Class 1	Class 2	Class 3
Class 1	12893	422	9
Class 2	1258	6291	13
Class 3	218	1136	760

Table 4.1: Overall confusion matrix using maximum correntropy criterion based dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3
Class 1	96.77%	3.17%	0.06%
Class 2	16.64%	83.19%	0.17%
Class 3	10.31%	53.74%	35.95%

Table 4.2: Overall confusion matrix in % using maximum correntropy criterion based dictionary pair learning framework on PAMAP2-IE dataset. The table shows how different annotated activities are classified into different classes [1].

Methodology	Proposed	C4.5 [3]	Adaboost [3]
Accuracy	**87.59%**	70.07%	73.93%

Table 4.3: Comparison of proposed approach on PAMAP2-IE dataset to state-of-the-art methods in terms of accuracy (calculated in %)

Reference/ Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Class 11	Class 12	Class 13	Class 14	Class 15
Class 1	1660	2	2	19	14	0	0	0	1	0	0	0	28	2	0
Class 2	0	1387	34	126	24	4	0	5	0	2	0	0	68	4	1
Class 3	0	23	1364	144	102	1	2	1	6	0	0	0	57	1	2
Class 4	0	10	55	1779	90	0	0	0	0	0	0	0	236	19	0
Class 5	0	0	13	194	1086	5	0	6	0	2	0	0	243	0	8
Class 6	0	0	0	0	2	474	13	210	69	2	0	0	9	0	0
Class 7	0	0	0	0	0	159	295	56	81	1	0	1	38	0	0
Class 8	0	0	0	0	0	73	16	1824	275	2	0	0	3	0	0
Class 9	0	0	0	0	0	5	6	98	1600	0	0	0	0	0	0
Class 10	0	0	0	3	10	8	9	49	90	1271	0	0	27	0	5
Class 11	0	0	0	0	199	0	0	0	6	0	624	0	0	0	6
Class 12	0	0	0	0	25	0	49	0	38	0	0	253	3	0	0
Class 13	0	9	23	320	478	10	6	1	8	11	0	0	860	16	6
Class 14	0	4	16	439	209	0	0	0	0	0	0	0	145	88	0
Class 15	0	0	0	0	15	50	1	22	52	2	8	3	55	0	187

Table 4.4: Confusion matrix using proposed framework on PAMAP2 -AR dataset.

4.3.1.2 Results on PAMAP2 Activity Recognition Task

To investigate our proposed framework's classification efficiency on PAMAP2−AR task, we determined its performance on 15 classes-based activity-recognition tasks on the PAMAP2 dataset. Table 4.4 enumerates the overall confusion matrix for these tasks. We find that our framework [1] outperforms the C4.5 decision tree with an accuracy of 74.12% versus 71.59%, and it gives competitive results when compared to an AdaBoost classifier on the PAMAP2 AR task (Table 4.6). In addition to competitive accuracy, our framework provides the additional advantages of lower training and testing times [1].

Further examination of our results indicate that averaged over 10 test runs, the confusion matrix of the best-performing classifier on the PAMAP2-AR task yields an overall accuracy of 74.12%, showing that some activities are recognized with high accuracies, such as lying, walking, or even distinguishing between ascending and descending.

Reference/ Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Class 11	Class 12	Class 13	Class 14	Class 15
Class 1	96.06%	11.57%	11.57%	1.1%	0.81%	0%	0%	0%	0.06%	0%	0%	0%	1.62%	0.11%	0%
Class 2	0%	83.81%	2.05%	7.61%	1.45%	0.244%	0%	0.3%	0%	0.13%	0%	0%	4.11%	0.24%	0.06%
Class 3	0%	1.35%	80.09%	8.46%	5.99%	0.06%	0.12%	0.06%	0.35%	0%	0%	0%	3.34%	0.06%	0.12%
Class 4	0%	0.04%	2.51%	81.27%	4.11%	0%	0%	0%	0%	0%	0%	0%	10.78%	0.87%	0%
Class 5	0%	0%	0.83%	12.46%	69.75%	0.32%	0%	0.39%	0%	0.13%	0%	0%	15.61%	0%	0.51%
Class 6	0%	0%	0%	0%	0.25%	60.85%	1.67%	26.96%	8.86%	0.25%	0%	0%	1.16%	0%	0%
Class 7	0%	0%	0%	0%	0%	25.2%	46.75%	8.87%	12.84%	0.16%	0%	0.16%	6.02%	0%	0%
Class 8	0%	0%	0%	0%	0%	3.33%	0.73%	83.17%	12.54%	0.09%	0%	0%	0.14%	0%	0%
Class 9	0%	0%	0%	0%	0%	0.29%	0.35%	5.73%	93.63%	0%	0%	0%	0%	0%	0%
Class 10	0%	0%	0%	0.2%	0.68%	0.54%	0.61%	3.33%	6.11%	86.35%	0%	0%	1.83%	0%	0.35%
Class 11	0%	0%	0%	0%	23.83%	0%	0%	0%	0.72 %	0%	74.73%	0%	0%	0%	0.72%
Class 12	0%	0%	0%	0%	6.79%	0%	13.32%	0%	10.33%	0%	0%	68.75%	0.81%	0%	0%
Class 13	0%	0.51%	1.32%	18.31%	27.35%	0.57%	0.34%	0.06%	0.45%	0.63%	0%	0%	49.2%	0.92%	0.34%
Class 14	0%	0.44%	1.78%	48.72%	23.2%	0%	0%	0%	0%	0%	0%	0%	16.09%	9.77%	0%
Class 15	0%	0%	0%	0%	3.8%	12.66%	0.25%	5.57%	13.16%	0.51%	2.02%	0.76%	13.92%	0%	47.35 %

Table 4.5: Confusion matrix in % using proposed framework on PAMAP2 -AR dataset.

Methodology	Proposed approach	C4.5 [3]	Adaboost [3]
Accuracy	**74.12**%	71.59%	71.78%

Table 4.6: Comparison of proposed approach on PAMAP2-AR dataset to state-of-the-art methods in terms of accuracy (calculated in %)

Overall, misclassifications, where activities belonging to one class are mistakenly classified as belonging to its neighboring classes, are lower. One example of overlapping activity characteristics is the over 5% confusion between nordic walk (class 7) and cycle (class 6) and ascend (class 9) that is function of the positioning of the sensors; thus an IMU on the thigh would reliably help differentiate these postures.

Another example comes from the playing soccer activity, because playing soccer (class 14) is a composite activity. Thus it becomes problematic to distinguish running with a ball from just running (class 4). Arguably, however, the main reason for these misclassifications is the diversity inherent in the subject's performance of physical activities. Therefore, further increasing the accuracy of physical activity recognition will require the introduction and investigation of personalized approaches.

In [3], the evaluation technique used was leave-one-activity-out (LOAO) where an activity monitoring system is used on a previously unknown activity. Our framework [1] takes in data randomly. Thus, our evaluation is tested on an entirely unknown activity from an unknown user, while training is performed on a different activity with a different user in our random 75%-25% validation approach. Such kind of subject-independent and activity-independent validation techniques are favorable for physical activity monitoring since they yield results with more practical applications. Using our framework [1], we can not only achieve good classifier performance but also exclude the need of pre-training a particular activity for classification. Thus, our proposed method [1] makes it possible to design a robust physical activity monitoring system having desired generalization characteristics.

4.3.2 SBHAR

We also conduct experiments on SBHAR activity recognition dataset [86] to verify the effectiveness of our proposed method. This dataset includes six activities i.e., walking, walking upstairs, walking downstairs, sitting, standing and laying. The activities in the SBHAR dataset are performed using a smartphone (Samsung Galaxy S II) mounted on the waist. A 3-axial linear acceleration and 3-axial angular velocity are captured using the phone's embedded accelerometer and gyroscope at a constant rate of 50Hz. The pre-processing is done by application of noise filters, followed by sampling in fixed-width sliding windows of 2.56 sec and 50% overlap (128 readings/window). Time and frequency domain (e.g. mean, standard deviation, signal magnitude area, entropy, signal-pair correlation, etc.) feature vectors were obtained from each window.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	1871	35	0	0	0	0
Class 2	372	1390	15	0	0	0
Class 3	0	0	1944	0	0	5
Class 4	0	0	0	1680	20	22
Class 5	0	0	0	0	1378	28
Class 6	0	0	0	0	6	1538

Table 4.7: Overall confusion matrix using correntropy based dictionary pair learning framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	98.16%	1.84%	0%	0%	0%	0%
Class 2	20.93%	78.22%	0.85%	0%	0%	0%
Class 3	0%	0%	99.74%	0%	0%	0.26%
Class 4	0%	0%	0%	97.56%	1.16%	1.28%
Class 5	0%	0%	0%	0%	98.01%	1.99%
Class 6	0%	0%	0%	0%	0.39%	99.61%

Table 4.8: Overall confusion matrix in % using correntropy based dictionary pair learning framework on SBHAR dataset. The table shows how different annotated activities are classified into different classes.

Method	Proposed	Multi-Class SVM [88]	Multi-Class HF SVM [88]	Naives Bayes [89]	Decision Tree [89]
Accuracy	**91.51%**	89.0%	89.3%	82.5%	86.8%

Table 4.9: Recognition results on SBHAR Dataset.

Varying Training Data %	20	40	60	80	100
Accuracy in %	91.42%	92.1%	92.49%	92.68%	93.11%

Table 4.10: Performance of the proposed algorithm on the SBHAR dataset with varying training data

Following the defined experimental settings as in [88], [87], we evaluated our proposed approach and compared with Naive's Bayes Classifier, Decision Tree, Multiclass HF SVM and other state-of-the-art approaches used in [88]. Table 4.7 and table 4.9 represent the confusion matrix and recognition rates using the proposed approach. Table 4.9 demonstrates that our approaches outperforms all the competing state-of-the-art approaches in [88] and [87].

4.3.3 WISDM

The Wireless Sensor Data Mining (WISDM) project [90] was designed to analyze the usefulness of accelerometer sensor for user activity recognition using mobile devices platform. The WISDM dataset includes daily activities like walking, jogging, climbing stairs, sitting, and standing performed by 29 volunteers. This aggregated time series data summarizes the user activity over 10-second intervals and a set of forty-three features were calculated from the accelerometer data contained in each 10-second interval. These forty-three features are an adaptation of the six basic features(e.g. average acceleration, standard deviation, average resultant acceleration, binned distribution). The goal is to leverage the resulting training data to develop a predictive model for activity recognition, thereby acquiring useful knowledge about the habits of millions of users without intruding their day-to-day activities.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	1580	12	54	29	3	0
Class 2	17	1297	17	10	0	0
Class 3	270	34	147	71	1	5
Class 4	262	8	141	61	7	0
Class 5	0	0	8	6	271	5
Class 6	2	0	4	6	21	159

Table 4.11: Overall confusion matrix using correntropy based dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes.

Reference/Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	94.16%	0.71%	3.22%	1.73%	0.18%	0%
Class 2	1.27%	96.72%	1.27%	0.74%	0%	0%
Class 3	51.14%	6.44%	27.84%	13.45%	0.19%	0.94%
Class 4	54.7%	1.67%	29.44%	12.73%	1.46%	0%
Class 5	0%	0%	2.76%	2.07%	93.45%	1.72%
Class 6	1.04%	0%	2.08%	3.12%	10.95%	82.81%

Table 4.12: Overall confusion matrix in % using correntropy based dictionary pair learning framework on WISDM dataset. The table shows how different annotated activities are classified into different classes.

Method	Proposed	Neural Networks [91]	SVM [91]	J48 [90]	RBFN [90]
Accuracy	**78.40**%	69.5%	70.17%	72.2%	73%

Table 4.13: Recognition results on WISDM Dataset.

We assess our approach [1] using tenfold cross-validation similar to [90], [91], wherein one fold is used for testing the data and the remaining nine folds are used

Varying Training Data %	20	40	60	80	100
Accuracy in %	76.29%	76.81%	77.52%	78.83%	79.91%

Table 4.14: Performance of the proposed algorithm on the WISDM dataset with varying training data

for training data. Recognition results are averaged over ten runs and are presented in Table 4.13. Table 4.11 represents the confusion matrices using the proposed algorithm on WISDM dataset. We also compared our approach [1] for varying training data case. For both the evaluation schemes i.e. varying training data case and ten-fold cross-validation case, our algorithm results prove superior when compared to other state-of-the-art approaches. Table 4.13 and table 4.14 indicates that the proposed algorithm provides a high classification accuracy and outperform the other three competing approaches significantly, even using a smaller size training data.

4.4 Conclusion

In this chapter, we proposed an effective dictionary pair learning-based framework based on the maximum correntropy criterion to evaluate the robustness of activity recognition and intensity estimation of aerobic activities using data from wearable sensors [1]. The proposed objective function is robust to outliers and can be efficiently optimized by the combination of iteratively reweighted technique and alternating direction method of multipliers [1]. Experimental results illustrate that classifiers built in this framework not only provide competitive performance but also demonstrate subject-independent activity classification using accelerometers. In pertinence to developing systems, both of these considerations are vital since they are a deciding factor for robustness, scalability and promising real time performance. Having foundational promising and reliable results, out goal is to further extend our work by incorporating tree structure and smooth constraint to the classification framework [1].

Chapter 5

A HIERARCHICAL DICTIONARY LEARNING FRAMEWORK FOR PHYSICAL ACTIVITY RECOGNITION USING WEARABLE SENSORS

Activity recognition based on wearable sensor technology is one of the most active research areas in computer vision due to promising applications in fields such as healthcare, smart homes, human sports performance analysis, and virtual reality simulations. Although classification accuracy is enhanced using state-of-the-art classifiers, actual recognition performance tends to fall off when classifying a large number of complex activities. To address this problem, we propose the concept of a cluster-based classifier hierarchy, each cluster serving to distinguish between child nodes at any given location in the hierarchy [97], [98]. By injecting a tree-based clustering model into a human activity recognition process, both a semantic- attribute representation and a multi-layer classifier are achieved. To model the sequential structure, a CFu-tree-based graphical model [5] is combined with feature and spectral mean similarity metric to take into account any dependency or similarity of activity attributes. Finally, a class-specific dictionary pair algorithm is employed at each macro-class level to both strengthen and reinforce activity recognition accuracy using minimal user feedback.

The proposed model is validated using two real-world daily life activities datasets. Evaluation results indicate that the CFu-tree-based incremental clustering algorithm [5] when incorporated with a dictionary pair learning model can outperform other hierarchical models as well as non-hierarchical baseline algorithms having limited labeled data for new classes, while maintaining the high accuracy of a learned model. Experimental results show that the proposed hierarchical framework can effectively classify activities with significant accuracy improvement, thereby advancing the state of the art of human activity recognition.

5.1 Introduction

Recent years have witnessed a rapid proliferation of wearable sensor technologies, a result of the many potential application domains such as healthcare, sports, fitness, entertainment, humancomputer interface (HCI), security, and commerce. Smart wearable sensors technology has revolutionized our lives, social interaction, and activities as they prove useful in providing accurate and reliable information about an individuals activities and behaviors, thereby promoting a safe and sound living environment.

In an application for elderly people in assisted living, any physical activity monitoring must be able to estimate intensity as well as distinguish between complex and similar activities like standing, walking, running, cycling, and other household activities. Wearable sensors in the form of panic buttons should send alerts for emergencies in case any unusual activity is detected, such as a person falling. Few datasets are specifically benchmarked for large complex tasks within physical activity monitoring. The PAMAP2 dataset is one of few datasets that involve large numbers of users and activities. Although many varied machine learning models have been defined for activity recognition, a robust classifier must be able to obtain high accuracy even in large, complex, sensor-confusing classes while also performing well with limited training data. This would allow pervasive lifestyle monitoring of more complex scenarios like exercise patterns and cooking habits, all of which have been shown to be recognizable using on-body IMUs in [99], [100], and [101], respectively.

Because the activity recognition dataset in the PAMAP2 dataset consists of many overlapping and sensor-confusing activities, standard classifiers have not been able to achieve optimal recognition rates. This difficulty can be addressed by employing the proposed hierarchical dictionary learning framework wherein an incremental conceptual clustering approach is leveraged to design the hierarchy of classifiers for grouping nodes at particular levels in the hierarchy. A CFU-tree-based clustering method [5] is presented for finding and characterizing each cluster. Our method, an agglomerative hierarchical clustering method, uses a bottom-up strategy. That is, it starts by letting each object be a cluster and then iteratively merges these clusters into larger clusters. For the merging step, the two clusters closest to each other are found (according to the similarity measure) and then combined to form a single cluster.

We examine the proposed hierarchical framework of the Class Specific Centralized Dictionary Pair Learning algorithm (i.e., CSCDPL) [82] and compare it with hierarchical models of other algorithm as well as on non-hierarchical models. Traditional flat models do not perform well when distinguishing between similar classes as they tend to cluster together in the feature space. This can be overcome by breaking the single CSCDPL classifier [82] into a hierarchical set of simpler CSCDPL classifiers, in which the CFU- tree-based clustering mechanism is used to distinguish within the mutually exclusive macro-classes defined by the hierarchy. For each dataset, we compare the total classification accuracy of the hierarchical model to the non-hierarchical model baseline, demonstrating significant improvements in both data representation and model design by means of a hierarchical classifier.

The rest of the chapter is organized as follows. Section 5.2 describes the relevant work on wearable sensor-based activity recognition and hierarchical classification. Section 5.3 introduces the clustering process using CFu-tree and presents our proposed framework of the classifier based on incremental clustering, feature-mean and spectral mean based similarity metric and CSCDPL algorithm. In Section 5.4 several comprehensive experiments are performed and results provided in order to evaluate the effectiveness of our method. Section 5.5 draws the conclusions from our work.

5.2 Related Work

The goal of activity recognition is to identify activities as they occur based on data collected by sensors. Approaches to activity recognition [102] vary according to the underlying sensor technologies used to monitor activities, the machine learning algorithms used to model those activitiesmodels.and the realism of the testing environment. Advances in pervasive computing and sensor networks have resulted in the development of a wide variety of sensor modalities useful for gathering information about human activities. As may be observed in most current studies, sensor data collected for activity recognition are usually analyzed using machine learning tools.

In activity recognition research, most studies have focused on flat classification in which predefined categories treated in isolation lack structure or layers that define the relationships among them [103], [104]. Such categories are also known as flat categories. Consequently, as the number of classes grows larger, the problem of defining relationships is exacerbated. One way to solve this problem is to organize the categories into hierarchies similar to the way cases used in web classification are handled by Yahoo. Hierarchical classification allows addressing a large classification problem using a divide-and-conquer approach. A few hierarchical classification methods have been recently proposed [104], [105], [106], [107], [108], [109]. Wang et al. [110] have proposed a framework that maps low-level patterns to high-level activities using a hierarchical framework. Van Kasteren et al. [111] employ a hierarchical hidden Markov model to model motion sensor data. In most hierarchical classification methods, the classes are organized in tree-like structures. A more relevant work in macro-class selection for hierarchical classification is given in [110], in which an algorithm based on a weighted support vector uses clustering to select macro-classes in forming a hierarchical classifier for multi-class classification that provides a binary classifier at each node.

The clustering of activities into a hierarchical activity taxonomy can facilitate analysis of activity patterns from multiple sources. In addition, we find that such a taxonomy can be used to scale activity recognition. Instead of using a single classifier to distinguish between large numbers of activities, we plan to design a hierarchy of classifiers, each of which distinguishes between child nodes at a particular location in the hierarchy. CFu-tree-based conceptual clustering provides a set of macro-classes that can be used to build a hierarchical classifier to improve the overall accuracy of a model incorporating a large number of confusing classes. We describe our proposed technique, evaluate classifiers based on clustering algorithms, and demonstrate in subsequent sections how the method improves model accuracy.

In sum, a wearable sensor technology for an activity recognition system is proposed in which an incremental CFu-tree-based conceptual clustering approach learns the cluster for classifications. Further, we analyze how a multi-layer structure scales activity recognition to a large number of complicated activity classes and training datasets. Experimental results show that the proposed hierarchical CSCDPL model can effectively recognize similar activities with higher accuracy. It also outperforms single-layered (flat) supervised learning algorithms as well as other multi-layer classifiers. The results advance state-of-the-art in human activity recognition, and represent a promising step towards bridging the gap between computers and humans, thereby illustrating the feasibility of the proposed method.

5.3 Proposed Methodology

Here we present the proposed CFu-tree-based hierarchical CSCDL classification algorithm which aims to achieve higher scalability along with reduced misclassifications as compared with a single-layer classification framework. The proposed hierarchical framework is a bottom-up incremental conceptual clustering approach and it starts with each input as a separate cluster, that is, an input item is defined a single cluster. The clustering process proceeds by joining two or several existing clusters to form a new one, according to the similarity measure between two clusters. The clustering framework is based on a CFu-tree that represents a cluster hierarchy, which makes it effective for incremental and dynamic clustering of incoming objects.

We first introduce the concept of incremental clustering and CFu-tree-based clustering in section 5.3.1, presenting details of the similarity measure and the clustering process using CFu-tree in section 5.3.2, and the CSCDPL classification approach in subsection 5.3.3.

5.3.1 CFU Tree based Clustering

Clustering itself can be defined as the process of partitioning a set of data into subclasses, in which each subclass is a cluster whose elements are similar in one or more aspects, and dissimilar to elements in other clusters [112], [113]. Unlike classification algorithms, clustering algorithms are based on unsupervised learning, meaning that they do not require any training data. In this paper, we employ an agglomerative multi-layer clustering method that uses a down-top strategy for finding and characterizing each cluster. The procedure is based on the concept of an incremental conceptual clustering system that seeks to maximize inference abilities. In particular, it starts by letting each feature vector be a cluster, and iteratively merges clusters into larger clusters based on a similarity metric. For the merging step, the system finds the two clusters that are closest to each other, and combines these to form one cluster. We describe the CFu-tree-based clustering mechanism [5] in detail in this section.

A CFu-tree structure, representing the process of hierarchical clustering, works by grouping data objects into a tree of clusters [114]. A CFu-tree thus symbolizes a cluster, even if only a single node. A leaf node symbolizes an activity which is a subcluster. A non-leaf node which represents cluster Ci, in the CFu-tree, has descendants that merge into Ci. By storing sums of the cluster features and documents of their

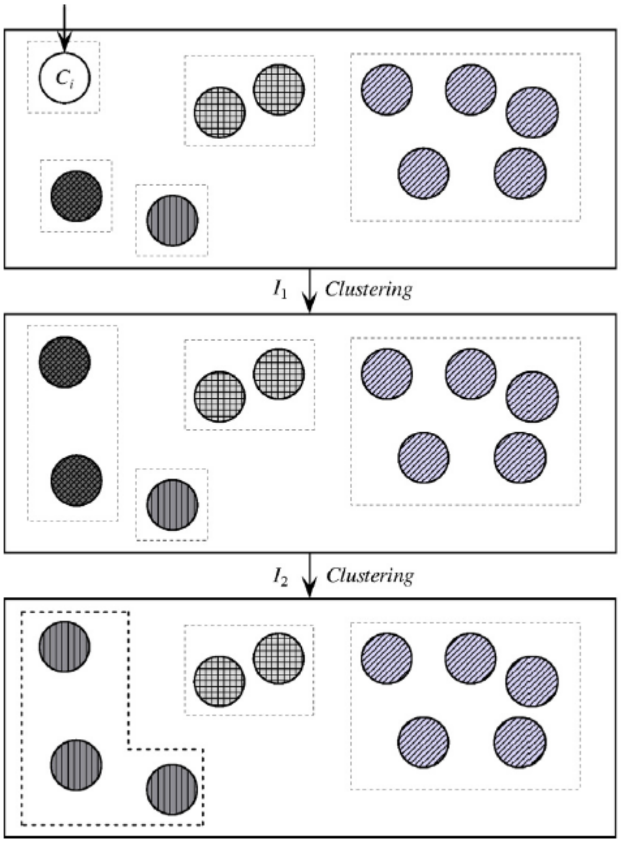

Figure 5.1: Example of clustering where C_i is the new input object [5].

child nodes, the non-leaf nodes thus summarize clustering information about their child nodes. Initially, our method places each activity into a cluster of its own (i.e., each input is treated as a separate cluster). Once a new cluster joins Ci, for each resulting pair of clusters (Ci, Cj), a similarity value (Sim(Xi,Xj)), which determines whether the pair of clusters has to been merged, will be computed [5]. After the computation, the pairwise clusters are merged (Fig. 5.1) according to the similarity criterion. The iterative cluster computing process repeats until the clusters no longer change.

The major steps of clustering by a CFu-tree include the following:

- Step 1. An activity is viewed as a single cluster Ci. After processing and feature extraction, a CFu-tree is built comprising only one node, that is C_i.

- Step 2. Add the CFu-tree to the CFu-tree list, and perform clustering for each pair of clusters.

  Step 3. Scan the CFu-tree list to calculate the similarity.

  Step 4. Merge or split clusters according to similarity criteria.

  Step 5. Repeat Step 3 until no cluster meets the conditions.

5.3.2 Creating the hierarchy using similarity metric

Learning an activity model can be time-consuming, even more so when the number of activity classes is large. Insights can be gained about activities and models can be learned effectively, by organizing activities into a cluster hierarchy. In the CFu hierarchy, each leaf node represents a single activity, and internal nodes represent unions of the activities that reside in the subtree rooted at the node. At each step of the process, the two most similar nodes are merged. When nodes are merged a parent node is created in the hierarchy, which represents a union of the two activities. The original (merged) activities become the children of the new node. Because merging occurs between two nodes at a time, the resulting hierarchy is represented as a binary tree. Of central importance to the problem of clustering, however, is the notion of finding similarity between two nodes. The similarity metric approach we adopted is to estimate similarity as the inverse mean distance between elements of each cluster. This method (also referred to as average linkage clustering) merges clusters with the smallest average distance between all pairs of their feature elements, as denoted by the following equation:

$$d_{avg} = \frac{1}{|X_1||X_2|} \sum_{x_1 \in X_1} \sum_{x_2 \in X_2} d(x_1, x_2) \tag{5.1}$$

To define a distance measure between the data points, we employ a weighted combination method, i.e., feature mean and spectral mean. The feature mean calculates d(x_1, x_2) as the Euclidean distance [115] between the feature vectors for data points x_1 and x_2. The spectral mean [116] performs spectral decomposition on the feature distances. Spectral clustering makes use of the spectrum of the feature distance matrix to reduce the dimensionality of the space and thus perform cluster merging in fewer dimensions. Let $c_1, .., c_L$ represent the set of nodes (activities). We first compute the pairwise distance matrix $D_{avg} = davg_{mn}$, where $davg_{mn}$is the distance between activities c_m and c_n (as computed by eq. 5.1). This distance matrix is then transformed into an adjacency matrix, W, by applying the Gaussian / heat kernel, where $w_m = \exp(-d^2/2\sigma^2)$and σ is the free parameter representing the kernel width. The normalized Laplacian, L, for this adjacency matrix is computed, in which I is the identity matrix and D is the degree matrix (Eq. 5.2).

$$L = 1 - D^{-1/2} W D^{-1/2} \tag{5.2}$$

In equation 5.2 , D is a diagonal matrix in which the diagonal elements contain the sum of all elements in the corresponding row of W. The eigenvectors of L up to K dimensions (where K corresponds to the index with maximum eigengap) represent the activity data points in the transformed space. Finally, cluster distances are computed based on the Euclidean distance between data points in the transformed space . For high-dimensionality data, spectral clustering should generate clusters that reflect the distribution of the data without being sensitive to redundancies in the feature vector description. At each level of the multi-level classifier, we adopt CSCDL for macro-class classification, as explained in the following section.

5.3.3 Class Specific Centralized Dictionary Pair Learning Classifier

The aim of our project is to recognize the human activity from the features. First, the features are clustered into macro classes using a CFu-tree and the similarity metric process defined in sections 3.1 and 3.2. Once the clusters and macro-classes are defined, each hierarchical framework level employs the CSCDPL algorithm for classifying different classes within that level.

The CSDPL algorithm jointly learns a synthesis dictionary and an analysis dictionary in order to simultaneously perform signal representation and classification once the time-domain features have been extracted. An additional term in CSCDPL, i.e., a class-specific regularizer term, ensures that the sparse codes belonging to the same class will be concentrated, thereby proving beneficial for the classification stage. In original DPL model, training samples of each class contribute equivalently to the dictionary, thus generating a dictionary consisting of training samples in corresponding class, resulting in instability and higher residual error. To address this, our main contribution here is to explicitly incorporate centralized class-specific sparse codes [84] to the dictionary pair learning objective function, thereby concentrating sparse codes in the same class. To attain this objective, we denote the mean of each row of sparse code A as E(A). The regularizer term can be formulated as

$$R(A_k) = \eta \sum_{n=1}^{N} \|(\mathbf{A}_k).n - E(\mathbf{A}_k)\|_2^2 \tag{5.3}$$

where η is the tradeoff parameter between the reconstruction error and the degree of deviation from the sparse codes to their centers, and $A.n$ represents the nth column of a sparse matrix A. Incorporating the class specific regularizer term to the DPL objective function, equation (3) can be formulated as follows:

$$\begin{aligned} P^*, A^*, D^* = \arg \min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{D}_k\mathbf{A}_k\|_F^2 + \tau\|\mathbf{P}_k\mathbf{X}_k - \mathbf{A}_k\|_F^2 \\ +\lambda||\mathbf{P}_k, \overline{\mathbf{X_i}}||_F^2 + \eta \sum_{n=1}^{N} \|(\mathbf{A}_k).n - E(\mathbf{A}_k)\|_2^2 \end{aligned} \tag{5.4}$$

The combination of an alternating-direction method of multipliers and a $l_1 - l_s$ minimization method is employed to approximately minimize the objective function. The optimization algorithm is presented in chapter 3 and the effectiveness of the CSCDPL model was validated by employing it on an activity recognition problem and an intensity estimation problem, both of which include a large number of physical activities from the recently released PAMAP2 dataset. Single layer/flat classifiers built in this centralized class specific dictionary learning based framework outperformed state-of-the-art algorithms.

5.4 Evaluation Techniques for the Proposed Framework with experimental results and discussions

5.4.1 Database

We evaluate our approach by using two realistic sensor-based databases, the PAMAP2 -AR dataset and the Wireless Sensor Data Mining Lab (WISDM) dataset, both of which employ a high number of volunteers performing numerous physical tasks. Activities performed in each dataset are similar. PAMAP2-AR active activities include running, playing soccer, walking and walking up and down; static activities include lying and standing. The WISDM dataset includes activities such as walking and ascending and descending stairs. State-of-the-art flat classification techniques make it difficult to distinguish and thus classify such closely similar activities. We, therefore, propose applying our framework to these two complex activity datasets to assess performance.

5.5 Evaluation Metrics

By varying choice of distance measure and merging criteria, a variety of cluster hierarchies can be generated. In order to compare these, a number of cluster quality measures introduced in the literature need to be employed here as well. Internal measures only evaluate cluster quality based on the clustered data itself. (Examples of internal evaluation including measuring compactness within a cluster vs. the separation between clusters are found in [117], [118]; measuring pairwise similarity within a cluster weighted by the cluster size is found in [119]). Further, while other methods of measure such as that of centrality and weakest link are useful when clustering graphs [120], the Davies-Bouldin index [115] and cophenetic distance measure [121], both used here, can be applied specifically to evaluate hierarchical clusters, as external measures relate the quality of clusters to external factors such as classification accuracy [122], [123].

We employ the clustering validity index, which is based on cluster compactness (in terms of intra-cluster variance), and density between clusters (in terms of inter-cluster density). Initially, we compare alternative hierarchies using internal evaluation methods. Subsequently, we perform external evaluation by determining effectiveness of the hierarchy in performing activity classification.

The clustering validity index (CVI) [118] measures compactness of a data set by means of cluster variance, and separation is measured by the density between clusters. Consequently, smaller ratios illustrating compact clusters are better. The cophenetic correlation coefficient (CCC) [121] measures how well cluster hierarchy maintains pairwise distances between original data points (i.e., individual activities). Specifically, the coefficient c measures how closely the original distances between data points (activities) i and j correlate with their distance in the hierarchy t(i,j), or the height of the node at which points i and j are merged (Eq. 5.5).

$$CCC = \frac{\sum_{i<j}(d_{avg}(i,j) - \overline{d}_{avg})(t(i,j) - \overline{t})}{\sqrt{\sum_{i<j}[(d_{avg}(i,j) - \overline{d}_{avg})^2][(\sum_{i<j} t(i,j) - \overline{t})^2]}} \tag{5.5}$$

The closer the value of c is to 1, the more accurately a cluster hierarchy reflects the similarity of the actual data points (activities). To compare cluster hierarchy quality using internal evaluation, we compute the Davies-Bouldin index and the cophenetic correlation coefficient for each hierarchy. The Davies-Bouldin index (DBI) is the ratio of the sum of the within-cluster distance to between-cluster separations. As noted, smaller ratios are better, because they indicate that the clusters are compact and far apart. The index is calculated by an equation, in which n is the number of clusters, S_n the average distance of cluster points to the cluster centroid, and $S(C_i, C_j)$ the distance between cluster centroids (Eq.5.6).

$$DBI = \frac{1}{n} \sum_{i=1}^{n} \max_{i \neq j} \frac{S_n(C_i) + S_n(C_j)}{S(C_i, C_j)} \tag{5.6}$$

5.6 Experimental Results and Discussion

5.6.1 Results from PAMAP2 dataset

To investigate the proposed frameworks performance on PAMAP2− AR task, we computed its performance on 15 classes derived from activity-recognition events. Here the classification task consists of distinguishing and hierarchically arranging 15 different activity classes represented as lie, sit, stand, walk, run, cycle, Nordic walk, drive car, ascend, descend stairs, vacuum, iron, fold laundry, clean house, play soccer, and jump rope. We also evaluate the performance of the flat classifier models that do not use clusters (i.e., wherein the classifier discriminates between all activity classes at once) and other classifiers using hierarchical models (based on a Cfu-tree hierarchy). In reporting recognition accuracy results for the classifiers, the non-hierarchical models for these classifiers are denoted as NH-C4.5, NH-Adaboost, and NH- CSCDPL, while the hierarchical models are denoted as H-C4.5, H-Adaboost, and H-CSCDPL for the CFu-tree based hierarchical models. The human intuition and feature correlation model are denoted as HI-H-CSCDPL and FC-H-CSCDPL model.

A question that naturally arises is how our automatically generated activity hierarchy compares with other possible hierarchies for activity recognition purposes. No single hierarchy is obviously intuitive or able to provide a ground-truth hierarchy for comparison. Annotators for the various datasets do not provide definitions for their interpretations of labeled activities, and yet there are striking interpretational differences of the activities between datasets. These variances are consistent with the work [124], in which humans showed divergent differences in their determination of the similarities of activities. Some participants ranked activity similarities based on function, while others used spatial relationships, temporal relationships, or other criteria in determining similarity. In addressing this disparity of intuition vs. function, we compare the results of our activity hierarchy with two other hierarchies: one based on human intuition of activity similarity and the other utilizing the correlation among features to define activity classes within a hierarchy (Fig. 5.2 and Fig 5.4).

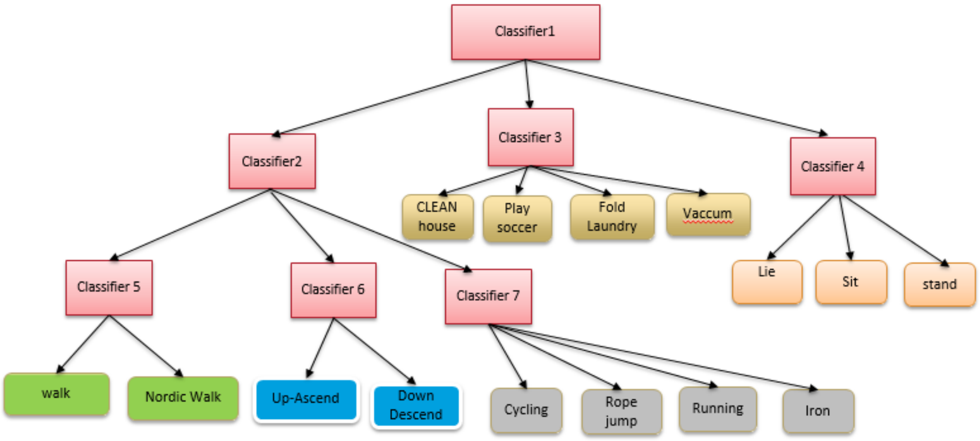

Figure 5.2: Structure of the human intuition based hierarchical classification framework

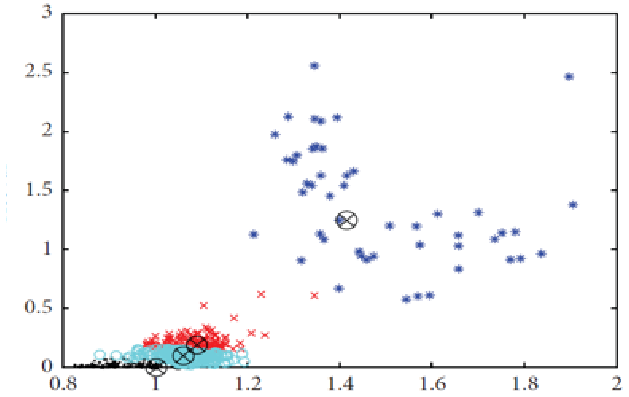

Figure 5.3: Dimensionality Reduction Results for determining feature Correlation.

The hierarchy using human intuition is represented in fig 5.2 wherein similarity of human activities is used to separate classes. The problem of recognizing fifteen activity classes was broken down into seven distinct classification problems. In the

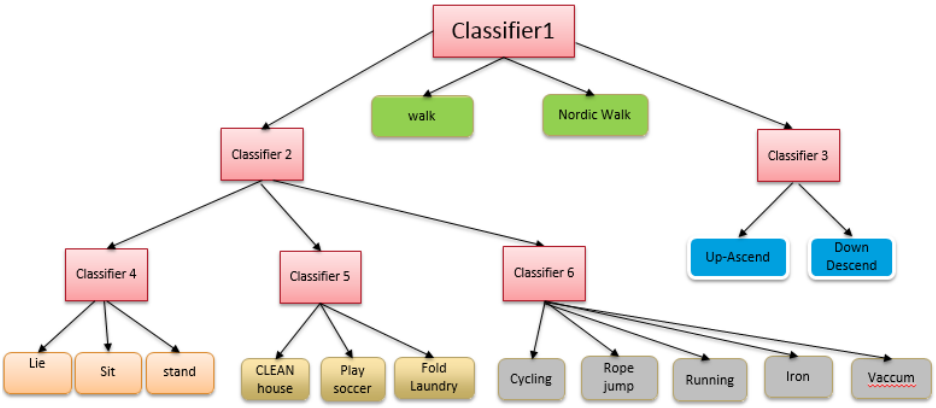

Figure 5.4: Structure of the feature correlation based hierarchical classification framework

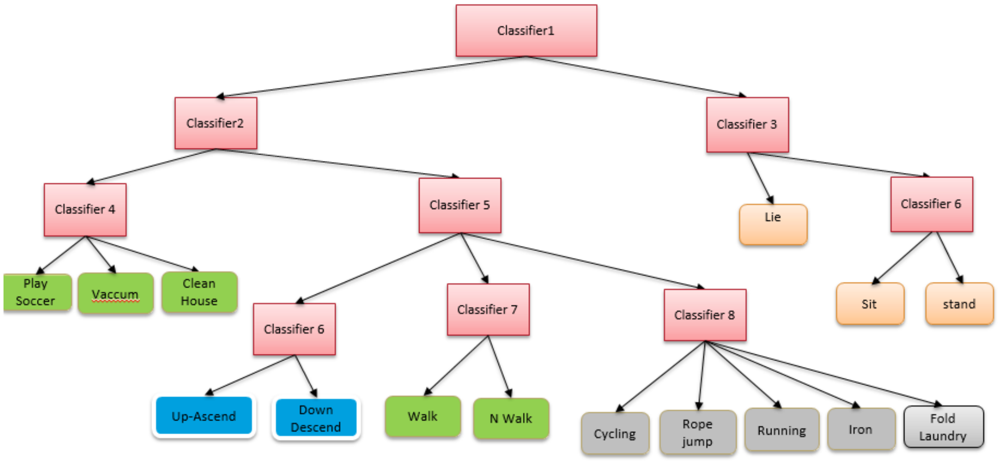

Figure 5.5: Structure of CFu-tree based hierarchical classification framework on PAMAP2 dataset

first layer, the walking-related activities, static activities and composite activities were differentiated. So in the first layer, classifier 2 corresponds to all walking-related activities, classifier 3 differentiates all composite activities and classifier 4 differentiates all static activities. The hierarchy using correlation among the features is represented in fig 5.4 and fig 5.3 helped to determine on how to separate classes based on their similarity. Within the first layer of the feature-based hierarchy, all walking-related activities (2D plane motion walking (walking, nordic walk) and 3D motion (ascend and descend stairs)) were separated from the composite activities, and the static activities. Within the second layer, classifier 3 separated the 3d motion activities (ascend and descend stairs). In the third layer, the static activity subset was differentiated as standing, sitting, and lying and all detailed activities of composite and multiple activities were also recognized by classifier 5 and classifier 6.

Methodology	NH-C4.5 [3]	NH-Adaboost [3]	NH-CSCDPL
Accuracy	71.59%	71.78%	**74.12**%

Table 5.1: Comparison of recognition accuracy of all non-hierarchical (NH- prefix) models (calculated in %)

Methodology	H-C4.5	H-Adaboost	HI-H-CSCDPL	FC-H-CSCDPL	H-CSCDPL
Accuracy	72.21%	75.45%	**79.61**%	**80.93**%	**82.244**%

Table 5.2: Comparison of recognition accuracy of all hierarchical(H- prefix) models(calculated in %)

The cluster hierarchy that generated by the weighted combination method approach on PAMAP2-AR dataset is shown in Figure 5.5. This hierarchy represents a fairly balanced tree. The resulting activity recognition accuracy using a human-generated hierarchy is 79.61%, while accuracy for the feature correlation-based hierarchy is 80.93%, and for our CFu-tree- generated hierarchy 82.244%, showing that the hierarchical model of CSCDPL not only outperforms other non-hierarchical models but other hierarchical models as well (Table 5.1 and Table 5.2). Such positive results indicate that well-formed hierarchies do impact the performance of activity recognition algorithms.

Regarding the internal evaluation scores for the hierarchical models, the hierarchy generated by using CFu-tree yielded a lower Davies-Bouldin index, a lower clustering validity index, and a higher cophenetic correlation coefficient than the other hierarchical model (table 5.3). Furthermore, the cophenetic correlation coefficient for

Methodology	Proposed CFu-CSCDPL	HI-H-CSCDPL	FC-H-CSCDPL
DB	**5.8**	8.34	8.85
CCC	**0.91**	0.78	0.64
CVI	**0.04**	0.11	0.14

Table 5.3: Comparison of Internal evaluation scores of all hierarchical approaches

the CFu-tree hierarchy is close to 1, indicating that the generated hierarchy accurately reflects the underlying data.

5.6.2 Results from the WISDM dataset

The WISDM dataset was collected by 36 subjects while performing six different activities [90]. The data was recorded using a smartphone having a sampling rate of 20 Hz. The dataset already contained 46 features extracted from fixed-length windows of 10 s each. The activities included 1) walking downstairs, 2) jogging, 3) sitting, 4) standing, 5) walking upstairs, and 6) walking, for a total number of 5,418 instances.

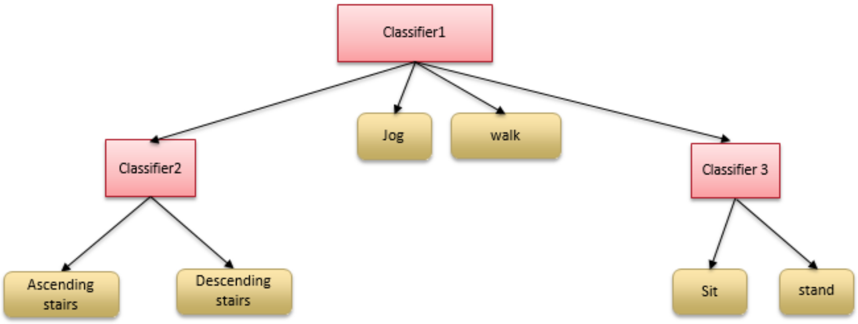

Figure 5.6: Structure of the CFu-tree based hierarchical classification framework on WISDM Dataset

The cluster hierarchy obtained using the proposed CFu-tree framework is shown in fig 5.6. As with the PAMAP2-dataset, we compare the proposed model to flat classifier models and other hierarchical classifiers (table 5.4 and table 5.5). The results of comparing our activity hierarchy with feature correlation-based hierarchy(Fig. 5.7)

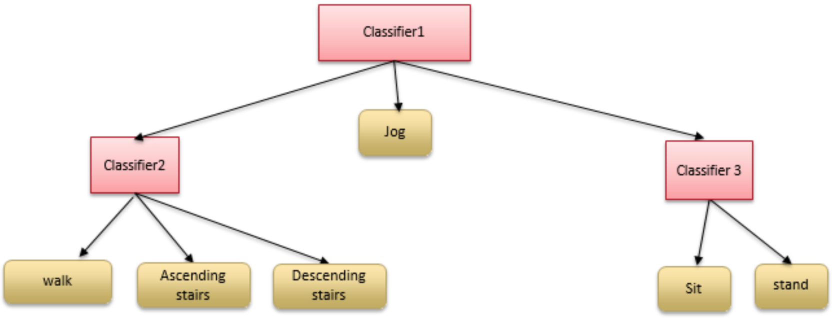

Figure 5.7: Structure of the feature correlation based hierarchical classification framework on WISDM Dataset

reveal recognition accuracy for these models . We report the results in terms of recognition accuracy for the classifiers (Table 5.5). The non-hierarchical models for these classifiers are denoted as NH-Neural Networks, NH-SVM, NH-J48, NH-RBFN, NH-CSCDPL and the hierarchical models are denoted as H-Neural Networks, H-SVM, H-J48, H-RBFN, H-CSCDPL.

Additionally, we examine the relationships existing at different cluster evaluation cores. The hierarchy generated using CFu-tree yielded the lowest Davies-Bouldin index as well as a lower clustering validity index as compared with that of other hierarchical models. For the WISDM dataset, the CFu-tree based hierarchy yielded a higher cophenetic correlation coefficient of 0.89. The resulting activity recognition accuracy using the feature correlation-based hierarchy is 78.95%, and the accuracy for our CFu-tree generated hierarchy is 79.44%, demonstrating that the hierarchical model of CSCDPL performs better than other non-hierarchical and hierarchical models (Table 5.4 and Table 5.5).

Method	NH-CSCDPL	NH-Neural Networks [91]	NH-SVM [91]	NH-J48 [90]	NH-RBFN [90]
Accuracy	**77.1**%	69.5%	70.17%	72.2%	73%

Table 5.4: Comparison of recognition accuracy of all non-hierarchical (NH- prefix) models (calculated in %) on WISDM Dataset.

Method	FC-H-CSCDPL	H-CSCDPL	H-Neural Network	H-SVM	H-RBFN
Accuracy	**78.95**%	**79.44**%	71.6%	74.12%	74.31%

Table 5.5: Comparison of recognition accuracy of all hierarchical (H- prefix) models (calculated in %) on WISDM Dataset.

Our experiments thus verify that compared with other state-of-the-art methods, the use of a hierarchical model for classification of wearable sensor activity allows for efficiencies in both learning and representation by achieving higher accuracy on complex activity datasets. Our research thus adds to a growing body of work exploring how hierarchical structures can be used to improve activity classification. Working with a large, heterogeneous collection of activity data, we successfully extended classifier models to an application that takes advantage of hierarchical structure, for both category learning and run-time efficiencies. In addition, hierarchical structure modeling furnishes evidence that decomposing the problem can lead to more accurate specialized classifiers. Through the creation of hierarchy clusters, each sub-problem becomes smaller than the original problem, and it is sometimes possible to use a much smaller set of features for each classification task. In the future, hierarchical structuring can also be used to determine the negative set for discriminative training, and thus at classification time to combine information from different levels.

5.7 Conclusion

In this paper, we present and evaluate methods for analyzing hierarchical and sequentially structured human activities, aiming to scale activity recognition by creating a hierarchical cluster of activity labels. Instead of using a single flat classifier to distinguish between large numbers of activities, we design a hierarchy of classifiers, each of which distinguishes between child nodes at a particular location in the hierarchy. We hypothesize that building such a hierarchy of activity will improve recognition performance over that of the flat classifier model without affecting the training time.

We validate our method for 15 different activities based on data collected from a PAMAP2 activity recognition dataset and also on WISDM dataset. The hierarchical structure was generated using our proposed CFU-tree clustering approach using the combination of feature mean and spectral mean similarity metric method. Our experiments also demonstrate that the advantage of employing hierarchical activity organization for modeling activities results in important improvement in the recognition rate and reduction of class imbalance inherent in the datasets. Experimental results show that the proposed approach achieves superior accuracy in recognizing activities, far outperforming single-layered supervised learning algorithms.

These results, advancing state of the art in human activity recognition, represent an important step towards reducing the mis-classsification rate that can occur in the analysis of similar activities. Future work is to design an unsupervised feature extraction approach for automatic activity recognition. An unsupervised learning framework of human activity recognition will automatically cluster a large amount of unlabeled similar features into discrete groups of activity, automatically discovering intrinsic patterns from data. Meaningful features learned from input data via a sequence of nonlinear processing can then be combined to build feature hierarchies that provide an effective model for learning visual features and achieving robust recognition rates.

Chapter 6

A DEEP LEARNING FRAMEWORK FOR SENSOR-BASED ECG CLASSIFICATION

6.1 Introduction

Cardiac arrhythmias (abnormal heart rhythms) pose a serious threat to patients recovering from acute myocardial infarction [125]. Some types of arrhythmias are life-threatening, capable of triggering cardiac arrest and sudden death. A non-invasive and inexpensive technique for detecting these disorders is analyzing electrocardiograms (ECGs) that furnish valuable information on the electrophysiology and functional aspects of the cardiovascular system. Therefore, early automatic detection and classification of ECG patterns is critical to diagnosing and treating patients with life-threatening cardiac arrhythmias [126], [127]. Current technologies in wearable sensors allow remote monitoring of physiological data, thus enabling patient's status monitoring. Embedded sensors provide a capability of recording electrocardiographic data (ECG) and electromyographic (EMG) data using different electrode configurations. Thus, physiological monitoring using wearable sensors will help in both diagnosis and ongoing treatment of a vast number of individuals with cardiovascular and pulmonary diseases such as seizures, hypertension and arrythymias.

Previously, several algorithms have focused on automatically classifying heartbeats in ECGs. Feature extraction methods to discriminate heartbeats have included wave shape functions [7], [128], [129], [130], [131], Hermite functions [132], wavelet-based features [133], [134], frequency-based features [135], ECG morphology [136], hermite polynomials [137], higher order cumulant features [138], statistical features

[139], [140] and Karhunen-Loeve expansion of ECG morphology [128]. Methodologies to classify these extracted features have included support vector machines [131], [141], [139], self-organizing maps with learning vector quantization [137], k-th nearest-neighbor rules [142], decision trees [141], artificial neural networks [143], linear discriminants [7], [128], [130], active learning framework [144] and back propagation neural networks [138]. Although some of these are statistically motivated approaches, to the best of our knowledge a deep learning framework has not yet been used to perform ECG classification tasks.

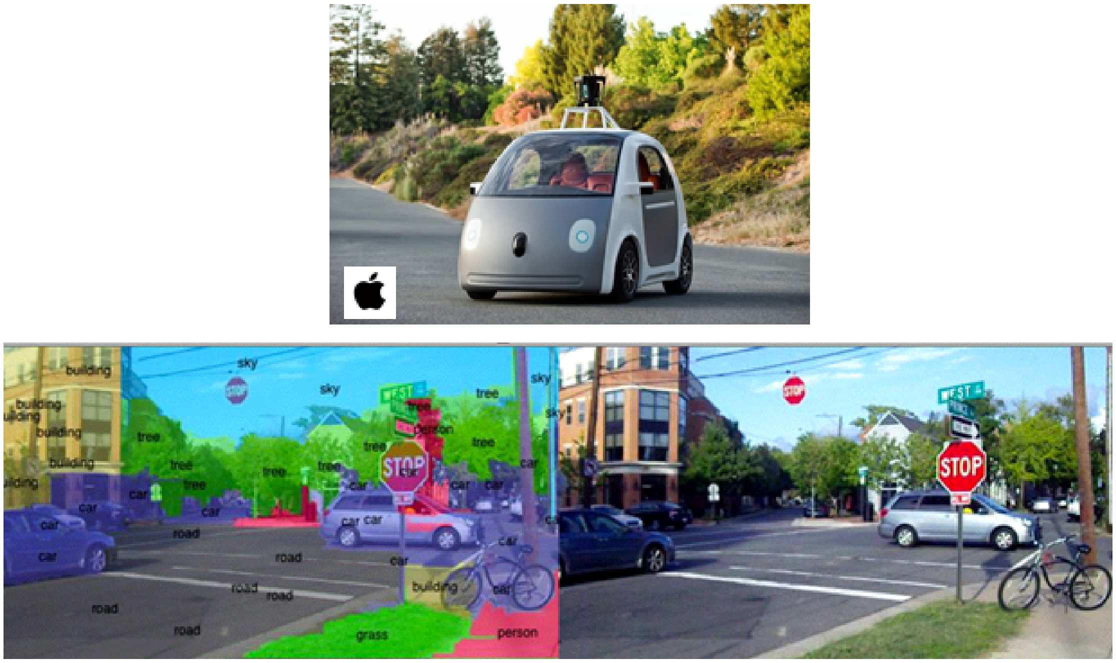

Figure 6.1: Applications of Deep Learning in object recognition and tracking

Deep learning [45], [46], [47], [48], [145], [146], [147], [148], [149] (also known as unsupervised feature learning or representation learning) is a new technique that is becoming mainstream in machine learning and pattern recognition. It has been successfully used in object recognition [150], [151], image verification [152], classification [153], and speech recognition [154], [155], [156], [157]. In recent years, deep learning approaches have dramatically improved the accuracy of recognition tools, creating a deep, multi-stage architecture for unsupervised learning and recognition systems. Deep learning networks are implemented using stacked autoencoders and can represent a highly expressive abstraction. Deep learning networks gain this power by hierarchically composing shallower feature representations into deeper representations and such abstractions can compactly represent a much larger set of functions than shallow networks can. Thus they offer tremendous representational power that can help reveal unknown feature coherences of input signals, an important capability for learning tasks that involve complicated models.

In past decades, computerized recognition of ECGs has become a well-established practice, assisting cardiologists in the task of classifying long-term ECG recordings. However, state-of-the-art automatic ECG recognition systems often rely on a pattern-matching framework that represents an ECG signal as a sequence of stochastic patterns, so they require high sampling rates and thus burdensome computational times to classify arrhythmias. Consequently, to enable implementation in real time and at a reasonable cost, these systems must enlarge their classification criteria by using a set of simple features and a lower sampling rate, and thus must sacrifice accuracy. Here we developed a more accurate and robust approach for single-lead ECG classification that generated fewer false alarms. Inspired by recent progress in the area of deep learning [158], [159], [160], especially its application to speech recognition [154], [155], [157], [161], we developed a deep learning framework that includes Restricted Boltzmann Machine (RBM) and deep belief networks (DBN) [162]. This framework of simple features and a low sampling rate yielded competitive ECG classification performance at lower computational cost, making it a highly practical option in a clinical setting.

The remainder of this chapter is organized as follows. Section 6.2 covers the proposed methodology. Here we present the data processing chain (which includes preprocessing, segmentation, and feature extraction). Section 6.3 briefly describes the proposed deep learning framework. Section 6.4 details the experiments, and provides, evaluates and discusses their results. Section 6.5 concludes the paper.

6.2 Proposed Methodology

A pattern recognition system provides a framework that automatically maps an input signal to a class label by analyzing the features extracted from the signal. The two symbolic stages of this recognition system are feature extraction and classification. Before feature extraction, the data is pre-processed (i.e., filtered), detected and segmented. Then, feature extraction uses mathematical techniques on the input signal to build an association with known models and to obtain the best discriminative representation of the data by exploiting the underlying signal characteristics. Each stage is described below.

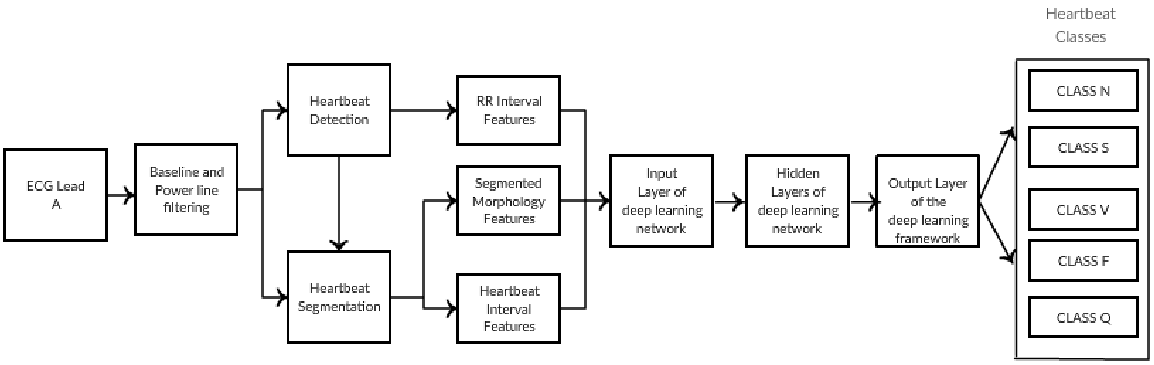
ECG Lead A
Baseline and Power line filtering
Heartbeat Detection
Heartbeat Segmentation
RR Interval Features
Segmented Morphology Features
Heartbeat Interval Features
Input Layer of deep learning network
Hidden Layers of deep learning network
Output Layer of the deep learning framework
Heartbeat Classes
CLASS N
CLASS S
CLASS V
CLASS F
CLASS Q

Figure 6.2: Block diagram of the proposed methodology

6.2.1 Preprocesing

Each ECG signal is first bandpass filtered at 0.110 Hz and sampled at 360 Hz. It is preprocessed to remove artifacts, such as baseline wander, power-line interference, high-frequency noise, and motion artifacts [7]. Baseline wander is a low-frequency artifact that may be caused by chest-lead ECG signals suffering from coughing or breathing with large chest movements, by the poor electrode to skin contact, or by limb-lead ECG signals suffering from arm or leg movements [7]. To remove baseline wander, we pass the signal through median filters with window sizes of 200ms and 600ms, thus removing P-waves, QRS complexes, and T-waves. Power line interference is an interfering voltage with frequencies at integral multiples of 50 Hz that can completely obscure an ECG waveform. This strong interference can stem from improper grounding, loose contact of a patient's cable, or disconnected electrodes. Power-line interference and high-frequency noise are removed from a baseline-corrected ECG using a 12-tap lowpass filter, a finite impulse response filter that has 3 dB at 35 Hz and equal ripple in both pass and stop bands [136]. Motion artifacts represent transient baseline interference that is introduced by electrode-skin impedance caused by electrode motion. Because the peak amplitude of a motion artifact is 500 percent of the peak-to-peak ECG amplitude, and its duration is about 100-500 ms, these artifacts can obscure ECG waveforms, making their interpretation quite difficult. Motion artifacts are removed using an adaptive filter.

6.2.2 Processing

The processing stage employs modules for heartbeat detection and segmentation. For detection, we use the manually verified heartbeat fiducial point times provided with the MIT-BIH arrhythmia database as in [163]. For segmentation, we utilize the heartbeat segmentation program of Laguna [7] since the accuracy of this system in determining heartbeat segmentation points has been validated on the MIT-BIH database and has proven to be commensurate with the inter-expert variation. The heartbeat segmentation stage provides QRS onset and T-wave offset times; a Boolean value indicates the presence/absence of a P-wave and, if present, gives the P-wave onset and offset time for each heartbeat fiducial point.

6.2.3 Feature Extraction

After down-sampling the ECG signal recordings to 114 Hz, we employ two feature extraction methods. Feature Set 1 (FS1) yielded 26 features comprising RR intervals, heartbeat intervals, and segmented morphology. Feature Set 2 (FS2) produced 22 features consisting of RR intervals and fixed interval morphologies [7]. We settled on the single-lead feature extraction method as its lower sampling rate and smaller feature vector both translate to lessened power consumption and lower hardware complexity.

6.2.3.1 Feature Set 1

FS1 consisted of 26 features comprising of RR intervals, heartbeat intervals, and segmented morphologies.

RR Intervals Features

RR intervals also known as Heartbeat fiducial point intervals correspond to the interval between successive heartbeat fiducial points. The following four features were extracted from RR intervals:

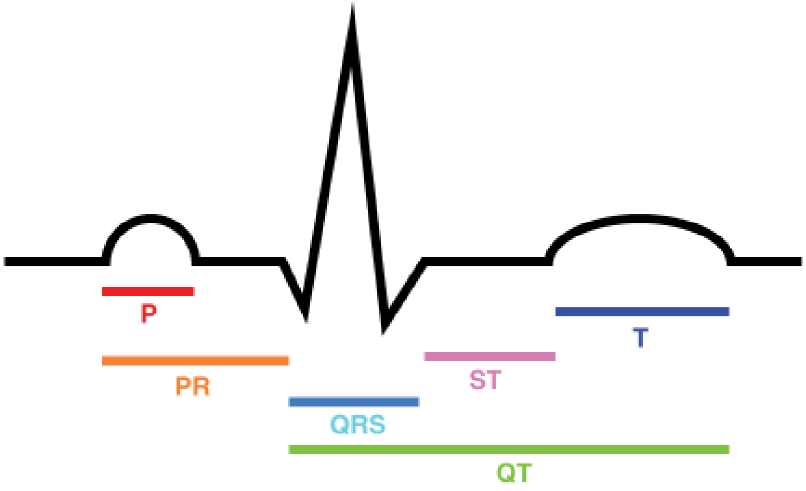

Figure 6.3: ECG Cardiac Trace [6]

- Pre-RR interval: the RR interval between a given heartbeat and the preceding heartbeat.

- Post-RR interval: the RR interval between a given heartbeat and the following heartbeat.

- Average RR interval: the mean of RR intervals for a recording. This value remains the same for all heartbeats in a recording.

- Local average RR interval: estimated by averaging ten RR-intervals surrounding a heartbeat.

Heartbeat Interval Features

Three features were extracted from post-heartbeat interval segmentation.

- QRS duration: time interval between QRS onset and offset.

- T-wave duration: time interval between QRS offset and T-wave offset.

- Boolean variable: a third variable which indicates the presence or absence of a P-wave.

Segmented Morphology Interval Features

Segmented morphology encompasses amplitude values of the ECG signal calculated by a sampling window between QRS onset and offset and a sampling window between QRS offset and T-wave offset points. Two sampling windows were used following the determination of the fiducial point (FP), the first of these bounded by the QRS onset and offset and the second bounded by the QRS offset and the T-wave offset. Ten evenly spaced sample features were derived by uniformly sampling the ECG amplitude in the first window (Fig. 1) and nine more by uniformly sampling the second window, resulting in a total of 19 features.

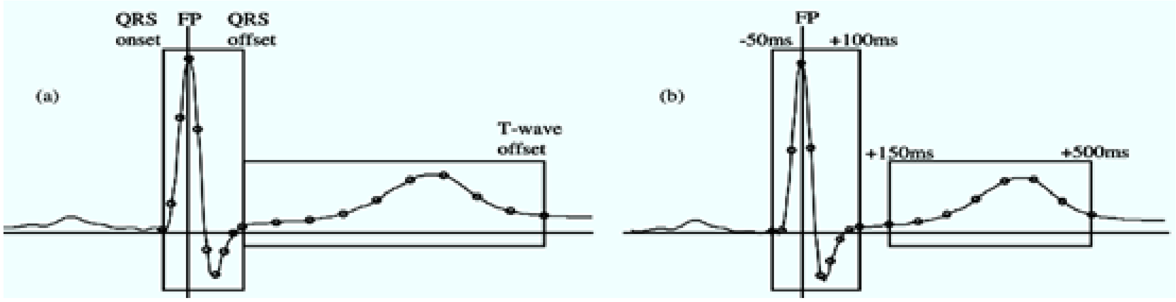

Figure 6.4: Segmented Morphology Intervals Features [7]

6.2.3.2 Feature Set 2

FS2's 22 features consisted of RR intervals and fixed interval morphologies [7].

RR Intervals Features

RR intervals (also known as Heartbeat fiducial point intervals) correspond to the interval between successive heartbeat fiducial points, and match the same four features extracted in Feature Set 1.

Fixed-interval morphology features

To determine fixed interval morphologies, sampling windows were first positioned at the heartbeat FP. Two sampling windows were formed based on FP. The first window approximately encompassed the QRS-complex and covered the portion of the ECG between FP-50 ms and 100 ms. Nine samples of the ECG between FP-50ms and FP+100ms were extracted from this window. The second window approximately covered the T-wave and started at 150 ms and finished at 500 ms. The next nine samples between FP+150ms and FP+500ms were extracted from the second window, for a total of 18 features used in FS2.

The entire feature extraction can be summarized as follows:

- **FS1 (26)**: RR intervals (4), heartbeat intervals (3), segmented morphology (19)
- **FS2 (22)**: RR intervals (4), fixed interval morphology (18)

6.3 Deep Learning Framework

Deep Learning, inspired by the human brain's deep hierarchical architecture, is a technique focused on learning deep hierarchical models of data [164], [165]. This system learns an empirical set of features at multiple levels of abstraction thereby allowing it to acquire complex functions from input data without using human-engineered features.

Deep learning networks, implemented using stacked autoencoders, are capable of representing highly expressive abstractions, thereby compactly yielding much larger sets of functions than shallow networks can [166]. Through the tremendous representational power of hierarchical feature learning, these networks can help discover unknown feature coherences of input signals, a characteristic that is crucial for learning tasks involving complicated models.

As suggested in [167], the central concept of a DBN training algorithm is to first initialize greedily the weights of each layer in an unsupervised manner by treating each pair of layers as a Restricted Boltzmann Machine (RBM), and to later jointly refine these weights to further improve the likelihood. The resulting DBN can be considered a hierarchy of nonlinear feature detectors that can capture complex statistical patterns in the data.

6.3.1 Restricted Boltzmann Machine

Derived from a Boltzmann Machine, the RBM is a bi-directionally connected network of stochastic processing units that learns significant features of an unknown probability distribution based on samples from that distribution. An RBM can be described as a bipartite graph having a visible layer and a hidden layer (Fig. 4). Units in the visible layer are typically characterized by Bernoulli or Gaussian distributions and those in the hidden layer are typically characterized by Bernoulli distributions. Stochastic units in the visible layer associate with stochastic units in the hidden layer by means of a weight matrix. No connections exist between units in the same layer. In schematic representation, each edge in the bipartite graph is attached to a weight, denoted as a symmetric matrix W, that is associated with the visible layer (v) and the hidden layer (h).

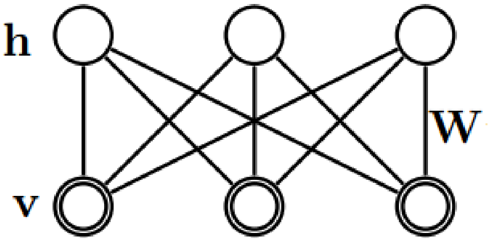

Figure 6.5: Schematic of a restricted Boltzmann machine

A given RBM defines an energy function for every configuration of visible and hidden state vectors. If both v and h are binary states (i.e., the Bernoulli-Bernoulli RBM) the energy function is given by

$$\mathrm{E}(v, h) = -v^T W h - b_v^T v - b_h^T h \tag{6.1}$$

Thus, an RBM represents the joint distribution $p(v; h)$ between visible unit v and hidden random unit h. The joint probability is defined as

$$p(v, h) = \frac{\exp(-\mathrm{E}(v, h))}{Z}, \tag{6.2}$$

where $Z = \sum_v \sum_h \exp(-\mathrm{E}(v, h))$ is the partition function

The probability assigned by the network model to a visible unit v is

$$p(v) = \frac{1}{Z} \sum_h \exp(-\mathrm{E}(v, h)), \tag{6.3}$$

The lack of connections within a given layer of an RBM results in the visible layer variables being conditionally independent, given the hidden layer variables, and vice versa. Thus the conditional probabilities can be rewritten as:

$$p(v_j = 1/h) = \sigma(a_i + \sum h_j w_i, j) \tag{6.4}$$

$$p(h_j = 1/v) = \sigma(b_j + \sum v_i w_i, j) \tag{6.5}$$

where σ is the sigmoid fnuction defined by $\sigma = \frac{1}{1+\exp^{-x}}$

Signal propagation manifests in two ways: recognition, where visible activations propagate to the hidden units; and reconstruction, where hidden activations propagate to the visible units. Both recognition and reconstruction use the same weight matrix (simply transposed). The Contrastive Divergence (CD) algorithm finds the parameters W, a, and b and performs Gibbs sampling [50]. We use CD to minimize the reconstruction error so the weights can be trained to generate input patterns that are presented to the RBM with high probability. (A guide to training an RBM is given in [168]).

6.3.2 Deep Belief Networks

DBNs are a type of multi-layer generative neural network that is recognized for its capability to model and visualize high-level learned features [164], [169]. It is composed of stacked, logistic RBMs wherein the lowest-level RBM learns a shallow model of the data and the next-level RBM learns to model first-layer hidden units, thereby representing high-level abstraction through hierarchical architecture (Fig. 5).

When a DBN is used for classification purposes, the RBM pre-training procedure can be used to initialize the weights of the deep neural network, and then these weights can be discriminatively fine-tuned by back-propagating error derivatives. The recognition weights of the DBN become the weights of a standard neural network. This unsupervised pre-training establishes the platform for a final training phase: a fine-tuning process with respect to a supervised training criterion based on gradient

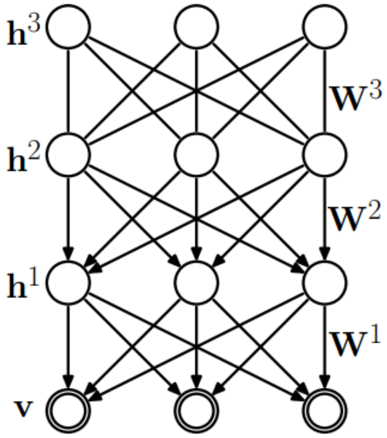

Figure 6.6: Schematic of a deep belief network of three layers.

descent optimization. (A detailed description of DBN varieties and their training is available in [150], [170]).

6.4 Evaluation of Proposed Methodology with results and discussion

6.4.1 The MIT-BIH Database

For our evaluation experiments, we used the acclaimed MIT/Beth Israel Hospital (BIH) Arrhythmia Database available at MIT medical data storage Physionet [163]. Briefly, this database incorporates 48 half-hour ECG recordings, each containing two ECG lead signals digitized at 360 samples per second with 11-bit resolution over a 10 mV range [163]. Twenty-three of the recordings were randomly selected from a set of 4,000 ambulatory 24-hour ECGs that were collected from a mixed population of inpatients and outpatients at the medical center. The remaining 25 recordings were selected from the same set but included less common but clinically symbolic arrhythmias. All recordings have been annotated by two or more cardiologists and contain modified limb lead II. In our experiments, we focused on using lead A only. In 45 recordings, lead A is modified lead II, and in the other three recordings, lead A is lead V5 [129]. According to the Association for the Advancement of Medical Instrumentation (AAMI) recommended practice, the four paced beats are excluded from this experimental evaluation process because these beats possess insufficient signal quality for reliable processing [7], [128].

6.4.2 AAMI Standard

MIT-BIH heartbeat types are combined according to AAMI recommendations [171]. Since the AAMI standard emphasizes the problem of distinguishing ventricular ectopic beats (VEBs) from non-ventricular ectopic beats [7], normal and arrhythmic beats are remapped to the five AAMI heartbeat classes [172], [171] using the mapping in [7] with each class including heartbeats of one or more types. Thus we used AAMI recommended practice to combine the MIT-BIH heartbeat types into the following five heartbeat classes that we used in all subsequent processing:

1. Class N corresponding to beats originating in the sinus node (normal and bundle branch block beat types)
2. Class S corresponding to supraventricular ectopic beats (SVEBs)
3. Class V corresponding to ventricular ectopic beats (VEBs)
4. Class F corresponding to beats that result from fusing normal and VEBs
5. Class Q corresponding to unknown beats including paced beats

6.4.3 Evaluation Metrics

The MIT-BIH database contained a series of manually verified QRS detection points that we utilized in this study. After the four recordings containing paced beats were removed as in [172], the remaining 44 recordings were divided into two equal-sized datasets, each containing ECG data from 22 recordings. The first dataset (DS1) was used to train the classifier and to set parameter values that optimized the performance of the classifier. The second dataset (DS2) was employed to carry out an independent and unbiased performance evaluation of the heartbeat classification system. To validate the algorithms on the MIT-BIH database, we used the following performance metrics:

accuracy (Acc), sensitivity (Se), positive predictive value (PPV), and false positive rate (FPR).

$$Accuracy(ACC) = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.6}$$

$$Sensitivity(Se) = \frac{TP}{TP + FN} \tag{6.7}$$

$$PositivePredictiveValue(PPV) = \frac{TP}{TP + FP} \tag{6.8}$$

$$FalsePositiveRate(FPR) = \frac{FP}{TN + FP} \tag{6.9}$$

where TP is a true positive that reflects the number of heartbeats belonging to a particular class A that are accurately classified to that class A; FN is a false negative that reflects the number of heartbeats belonging to class A that are inaccurately classified to a different class B; FP is a false positive that reflects the number of heartbeats belonging to class B that are inaccurately classified to class A; and TN is a true negative that reflects the number of heartbeats belonging to class B that are accurately classified to that class B.

6.4.4 Experimental Results and Discussion

We performed our classification on the MIT-BIH arrhythmia database [163] to detect two types of heartbeat arrhythmias: VEBs and SVEBs. In agreement with AAMI recommended practice, four recordings containing paced beats were removed from the 48 recordings. The data from the remaining 44 recordings were divided into two sets: training (DS1) and test (DS2). We trained the classifier using DS1 and assessed classifier performance using DS2.

For the RBM and DBN algorithms, we used the toolbox developed by Drausin Wulsin [175]. To determine the best configurations and parameters for the RBM, we

Method	Rate (Hz)	SVEB				VEB			
		Acc	Se	PPV	FPR	Acc	Se	PPV	FPR
FS1+Deep Learning	360	93.13	88.39	33.63	6.68	96.63	77.74	69.20	2.17
FS2+Deep Learning	360	93.47	70.99	32.44	5.66	95.24	85.22	56.63	4.11
Chazel et al [7]	360	94.6	75.9	38.5	4.7	97.4	77.7	81.9	1.2
Chazel et al [129]	360	93.6	61.2	31.2	5.2	95.4	72.4	62.3	3.0
Chazel et al [129]	360	94.4	73.5	37.0	4.8	97.8	87.6	80.3	1.5

Table 6.1: Comparison of classification results at sampling rate of 360 Hz

Method	Rate (Hz)	SVEB				VEB			
		Acc	Se	PPV	FPR	Acc	Se	PPV	FPR
FS1+Deep Learning	114	93.63	88.62	35.49	6.17	95.57	78.49	59.65	3.34
FS2+Deep Learning	114	93.42	59.16	30.10	5.26	95.87	85.54	60.83	3.47
FS1+LCKSVD [173]	114	93.4	75.12	32.84	5.89	93.51	76	49.97	5.27
FS2+LCKSVD [173]	114	94.61	68.86	37.52	4.39	97.18	80.44	70.13	1.65
LDA_Basil [174]	114	-	-	-	-	93.4	75.8	61.9	4.8
QDA_Basil [174]	114	-	-	-	-	83.1	97	35.2	18.4
ANN_Basil [174]	114	-	-	-	-	96.9	79.7	74.6	1.9

Table 6.2: Comparsion of classification results at sampling rate of 114 Hz

performed a large number of experiments where we used varying combinations of batch sizes (i.e., number of training vectors used in each pass of each epoch for the contrastive divergence algorithm), numbers of hidden units, learning rates, and numbers of stacked RBMs. The final classification layer had five output units, one for each class, and the unit with the highest activation level was considered the most probable class.

We have reported our ECG classification results at sampling rates of 360 Hz and 114 Hz in Table 6.1 and Table 6.2, respectively. Column 1 indicates the methodology and column 2 indicates the sampling rate; columns 3-10 indicate the gross classifier performance in terms of Acc (Accuracy), Se (Sensitivity), PPV (Positive Predictive value) and FPR (False positive rate) for the two heartbeat types. Rows 1 and 2 in both tables report the overall performance of our classification using Feature Set 1 and

Deep Learning and using Feature Set 2 and DBN, respectively. The independent performance assessment of the configuration of FS1 and Deep learning framework resulted in an accuracy of 93.13%, a sensitivity of 88.39%, a positive predictivity of 33.63%, and an FPR of 6.68% for the SVEB class. For the VEB class, accuracy was 93.63%, sensitivity 77.74%, positive predictivity 69.20%, and FPR 2.17%.

Method	N	S	V	F	Q
N	31228	2758	841	7011	422
S	38	1539	126	36	2
V	54	275	2173	293	0
F	16	3	5	360	0
Q	0	1	1	2	2

Table 6.3: Comparison of FS1 + Deep Learning Framework at Sampling Rate of 360 Hz

Method	N	S	V	F	Q
N	73.89%	6.5%	1.9%	16.59%	0.99%
S	2.18%	88.39%	7.23%	2.06%	0.11%
V	1.93%	9.83%	77.74%	10.48%	0%
F	4.16%	0.78%	1.30%	93.75%	0%
Q	0%	16.66%	16.66%	33.33%	33.33%

Table 6.4: Classification results in % for FS1 + Deep Learning Framework at Sampling Rate of 360 Hz

Method	N	S	V	F	Q
N	36386	2347	1470	2045	12
S	148	1236	354	3	0
V	96	225	2382	91	1
F	22	1	103	258	0
Q	1	1	1	0	1

Table 6.5: Comparison of FS2 + Deep Learning Framework at Sampling Rate of 360 Hz

Our classification at a sampling rate of 360Hz provided an accuracy of 93.13% for the SVEB class and 96.63% for the VEB class. Our classification at a sampling

Method	N	S	V	F	Q
N	86.1%	5.55%	3.47%	4.83%	0.02%
S	8.5%	70.99%	20.33%	0.17%	0%
V	3.43%	8.05%	85.22%	3.25%	0.04%
F	5.72%	0.26%	26.82%	67.18%	0
Q	25%	25%	25%	0%	25%

Table 6.6: Classification results in % for FS2 + Deep Learning Framework at Sampling Rate of 360 Hz

Method	N	S	V	F	Q
N	31238	2565	1366	6451	640
S	41	1543	118	39	0
V	60	238	2194	303	0
F	17	1	22	344	0
Q	1	0	2	0	1

Table 6.7: Comparison of FS1 + Deep Learning Framework at Sampling Rate of 114 Hz

Method	N	S	V	F	Q
N	73.91%	6.06%	3.23%	15.26%	1.51%
S	2.35%	88.62%	6.77%	2.24%	0%
V	2.14%	8.51%	78.49%	10.8%	0%
F	4.42%	0.26%	5.72%	89.58%	0%
Q	25%	0%	50%	0%	25%

Table 6.8: Classification results in % for FS1 + Deep Learning Framework at Sampling Rate of 114 Hz

Method	N	S	V	F	Q
N	36213	2189	1181	2648	29
S	347	1030	358	6	0
V	100	201	2391	100	3
F	19	1	113	251	0
Q	1	1	1	0	1

Table 6.9: Comparison of FS2 + Deep Learning Framework at Sampling Rate of 114 Hz

rate of 114Hz provided similar levels of accuracy, 93.63% and 95.57% for SVEB and

Method	N	S	V	F	Q
N	85.69%	5.17%	2.79%	6.26%	0.07%
S	19.93%	59.16%	20.56%	0.34%	0%
V	3.57%	7.19%	85.54%	3.57%	0.1%
F	4.94%	0.26%	29.42%	65.36%	0%
Q	25%	25%	25%	0%	25%

Table 6.10: Classification results in % for FS2 + Deep Learning Framework at Sampling Rate of 114 Hz

VEB classes, respectively. Thus at both sampling rates, our algorithm provides competitive accuracy performance when compared to previously reported results (rows 3-5 in Table 6.1 and rows 3-7 in Table 6.2) for automated heartbeat classification systems in [7] and [129].

Comparing Table 6.1 against Table 6.2, our results demonstrate that a deep learning algorithm framework is better suited to detect VEB and SVEB types of arrhythmia at the lower sampling rate of 114 Hz. Since increasing the sampling rate to 360 Hz did not provide any significant gain in performance, it follows that an 114 Hz sampling rate can provide sufficient discriminatory power for this classification task.

In addition to evaluating the performance of VEB and SVEB classification, we also assessed the per-class classification across all five classes at sampling rates of 360 Hz and 114 Hz (Tables 6.3-6.10). Our per-class classification results are competitive when compared to results from methods in [7], [129], [174]. Since varying the sampling rate had minimal impact on performance, we conclude that our approach emulates the performance of state-of-the-art models at a lower sampling rate and with a set of simple features.

Notably, the combination of parameters that yielded the best result was 112 hidden units, a batch size of 42, and a learning rate of 0.00001. With this combination, the DBN achieved a very low error rate of 4.7 %. We used this DBN configuration while varying the number of stacked RBMs and found that outputs of three layered trained RBMs achieved the best performance results. Moreover, this performance level was comparable to that of state-of-the-art ECG classification algorithms.

In summary, we demonstrate that our approach can emulate state-of-the-art classification results while using a significantly lower sampling rate. In the case of an average sampling rate of 114 samples versus 360, we are achieving a gain factor of three. In fact, the smaller feature-set representation and the deep learning framework together contain sufficient discriminative information for accurate ECG classification. Thus, with a suitable choice of parameters, the classifiers built using this deep learning framework provide competitive performance. Also, our proposed framework opens a new window for future research, highlighting the enormous potential of deep learning based methods for accurate classification of other physiological signals, such as arterial blood pressure (ABP), electromyograms (EMG), and heart rate variability (HRV).

6.5 Conclusion

In this work, we considered the application of Restricted Boltzmann Machines and Deep Belief Networks to the automatic classification of single-lead ECG signals. Experimental results indicate that our deep learning framework demonstrates a classification accuracy of 93.47% for SVEB class signals and 95.24% for VEB class signals on the MIT-BIH database at a sampling rate of 360 Hz. Thus our framework provides performance competitive with that of state-of-the-art methods. Experimental results also demonstrate that this framework provides similar classification accuracy (93.63% for SVEB and 95.57% for VEB) when sampling at only 114 Hz. Thus a lower sampling rate of 114 Hz is sufficient to provide good discriminatory power for the ECG classification task.

In conclusion, our approach emulated the performance of state-of-the-art models using a lower sampling rate and a set of simple features. In future work, we will investigate other types of embedding that represent ECG recordings as a feature vector and then use hierarchical deep learning algorithms for robust performance. We will also extend our framework to the classification of sensor-based cognitive assessment data and the recognition of daily life activities, areas critical in healthcare for ubiquitous health computing and medical informatics.

Chapter 7

CONCLUSION AND FUTURE WORK

In this dissertation, a variety of algorithms are proposed based on dictionary learning and deep learning to solve different machine learning problems, such as classification, supervised learning and feature selection. The motivation is that many natural signals have a sparse structure, where only a few non-zero elements are capable of representing the majority of information conveyed by the target signal. Inspired by the sparse signal representation paradigm, we developed several robust dictionary learning algorithms for wearable sensor-based activity recognition, supervised learning, hierarchical classification and feature selection. We have empirically shown the effectiveness of proposed frameworks for recognition and obtained significant improvement over commonly used classification algorithms on different kinds of data sets. The remainder of the chapter presents a summary of major contributions of this thesis and proposes directions for future research.

7.1 Summary of research outcomes

The main contributions of the thesis can be summarized as follows:

- A comprehensive review of conventional sparse representation techniques and state- of-the-art dictionary learning algorithms is conducted in Chapter 2. The chapter also presents the dictionary pair learning concept wherein an analysis and a synthesis dictionary are jointly employed for learning and classification.

- A novel class specific dictionary pair learning algorithm method for classification is introduced in Chapter 3 [82]. The main contribution here involves the class specific regularizer term that ensures sparse codes belonging to the same class will be concentrated, thereby proving beneficial for the classification stage. A combination of an alternating direction method of multipliers and a $l_1 - l_s$ minimization method is adopted to approximately minimize the objective function. The proposed algorithm was tested on four recognition tasks and this proposed algorithm led to higher recognition rate when compared with other state-of-the-art methods.

- In Chapter 4, a novel maximum correntropy criterion based dictionary pair learning algorithm is introduced [1], [94]. Maximum correntropy criterion has demonstrated to efficiently handle non-Gaussian noise and is more insensitive to outliers. To develop a more tractable approach, we employ a combination of alternating direction method of multipliers and an iteratively reweighted method to approximately minimize the objective function. In general, the proposed training method requires lower training time compared to existing methods with much higher classification rates for activity recognition problems defined on four datasets.

- To reduce the problem of mis-classifications between similar activity classes, we designed a conceptual clustering based hierarchy of classifiers, each of which distinguishes between child nodes at a particular location in the hierarchy in Chapter 5 [97], [98]. A CFu-tree-based graphical model was employed along with feature mean and spectral mean similarity metric to design the multi-layer classifier. Through the creation of hierarchy clusters, each sub-problem become smaller than the original problem and building such a hierarchy of classifiers improved recognition performance compared to the flat classifiers.

- Chapter 6 presents the novel concept of applying deep learning methodology to the classification of single-lead electrocardiogram (ECG) signals [162]. The effectiveness of this proposed algorithm is illustrated using real ECG signals from the widely-used MIT-BIH database and simulation results demonstrate that RBM and DBN can achieve high average recognition accuracies on ventricular ectopic beats (93.63%) and on supraventricular ectopic beats (95.57%) at a low sampling rate of 114 Hz. Experimental results indicate that classifiers built into this deep-learning-based framework achieve state-of-the-art performance using lower sampling rates and simple features than traditional methods. Further, employing features extracted at a sampling rate of 114 Hz when combined with deep learning was found to provide enough discriminatory power for the classification task.

In summary, the new paradigm of dictionary and deep learning is exploited to develop robust machine learning classifier algorithms. The proposed algorithms are capable of representing the salient information by only using fewer parameters. This enables the design of high performance algorithms for supervised training, model optimization, classification and feature selection. Experimental results show that the proposed dictionary and deep learning algorithms are superior to existing methods.

7.2 Future Work

Wearable electronics devices such as heart rate monitors, smart watches continue to expand in health-care sectors and consumer sectors. Fitness devices are by far the most mature market, and products with embedded sensors that track and analyze physical or other movements will soon change the future of medical technology. Wearable technologies will thus affect our future, impacting our health and fitness decisions, redefining the doctor-patient relationship and thereby reducing healthcare cost and time [176].

In this dissertation, we had introduced novel dictionary and deep learning frameworks aimed at development of a personalized physical activity monitoring system and sensor based ECG signal recognition system applicable for everyday life scenarios. There are possible future research directions to continue and extend the work presented in this thesis. We outline them in the following paragraphs:

Online sparse learning and ensemble learning

Our models, i.e supervised learning algorithms, are offline learning algorithms in which the performance depends on the available data. However, it would be interesting to extend the proposed methods to online learning. Also, another interesting area of investigation is to apply the dictionary pair learning algorithms as classifier ensembles. Implementing dictionary learning based ensemble methods can further improve the generalization ability of the machine learning algorithm.

Extended Dataset

The existing datasets for wearable sensors have limited activities involving few volunteers. High accuracy has been obtained while distinguishing basic tasks like sitting, standing and walking. However, erroneous classifications were observed while dealing with composite activities like playing soccer, running and walking. A promising way to overcome this was by designing hierarchical classifier. The proposed hierarchical classifier can be extended to adopt a more robust clustering algorithm with similarity metric. Such cases of mis-classifications occurring in similar activities necessitates developing of a large database which involves many similar activities (for instance: preparing food, eating food or standing , talking while standing) belonging to larger set of subjects from the different age groups. Distinguishing such closely similar activities will be an interesting task and approaches which can distinguish such high level or composite activities needs to be developed.

Use of Semi-supervised learning

While considering extending the datasets to include more activities, obtaining corresponding annotated ground truth for recorded sensory data will not be an easy task and would require greater human efforts. Semi-supervised learning algorithms are known to achieve good recognition rate while requiring small amount of labeled data with large amounts of unlabeled data. With the underlying aim of adding new extended activities in the datasets , semi-supervised learning methods will deserve further attention.

Multi-modal Learning

This thesis has opened an area of research for dictionary and deep learning and its possible wide range of applications in wearable sensor technology. Our present system has been implemented on accelerometer, gyroscope, HR rate and an ECG recognition system. This system can be extended with additional sensors to increase the accuracy of the already provided functions or can be extended as systems with new functionality. Potential real world applications with new functionality can be about detecting stress, tiredness or assessing cognitive load by the multi- modular system.

A multi-modal learning framework can be envisioned as the one that uses multiple input domains for learning and inference. (For example, when it is hard to identify an activity with just the accelerometer data,it can be useful to examine the gyroscope, heart rate and additional sensors aside from also examining the accelerometer). A multi-modal learning framework will make it easier to use unlabeled data as the learning algorithm can more robustly infer about the underlying labels by the use of an unified learning framework rather than depending on manually engineering features for each sensor.

High Level feature learning from Unlabeled data

In chapter 6, we demonstrated that incorporating invariance in DBNs allowed learning useful high-level features and achieving good performance in recognition tasks. It is highly desirable for a good feature representation to be invariant as invariance for a feature mapping results in feature values to be close to the maximal value when the input data is slightly transformed from the optimal input and yet the feature mapping will be selective to the optimal input. A solution is exploiting topographic representations [177], [178], however, this is challenging as real-world data is very complex and highly-variable. Therefore, achieving more invariant feature representations, discovering hierarchical representations and learning high-level features from unlabeled data will be one of many fundamental questions that we need to continuously address.

BIBLIOGRAPHY

[1] S. M. Mathews, C. Kambhamettu, and K. E. Barner, “Maximum correntropy based dictionary learning framework for physical activity recognition using wearable sensors,” in *International Symposium on Visual Computing Advances in Visual Computing.* Springer, 2016, pp. 123–132.

[2] S. Patel, H. Park, P. Bonato, L. Chan, and M. Rodgers, “A review of wearable sensors and systems with application in rehabilitation,” *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, p. 1, 2012.

[3] A. Reiss, “Personalized mobile physical activity monitoring for everyday life,” Ph.D. dissertation, Technical University of Kaiserslautern, 2014.

[4] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications.* Cambridge University Press, 2012.

[5] T. Peng and L. Liu, “A novel incremental conceptual hierarchical text clustering method using cfu-tree,” *Applied Soft Computing*, vol. 27, pp. 269–278, 2015.

[6] “Electrocardiogram (EKG) Stress Test holter monitor, http://www.hopkinsmedicine.org/healthlibrary/conditions,” Online accessed 20-April-2015.

[7] P. De Chazal, M. O’Dwyer, and R. B. Reilly, “Automatic classification of heartbeats using ECG morphology and heartbeat interval features,” *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.

[8] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, “Activity classification using realistic data from wearable sensors,” *IEEE Transactions on information technology in biomedicine*, vol. 10, no. 1, pp. 119–128, 2006.

[9] J. Lester, T. Choudhury, and G. Borriello, “A practical approach to recognizing physical activities,” in *International Conference on Pervasive Computing.* Springer, 2006, pp. 1–16.

[10] C. Pham and P. Olivier, “Slice&dice: Recognizing food preparation activities using embedded accelerometers,” in *European Conference on Ambient Intelligence.* Springer, 2009, pp. 34–43.

[11] B. Logan, J. Healey, M. Philipose, E. M. Tapia, and S. Intille, "A long-term evaluation of sensing modalities for activity recognition," in *International conference on Ubiquitous computing.* Springer, 2007, pp. 483–500.

[12] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *International Conference on Pervasive Computing.* Springer, 2004, pp. 1–17.

[13] J. Hoey, T. Plötz, D. Jackson, A. Monk, C. Pham, and P. Olivier, "Rapid specification and automated generation of prompting systems to assist people with dementia," *Pervasive and Mobile Computing*, vol. 7, no. 3, pp. 299–318, 2011.

[14] S. Consolvo, D. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby *et al.*, "Activity sensing in the wild: a field trial of ubifit garden," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, 2008, pp. 1797–1806.

[15] M. Rabbi, S. Ali, T. Choudhury, and E. Berke, "Passive and in-situ assessment of mental and physical well-being using mobile sensors," in *Proceedings of the 13th international conference on Ubiquitous computing.* ACM, 2011, pp. 385–394.

[16] T. Plötz, N. Y. Hammerla, A. Rozga, A. Reavis, N. Call, and G. D. Abowd, "Automatic assessment of problem behavior in individuals with developmental disabilities," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing.* ACM, 2012, pp. 391–400.

[17] A. J. Perez, M. A. Labrador, and S. J. Barbeau, "G-sense: a scalable architecture for global sensing and monitoring," *IEEE Network*, vol. 24, no. 4, pp. 57–64, 2010.

[18] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1082–1090, 2008.

[19] B. E. Ainsworth, W. L. Haskell, M. C. Whitt, M. L. Irwin, A. M. Swartz, S. J. Strath, W. L. O Brien, D. R. Bassett, K. H. Schmitz, P. O. Emplaincourt *et al.*, "Compendium of physical activities: an update of activity codes and met intensities," *Medicine and science in sports and exercise*, vol. 32, no. 9; SUPP/1, pp. S498–S504, 2000.

[20] A. Reiss and D. Stricker, "Aerobic activity monitoring: towards a long-term approach," *Universal Access in the Information Society*, vol. 13, no. 1, pp. 101–114, 2014.

[21] X. Long, B. Yin, and R. M. Aarts, "Single-accelerometer-based daily physical activity classification," in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE.* IEEE, 2009, pp. 6107–6110.

[22] K. K. I. L. S. H. J. Mi-hee Lee, Jungchae Kim and S. K. Yoo, "Physical activity recognition using a single tri-axis accelerometer," in *Proceedings of the world congress on engineering and computer science*, vol. 1, 2009.

[23] M. Ermes, J. Pärkkä, and L. Cluitmans, "Advancing from offline to online activity recognition with wearable sensors," in *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS).* IEEE, 2008, pp. 4451–4454.

[24] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman, "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor," in *Wearable Computers, 2007 11th IEEE International Symposium on.* IEEE, 2007, pp. 37–40.

[25] J. Parkka, M. Ermes, K. Antila, M. van Gils, A. Manttari, and H. Nieminen, "Estimating intensity of physical activity: a comparison of wearable accelerometer and gyro sensors and 3 sensor locations," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE.* IEEE, 2007, pp. 1511–1514.

[26] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *AAAI*, vol. 5, 2005, pp. 1541–1546.

[27] A. Reiss and D. Stricker, "Introducing a modular activity monitoring system," in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE.* IEEE, 2011, pp. 5621–5624.

[28] P. Dohnálek, P. Gajdoš, and T. Peterek, "Tensor modification of orthogonal matching pursuit based classifier in human activity recognition," in *Nostradamus 2013: Prediction, Modeling and Analysis of Complex Systems.* Springer, 2013, pp. 497–505.

[29] H. Gjoreski, S. Kozina, M. Gams, M. Lustrek, J. A. Álvarez-García, J.-H. Hong, J. Ramos, A. K. Dey, M. Bocca, and N. Patwari, "Competitive live evaluations of activity-recognition systems," *Pervasive Computing, IEEE*, vol. 14, no. 1, pp. 70–77, 2015.

[30] A. M. Martınez, G. I. Webb, S. Chen, and N. A. Zaidi, "Scalable learning of bayesian network classifiers," 2015.

[31] T.-K. Huang and J. Schneider, "Spectral learning of hidden markov models from dynamic and static data," in *Proceedings of the 30th International Conference on Machine Learning*, 2013, pp. 630–638.

[32] D. A. Clifton, L. Clifton, S. Hugueny, D. Wong, and L. Tarassenko, "An extreme function theory for novelty detection," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 7, no. 1, pp. 28–37, 2013.

[33] D. M. Bradley and J. A. Bagnell, “Differential sparse coding,” 2008.

[34] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *Image Processing, IEEE Transactions on*, vol. 15, no. 12, pp. 3736–3745, 2006.

[35] C. Poultney, S. Chopra, Y. L. Cun *et al.*, “Efficient learning of sparse representations with an energy-based model,” in *Advances in neural information processing systems*, 2006, pp. 1137–1144.

[36] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online learning for matrix factorization and sparse coding,” *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 19–60, 2010.

[37] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image super-resolution as sparse representation of raw image patches,” in *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2008.* IEEE, 2008, pp. 1–8.

[38] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” in *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2009.* IEEE, 2009, pp. 1794–1801.

[39] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[40] K. Huang and S. Aviyente, “Sparse representation for signal classification,” in *Advances in neural information processing systems*, 2006, pp. 609–616.

[41] J. Mairal, M. Elad, and G. Sapiro, “Sparse learned representations for image restoration,” in *Proc. of the 4th World Conf. of the Int. Assoc. for Statistical Computing (IASC).* Citeseer, 2008.

[42] D. L. Donoho, “Compressed sensing,” *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[43] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, “Compressed sensing for real-time energy-efficient ecg compression on wireless body sensor nodes,” *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 9, pp. 2456–2466, 2011.

[44] E. J. Candè and M. B. Wakin, “An introduction to compressive sampling,” *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[45] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *ECCV Computer Vision.* Springer, 2014, pp. 818–833.

[46] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014, pp. 512–519.

[47] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1701–1708.

[48] L. Deng, O. Abdel-Hamid, and D. Yu, "A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 6669–6673.

[49] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 625–660, 2010.

[50] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 689–696.

[51] B. A. Olshausen *et al.*, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[52] B. A. Olshausen, "Sparse coding of time-varying natural images," in *Proc. of the Int. Conf. on Independent Component Analysis and Blind Source Separation.* Citeseer, 2000, pp. 603–608.

[53] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems*, 2006, pp. 801–808.

[54] M. Unser, "Sampling-50 years after shannon," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569–587, 2000.

[55] M. Fornasier and H. Rauhut, "Compressive sensing," in *Handbook of mathematical methods in imaging.* Springer, 2011, pp. 187–228.

[56] M. Aharon, M. Elad, and A. Bruckstein, "¡ img src="/images/tex/484. gif" alt="\ rm k"¿-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[57] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on signal processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[58] K. Kreutz-Delgado and B. D. Rao, "Focuss-based dictionary learning algorithms," in *International Symposium on Optical Science and Technology.* International Society for Optics and Photonics, 2000, pp. 459–473.

[59] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[60] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 3360–3367.

[61] K. Engan, S. O. Aase, and J. Husoy, "Frame based signal compression using method of optimal directions (mod)," in *Circuits and Systems, 1999. ISCAS'99. Proceedings of the 1999 IEEE International Symposium on*, vol. 4. IEEE, 1999, pp. 1–4.

[62] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on.* IEEE, 2008, pp. 1–8.

[63] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 2691–2698.

[64] D.-S. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on.* IEEE, 2008, pp. 1–8.

[65] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on.* IEEE, 2011, pp. 625–632.

[66] A. Gersho and R. M. Gray, *Vector quantization and signal compression.* Springer Science & Business Media, 2012, vol. 159.

[67] R. Jenatton, J. Mairal, F. R. Bach, and G. R. Obozinski, "Proximal methods for sparse hierarchical dictionary learning," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 487–494.

[68] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce, "Discriminative sparse image models for class-specific edge detection and image interpretation," in *European Conference on Computer Vision.* Springer, 2008, pp. 43–56.

[69] J. Mairal, J. Ponce, G. Sapiro, A. Zisserman, and F. R. Bach, "Supervised dictionary learning," in *Advances in neural information processing systems*, 2009, pp. 1033–1040.

[70] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, “Learning mid-level features for recognition,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 2559–2566.

[71] N. Zhou, Y. Shen, J. Peng, and J. Fan, “Learning inter-related visual dictionary for object recognition,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on.* IEEE, 2012, pp. 3490–3497.

[72] M. Yang, L. Zhang, X. Feng, and D. Zhang, “Fisher discrimination dictionary learning for sparse representation,” in *2011 International Conference on Computer Vision.* IEEE, 2011, pp. 543–550.

[73] X.-C. Lian, Z. Li, C. Wang, B.-L. Lu, and L. Zhang, “Probabilistic models for supervised dictionary learning.” in *CVPR*, 2010, pp. 2305–2312.

[74] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos, “Positive definite dictionary learning for region covariances,” in *2011 International Conference on Computer Vision.* IEEE, 2011, pp. 1013–1019.

[75] F. Perronnin, “Universal and adapted vocabularies for generic visual categorization,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 30, no. 7, pp. 1243–1256, 2008.

[76] X. Zhang, D. Schuurmans, and Y.-l. Yu, “Accelerated training for matrix-norm regularization: A boosting approach,” in *Advances in Neural Information Processing Systems*, 2012, pp. 2906–2914.

[77] I. Ramirez, P. Sprechmann, and G. Sapiro, “Classification and clustering via dictionary learning with structured incoherence and shared features,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 3501–3508.

[78] A. Shrivastava, H. V. Nguyen, V. M. Patel, and R. Chellappa, “Design of non-linear discriminative dictionaries for image classification,” in *Asian Conference on Computer Vision.* Springer, 2012, pp. 660–674.

[79] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, “Design of non-linear kernel dictionaries for object recognition,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5123–5135, 2013.

[80] M. Soltanolkotabi, E. Elhamifar, E. J. Candes *et al.*, “Robust subspace clustering,” *The Annals of Statistics*, vol. 42, no. 2, pp. 669–699, 2014.

[81] M. Yang, L. Zhang, J. Yang, and D. Zhang, “Metaface learning for sparse representation based face recognition,” in *2010 IEEE International Conference on Image Processing.* IEEE, 2010, pp. 1601–1604.

[82] S. M. Mathews, C. Kambhamettu, and K. E. Barner, “Class specific centralized dictionary learning for wearable sensors based physical activity recognition,” in *Annual Conference on Information Sciences and Systems (CISS), 2016*.

[83] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Projective dictionary pair learning for pattern classification,” in *Advances in Neural Information Processing Systems*, 2014, pp. 793–801.

[84] B.-D. Liu, L. Gui, Y. Wang, Y.-X. Wang, B. Shen, X. Li, and Y.-J. Wang, “Class specific centralized dictionary learning for face recognition,” *Multimedia Tools and Applications*, pp. 1–19, 2015.

[85] A. Reiss, G. Hendeby, and D. Stricker, “A novel confidence-based multiclass boosting algorithm for mobile physical activity monitoring,” *Personal and Ubiquitous Computing*, vol. 19, no. 1, pp. 105–121, 2015.

[86] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, “A public domain dataset for human activity recognition using smartphones.” in *ESANN*, 2013.

[87] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, “Transition-aware human activity recognition using smartphones,” *Neurocomputing*, vol. 171, pp. 754–767, 2016.

[88] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, “Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine,” in *International Workshop on Ambient Assisted Living*. Springer, 2012, pp. 216–223.

[89] C. A. Ronaoo and S.-B. Cho, “An optimal feature selection method based on random forests for activity recognition with smartphone sensors,” pp. 448–450, 2014.

[90] J. Kwapisz, G. Weiss, and S. Moore, “Activity recognition using cell phone accelerometers. sigkdd explor. newsl. 12 (2), 74–82 (2011),” 2013.

[91] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, “Activity recognition using cell phone accelerometers,” *ACM SigKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[92] W. Liu, P. P. Pokharel, and J. C. Príncipe, “Correntropy: properties and applications in non-gaussian signal processing,” *IEEE Transactions on Signal Processing*, no. 11, pp. 5286–5298, 2007.

[93] J. C. Principe, D. Xu, and J. Fisher, “Information theoretic learning,” *Unsupervised adaptive filtering*, pp. 265–319, 2000.

[94] S. M. Mathews, C. Kambhamettu, and K. Barner, "Maximum correntropy based dictionary learning framework for physical activity recognition using wearable sensors," *Submitted for publication at Image Vision Computing (IVC).*

[95] F. Nie, J. Yuan, and H. Huang, "Optimal mean robust principal component analysis," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1062–1070.

[96] K. Bache and M. Lichman, "UCI machine learning repository, URL http://archive.ics.uci.edu/ml."

[97] S. M. Mathews, C. Kambhamettu, and K. Barner, "Hierarchical Class Specific Centralized Dictionary Pair Learning for Wearable Sensors based Activity Recognition," *Submitted for publication at Pattern Recognition (PR).*

[98] S. M. Mathews, C. Kambhamettu, and K. E. Barner, "A Hierarchical Dictionary Learning framework for physical activity recognition using wearable sensors," *In Preparation to IEEE International Conference on Image Processing (ICIP 2017).*

[99] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2009, pp. 17–24.

[100] M. Ermes, J. Pärkkä, J. Mäntyjärvi, and I. Korhonen, "Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, no. 1, pp. 20–26, 2008.

[101] K.-J. Kim, M. M. Hassan, S.-H. Na, and E.-N. Huh, "Dementia wandering detection and activity recognition algorithm using tri-axial accelerometer sensors," in *Proceedings of the 4th International Conference on Ubiquitous Information Technologies & Applications, ICUT*. IEEE, 2009, pp. 1–5.

[102] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 48–53, 2010.

[103] Y. Yang, "An evaluation of statistical approaches to text categorization," *Information retrieval*, vol. 1, no. 1-2, pp. 69–90, 1999.

[104] S. D'Alessio, K. Murray, R. Schiaffino, and A. Kershenbaum, "The effect of using hierarchical classifiers in text categorization," in *Content-Based Multimedia Information Access-Volume 1*, 2000, pp. 302–313.

[105] S. Dumais and H. Chen, "Hierarchical classification of web content," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2000, pp. 256–263.

[106] D. Mladenic, "Turning yahoo into an automatic web-page classifier," 1998.

[107] M. Sasaki and K. Kita, "Rule-based text categorization using hierarchical categories," in *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, vol. 3. IEEE, 1998, pp. 2827–2830.

[108] K. Wang, S. Zhou, and Y. He, "Hierarchical classification of real life documents." in *SDM*. SIAM, 2001, pp. 1–16.

[109] A. S. Weigend, E. D. Wiener, and J. O. Pedersen, "Exploiting hierarchy in text categorization," *Information Retrieval*, vol. 1, no. 3, pp. 193–216, 1999.

[110] Y.-C. F. Wang and D. Casasent, "New support vector-based design method for binary hierarchical classifiers for multi-class classification problems," *Neural Networks*, vol. 21, no. 2, pp. 502–510, 2008.

[111] T. Van Kasteren, A. Noulas, G. Englebienne, and B. Kröse, "Accurate activity recognition in a home setting," in *Proceedings of the 10th international conference on Ubiquitous computing*. ACM, 2008, pp. 1–9.

[112] Y. Guan, A.-A. Ghorbani, and N. Belacel, "Y-means: A clustering method for intrusion detection," 2003.

[113] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[114] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[115] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE transactions on pattern analysis and machine intelligence*, no. 2, pp. 224–227, 1979.

[116] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[117] J. C. Dunn, "Well-separated clusters and optimal fuzzy partitions," *Journal of cybernetics*, vol. 4, no. 1, pp. 95–104, 1974.

[118] M. Halkidi and M. Vazirgiannis, "Clustering validity assessment: Finding the optimal partitioning of a data set," in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*. IEEE, 2001, pp. 187–194.

[119] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proceedings of the eleventh international conference on Information and knowledge management*. ACM, 2002, pp. 515–524.

[120] W. Hwang, T. Kim, M. Ramanathan, and A. Zhang, “Bridging centrality: graph mining from element level to group level,” in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2008, pp. 336–344.

[121] R. R. Sokal and F. J. Rohlf, “The comparison of dendrograms by objective methods,” *Taxon*, pp. 33–40, 1962.

[122] W. M. Rand, “Objective criteria for the evaluation of clustering methods,” *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

[123] E. B. Fowlkes and C. L. Mallows, “A method for comparing two hierarchical clusterings,” *Journal of the American statistical association*, vol. 78, no. 383, pp. 553–569, 1983.

[124] D. H. Hu, V. W. Zheng, and Q. Yang, “Cross-domain activity recognition via transfer learning,” *Pervasive and Mobile Computing*, vol. 7, no. 3, pp. 344–358, 2011.

[125] A. Guyton and J. Hall, “11th textbook of medical physiology,” 2006.

[126] L. Glass, “Cardiac oscillations and arrhythmia analysis,” in *Complex Systems Science in Biomedicine.* Springer, 2006, pp. 409–422.

[127] “ACC/AHA clinical competence statement on electrocardiography and ambulatory electrocardiography,” *Journal of the American College of Cardiology*, vol. 38, no. 7, pp. 2091–2100, 2001.

[128] P. de Chazal and R. B. Reilly, “A patient-adapting heartbeat classifier using ECG morphology and heartbeat interval features,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 12, pp. 2535–2543, 2006.

[129] P. de Chazal, “Detection of supraventricular and ventricular ectopic beats using a single lead ECG,” in *35th Annual International Conference of IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 45–48.

[130] M. Llamedo and J. P. Martínez, “Heartbeat classification using feature selection driven by database generalization criteria,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3, pp. 616–625, 2011.

[131] C. Ye, B. V. Kumar, and M. T. Coimbra, “Heartbeat classification using morphological and dynamic features of ECG signals,” *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2930–2941, 2012.

[132] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo, “Clustering ECG complexes using hermite functions and self-organizing maps,” *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 7, pp. 838–848, 2000.

[133] T. Ince, S. Kiranyaz, and M. Gabbouj, “A generic and robust system for automated patient-specific classification of ECG signals,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1415–1426, 2009.

[134] X. Jiang, L. Zhang, Q. Zhao, and S. Albayrak, “ECG arrhythmias recognition system based on independent component analysis feature extraction,” in *IEEE TENCON Region Conference*, 2006, pp. 1–4.

[135] L. Senhadji, G. Carrault, J. Bellanger, and G. Passariello, “Comparing wavelet transforms for recognizing cardiac patterns,” *in IEEE Engineering in Medicine and Biology Magazine*, vol. 14, no. 2, pp. 167–173, 1995.

[136] Y. H. Hu, S. Palreddy, and W. J. Tompkins, “A patient-adaptable ECG beat classifier using a mixture of experts approach,” *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 9, pp. 891–900, 1997.

[137] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo, “Clustering ECG complexes using hermite functions and self-organizing maps,” *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 7, pp. 838–848, 2000.

[138] S. Osowski and T. H. Linh, “ECG beat recognition using fuzzy hybrid neural network,” *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 11, pp. 1265–1271, 2001.

[139] S. Osowski, L. T. Hoai, and T. Markiewicz, “Support vector machine-based expert system for reliable heartbeat recognition,” *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 4, pp. 582–589, 2004.

[140] G. de Lannoy, D. François, J. Delbeke, and M. Verleysen, “Weighted conditional random fields for supervised interpatient heartbeat classification,” *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 1, pp. 241–247, 2012.

[141] J. Rodriguez, A. Goni, and A. Illarramendi, “Real-time classification of ECGs on a PDA,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 1, pp. 23–34, 2005.

[142] I. Christov, I. Jekova, and G. Bortolan, “Premature ventricular contraction classification by the kth nearest-neighbours rule,” *Physiological measurement*, vol. 26, no. 1, p. 123, 2005.

[143] W. Jiang and S. G. Kong, “Block-based neural networks for personalized ECG signal classification,” *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1750–1761, 2007.

[144] J. Wiens and J. V. Guttag, “Active learning applied to patient-adaptive heartbeat classification,” in *Advances in neural information processing systems*, 2010, pp. 2442–2450.

[145] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, "Pedestrian detection with unsupervised multi-stage feature learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3626–3633.

[146] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in *IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 1385–1392.

[147] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1653–1660.

[148] A. Jain, J. Tompson, M. Andriluka, G. W. Taylor, and C. Bregler, "Learning human pose estimation features with convolutional networks," *arXiv preprint arXiv:1312.7302*, 2013.

[149] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 5, p. 169, 2014.

[150] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[151] V. Nair and G. E. Hinton, "3D object recognition with deep belief nets," in *Advances in Neural Information Processing Systems*, 2009, pp. 1339–1347.

[152] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Advances in Neural Information Processing Systems*, 2014, pp. 1988–1996.

[153] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[154] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[155] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[156] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks." in *Interspeech*, 2011, pp. 437–440.

[157] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: An overview," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 8599–8603.

[158] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR)*, vol. 1, 2005, pp. 539–546.

[159] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1875–1882.

[160] Y. Sun, X. Wang, and X. Tang, "Hybrid deep learning for face verification," in *IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 1489–1496.

[161] A. R. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, M. Picheny *et al.*, "Deep belief networks using discriminative features for phone recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5060–5063.

[162] S. M. Mathews, C. Kambhamettu, and K. E. Barner, "A Deep Learning Framework for Single-lead ECG classification," *Submitted for publication at Journal of Multimedia Tools and Applications (MTA).*

[163] R. Mark and G. Moody, "MIT-BIH arrhythmia database," *3rd Edition*, 1997.

[164] Q. V. Le, "Building high-level features using large scale unsupervised learning," in *IEEE International Conference onAcoustics, Speech and Signal Processing (ICASSP),*. IEEE, 2013, pp. 8595–8598.

[165] Y. Bengio, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[166] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[167] L. Deng and D. Yu, "Deep learning," *Signal Processing*, vol. 7, pp. 3–4, 2014.

[168] G. E. Hinton, "A practical guide to training Restricted Boltzmann Machines," in *Neural Networks: Tricks of the Trade.* Springer, 2012, pp. 599–619.

[169] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual International Conference on Machine Learning.* ACM, 2009, pp. 609–616.

[170] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[171] *Recommended practice for testing and reporting performance results of ventricular arrhythmia detection algorithms*, 1986.

[172] *Testing and Reporting Performance Results of Cardiac Rhythm and ST-segment Measurement Algorithms*, 1999.

[173] S. M. Mathews, L. F. Polania, and K. E. Barner, "Leveraging a discriminative dictionary learning algorithm for single-lead ecg classification," in *41st Annual Northeast Biomedical Engineering Conference (NEBEC).* IEEE, 2015, pp. 1–2.

[174] T. Basil, B. S. Chandra, and C. Lakshminarayan, "A comparison of statistical machine learning methods in heartbeat detection and classification," in *Big Data Analytics.* Springer, 2012, pp. 16–25.

[175] D. Wulsin, "DBN Toolbox, Department of Bioengineering, University of Pennsylvania (2010)."

[176] "Wearable Technology Reports, URL http://www.futuresource-consulting.com," 2014-09-19.

[177] Z. Chen, S. Xiong, Z. Fang, R. Zhang, X. Kong, and Y. Rong, "Topologically ordered feature extraction based on sparse group restricted boltzmann machines," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[178] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.