MOBILE SENSING BASED HUMAN STRESS MONITORING FOR SMART HEALTH APPLICATIONS

CHUN TUNG LI

PhD

The Hong Kong Polytechnic University

2021

The Hong Kong Polytechnic University
Department of Computing

Mobile Sensing Based Human Stress Monitoring for Smart Health Applications

Chun Tung LI

A thesis submitted in partial fulfilment
of the requirements for the degree of

Doctor of Philosophy

October 2020

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

____________________ (Signed)

Chun Tung LI (Name of student)

Abstract

In the last decade, the advancement of the Internet of Things (IoT) enables communication and computation to appear anytime and anywhere. Numerous mobile devices and sensing equipment are interconnected that realize continuous data collection and analysis, which we refer to as mobile sensing. Based on the integration of artificial intelligence (AI) and IoT, health-related information could be extracted from data collected and provides real-time intervention automatically that has led to the emergence of smart health (s-Health) applications. However, the fundamental issues posed by the new paradigm of healthcare create barriers against the practical use of the s-Health system. One key challenge is the dynamic nature of human responses under different health conditions. It raises the need for calibration of the system to provide personalized diagnosis, making it difficult to scale for a large population and increases the cost of the initial setup. The other challenge stems from the widely used machine learning techniques that require a vast amount of training data to build the model.

To unleash the full potential of s-Health, methods that can sense the general set of health indicators and efficiently predict the influences are necessary. This thesis focuses on the monitoring of stress as one major factor that affects our physical and mental health. We carry out a series of studies to 1) detect and recognize repetitive activities related to health conditions; 2) measure the symptoms and predict the impact of stress, using data collected from mobile devices. This thesis made three main contributions.

The first contribution is the general approaches proposed for monitoring repetitive

activities such as exercises, heartbeat, and respiration. We consider the repeated physical motion as patterns that occur consecutively in multivariate time series obtained from mobile devices. We proposed a multiple-length successive similar pattern detector (mSIMPAD) to detect repetitive activities using the sensor data. The mSIMPAD has barely any assumption regarding the target pattern and scales linearly, that can naturally adapt to different individuals and efficient enough to deploy on resource-constrained devices. On this basis, we proposed a scalable template extraction method (STEM) to locate and identify repeating patterns from multivariate time series. It substitutes the commonly used sliding-window-based technique and achieved more generic and efficient monitoring of repetitive activity. We also demonstrate the proposed approaches have a wide range of applications including a use case of respiration monitoring using wireless signals.

The second contribution is the investigation of stress recognition using mobile devices. There is positive stress (eustress) and negative stress (distress) that affects our mental status and altering our behavior in different aspects. We developed a data collection platform using smartphones, wearable sensors, and computers and conducted an empirical study to examine the feasibility of stress recognition by exploiting the data collected. We found that physical and behavioral data help discriminate eustress from distress defined by its effect on performance.

The third contribution is the work on predicting the impact of stress. We proposed a computational continuous stress performance prediction (CCSP) method that leverages domain knowledge to model the interaction between stress and cognitive performance over time. An experiment was designed and conducted in a rigorous laboratory environment. Data includes cognitive performance, physiological signals, and the concentrations of cortisol in saliva were collected during the experiment. Our computational model shows improvement in prediction performance on a small dataset with the aid of domain knowledge. It also shed light on the problem of cognitive performance prediction (increase, decrease, same) by estimating the physical stress symptoms with mobile devices.

List of Publications

1. **C.T. Li**[*], Y. Yang[*], J. Shen, J. Cao, and M. Stojmenovic. "How Continuous Stress Affects Cognition: A Computational Model Towards Cognitive Performance Prediction".[†]

2. **C.T. Li**, J. Shen, Y. Yang, J. Cao, and M. Stojmenovic. "Repetitive Activity Monitoring from Multivariate Time Series: A Generic and Efficient Approach".[†]

3. **C.T Li**, J. Shen, C. Ma, K.H. Wong, and J. Cao. "An Agnostic and Efficient Approach to Identifying Features from Execution Traces".[†]

4. C. Ma, **C.T Li**, and J. Cao. "Run-and-Tell: A Trace-driven Approach to Automated Functionality Identification of MobileApps".[†]

5. **C.T. Li**, J. Cao, X. Liu, and M. Stojmenovic. "mSIMPAD: Efficient and Robust Mining of Successive Similar Patterns of Multiple Lengths in Time Series". Accepted by *ACM Transactions on Computing for Healthcare*, 2020.

6. Tim M.H. Li, **C.T. Li**, Paul W.C. Wong, and J. Cao. "Withdrawal Behaviors and Mental Health Among College Students". In *Behavioral Psychology*, 2017.

7. **C.T. Li**, J. Cao, and Tim M.H. Li. "Eustress or Distress: An Empirical Study of Perceived Stress in Everyday College Life". In *ACM International Joint Conference on Pervasive and Ubiquitous Computing* Adjunct, 2016.

[*]Co-first Author

[†]In submission

Acknowledgements

I would like to express my very great appreciation to my advisor Prof. Jiannong Cao, who has the attitude and substance of a genius. Without his insightful guidance, patient supervision, and valuable comments, this dissertation would not have been possible. I would also like to thank Prof. Xue (Steve) Liu of McGill University, for his advice and guidance during my visit to his lab. My appreciation also goes to Prof. Milos Stojmenovic of Singidunum University, for his effort and attention to my research work. Both professors generously giving me their time have been very much appreciated.

I am also profoundly grateful for the substantial contribution of my advisors and co-authors. Thanks, Dr. Jiaxing Shen, Dr. Tim Man-Ho Li, Dr. Chao Ma, Mr. Yang Yu, and Ms. Yanni Yang for all of the hard work.

My appreciation also extends to my laboratory colleagues. They are Dr. Linchuan Xu, Dr. Wengen Li, Dr. Yuqi Wang, Dr. Xuefeng Liu, Dr. Guanqing Liang, Mr. Jiaqi Wen, Dr. Yanwen Wang, Dr. Xiulong Liu, Dr. Yang Lei, Dr. Divya Saxena, and Dr. Tarun Kulshrestha. I am sincerely grateful for the companion, supports, and suggestions generously offered by my colleagues as well as young researchers in our lab, they are Mr. Hangqing Wu, Mr. Shan Jiang, Mr. Ruosong Yang, Ms. Jia Wang, Mr. Yuvraj Sahni, Mr. Zhuo Li, Mr. Zhiyuan Wen, Mr. Mingjin Zhang, Mr. Shuaiqi Liu, Mr. Qianyi Chen, Mr. Zhixuan Liang, Mr. Ka Ho Wong, Mr. Zhongyu Yao, Mr. Cheung Leong Tung, and Mr. Marco Tumaini.

Special thanks to many talented researchers I met in PolyU and during my visit to McGill University, Dr. Fei Gu, Dr. Hang Luo, Dr. Xiao Bin, Dr. Mohammed Aquil

Maud Mirza, Dr. Hiu Fung Ng, Dr. Dorothy Chau, Dr. Landu Jiang, Mr. Chen Ma, Mr. Qinglong Wang, and many others. Thanks for their valuable support and constructive recommendations on both research and life.

In addition, I would like to thank the Hong Kong Polytechnic University for all the opportunities, resources, and financial support. I also appreciate the warm assistance from the staff of our department. They are Ms. Carman Au, Ms. Christy Au, Ms. Esther Ku, Ms. Jolie Chick, Ms. Anna Cheng, Mr. Fu Wang, and many others.

Finally, I wish to thank my parents, family members, and friends for their support and encouragement throughout my study.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Population aging has aroused grave concerns all over the world. Most medical systems have gradually begun to digitize and as a result healthcare service incorporating information technology has emerged. Smart Health (s-Health) is an umbrella term for technology that aims to promote health. It is capable to collect health information from various sources, processing this information, and deliver advice and action to improve medical status in an intelligent manner. In this thesis, we focus on the smart health application in monitoring human stress as it plays a vital role in our physical and mental health. The presence of stress is unavoidable that has become an emerging risk of public health. It is related to various health issues including heart diseases, obesity, diabetes, depression, and anxiety. Beyond the individual health problem, stress is also an issue for the economy. The financial cost of job stress in the US is about $300 billion annually [9], as a result of accidents, absenteeism, employee turnover, diminished productivity, direct medical, legal, and insurance costs, and so on.

Given the importance of stress in our health and wellbeing, an increasing number of studies have been published monitoring stress using mobile devices. A large portion of the literature investigates the detection of stress using physiological parameters such as heart rate variability, galvanic skin response, pupil diameter, etc; and multi-modal data including speech, facial expression, physical activities, and behavior. The results are promising in which the state-of-the-art can classify if stress occurs for different people under different contexts. However, the presence and effect of stress are complicated. While most of the previous work simplifies the problem as a binary classification that overlooked the positive impact of stress and therefore ignored the situation where stress is sometimes desired.

The measurement of stress and its effect is crucial for improving stress resilience, the ability to adapt and recover from strain. Understanding the relationships between stress and its impact provides a knowledge basis for stress monitoring strategy, to promote physical and mental wellbeing, and increases efficiency in workplaces. In view of this, we aim to fill the research gap by studying the measurement of stress and its impact, to provide more suitable interventions for better stress resilience, and eventually promote general health. Particularly, we investigated the methods for repetitive activities detection and recognition to estimate various health indicators including respiration and exercises. Also, we studied the measurement of stress, and a computational approach to predict the impact of stress on cognitive performance. In theory, this will enable s-Health applications by facilitating the general assessment of health conditions and the discovery of stressful behaviors in the near future.

Measuring the repetitive activities and stress, as well as its impact on our body is non-trivial given the following reasons. First of all, the existing approaches are usually scenario dependant where providing a generic method to all possible circumstances is challenging. Second, stress is an internal mental state that is difficult to quantify for measurement and prediction. Third, the relationship between the mental state and its response is complicated. The response might differ from one to another due to individual differences, and various types of stimuli. To address these challenges, we focus on the measure of a general set of daily activities - repetitive activities - as it relates to our health and mental state in many ways. We also focused on the objective measure of symptoms such as behavioral and physiological parameters to estimate the perceived stress, and the computational model to predict the change of cognitive ability. In advance of the detailed explanation of our research framework, we first introduce the background of stress including the definition and essential knowledge to illustrate the rationale of this thesis.

1.1.1 What is Stress?

Everyone has the experience of feeling the pressure from various life events. Surprisingly, the definition of stress is still under debate in the research community due to its complex nature. There are multiple aspects of studying stress related to psychology and physiology, yet no consensus of understanding stress was made by far.

The most widely adopted definition of stress stems from Hans Selye, who coined the term in 1936 [120]. He defined stress as "*the nonspecific response of the body to any demand made upon it*" [122]. In general terms, stress is the biological and psychological response to any external stimulus perceived as a threat. Those stimuli are called stressors, which can be physical (e.g. fighting off a physical attack, putting hands into cold water, physical office clutter) or psychological (e.g. facing humiliation, time pressure on a decision, financial worries). The perception of stressor is a psychological process, where a situation is perceived as a threat when the estimated demands are greater than the resources that the individual has available [10]. Therefore, a situation that is stressful for someone might not be stressful for another one, which makes it difficult to study with subjectivity.

The degree of perceived stress determines the body's response through the autonomic nervous system, which controls involuntary body functions such as breathing, blood pressure, and heartbeat. A stressful stimulus will activate the sympathetic nervous system (SNS), and it regulates bodily functions by signaling the adrenal glands to releases hormones including adrenalin and cortisol into the bloodstream. The reaction as a survival mechanism is also known as the "fight-or-flight" response [8]. It results in several physiological changes such as increased heartbeat, breath, and muscle tension, that prepare one's body to counter the threatening situation. After the stressful situation, the parasympathetic nervous system (PNS) that responsible for the "rest and digest" processes, reduces the stress response and restores homeostasis.

Some suggest that stress could be the stress itself, the cause, as well as the result of it. In this thesis, we stick with the original definition and consider stress as the "*response*" of any stressful events. More specifically, we regard stress as the "*symptoms*" that could be objectively measured, such as cortisol, physiological and behavioral changes. While the "*perceived stress*' is the subjective evaluation of the stress feeling. There is a good explanation in [124], where the authors suggest that "there is an underlying property called stress for which both the symptom measurements and the reported measures (perceived stress) are approximations". Stress could be derived from the symptom measurements using machine learning models to learn the underlying model of true stress. They also coined the word *computed stress* to distinguish from traditional stress measure, defined as the "stress computationally derived from instantaneous measures of stress symptoms obtained by non-invasive methods". In the rest of the thesis, we regard stress derived from the computational model as computed stress and used with stress interchangeably.

1.1.2 Impact of Stress

Based on general stress, researchers also proposed different types of stress. Regarding the duration, it can be classified as *Acute Stress*, a short-term stress response that usually lasts no more than a couple of hours (e.g. football game, public speech, taking an exam). *Chronic Stress*, a long-term stress that usually lasts for several days to years, and the individual has little or no control of it. The prolonged activation of SNS affects numerous health outcomes. It gradually increases resting heart rate, blood pressure, respiration rate, which also increases the risk of cardiovascular diseases including heart disease, high blood pressure, and stroke. Exposure to stress for a long period can lead to serious mental issues such as depression, anxiety, and personality disorders.

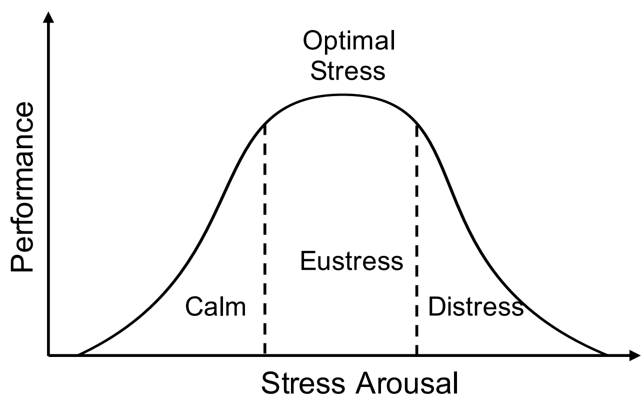


Figure 1.1: Relation between stress and performance based on the Yerkes-Dodson Law.

Considering the types and impacts of stress, Selye also introduced the concept of positive and negative stress, namely the Eustress and Distress accordingly [122]. He distinguishes eustress and distress in terms of the adaptive and non-adaptive effects of the stress response. Eustress is the "healthy, positive, constructive results of stressful events and stress response"; In contrast, distress is associated with negative feelings and physical impairments [66]. Besides, there is an opinion that eustress and distress can be distinguished quantitatively based on the law of Yerkes-Dodson [145]. It suggests that stress is beneficial to the performance until some optimal level is reached, after which performance will decline. The relationship between stress and performance follows the inverted U shape as shown in figure 1.1. Among different attempts to understand the concept of eustress, the common root is the adaptive nature of the response, in which eustress should bring positive health effects. While the two dominating approaches used to distinguish eustress from distress is depending on the interpretation of the feeling, and the impact on performance.

Most of the previous work focuses on the unpleasant aspect of stress. Although the concept of eustress has been introduced in the early 70s, there is surprisingly little work studied about it. Therefore, we investigated the positive stress by quantitatively measure the symptoms of stress and the effect of it. In the next section, we will provide a detailed description of the research framework and the scope of this thesis.

1.2 Research Focus

Stress presents in almost every aspect of life that could seem too broad to study. In order to monitor stress for better health, being able to measure stress and its impact is key. It allows the quantification of stress and its effect that can facilitate the design of stress management strategy. This research focuses on the measurement of health related repetitive activities, estimate the symptoms and predict the impacts of stress using data collected from mobile devices. The health indicators and stress responses span along with different aspects of a person from physical, mental, to behavioral, which outlines this research work. In this section, we first introduce the overview of the research area and the key challenges, then we introduce the research framework to tackle the challenges raised from the research question.

1.2.1 Stress Monitoring: An Overview

We provide an overview of the research area in figure 1.2, offering some examples of symptoms and impacts in each of the aspects mentioned above. From the physical point of view, stress can lead to several physical changes in the body such as increased heartbeat, blood pressure, respiration rate, and muscle tension. The physical symptoms can be monitored by mobile devices and potentially used for estimating the stress of a person. When people experiencing stress for a long period, it might cause serious health issues such as heart diseases, high blood pressure, obesity, and diabetes [10].

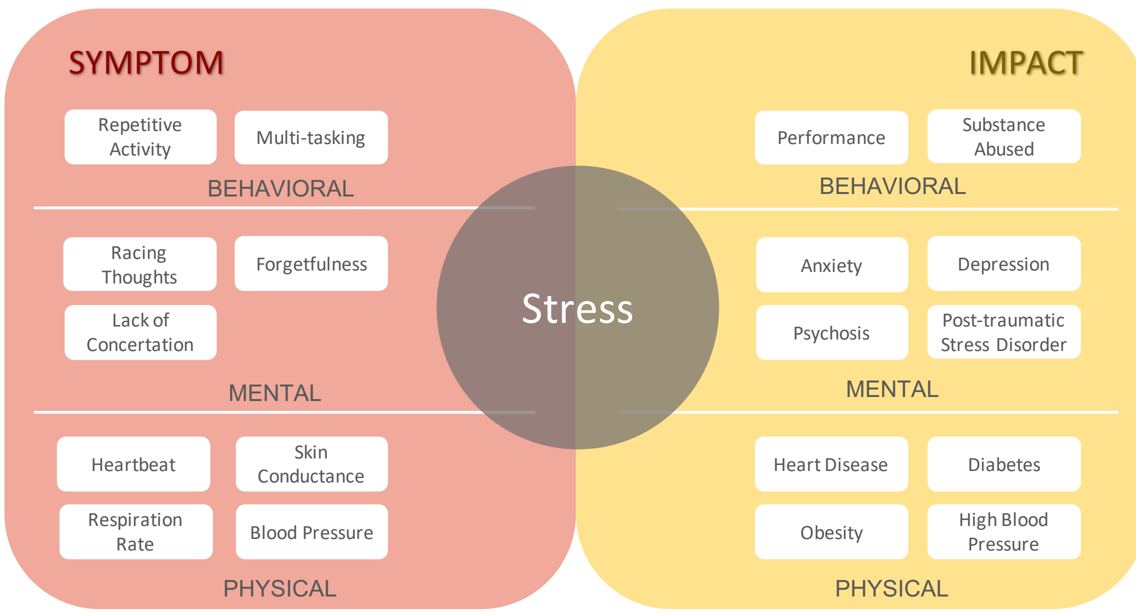


Figure 1.2: An overview of the stress symptoms and impacts with respect to the three aspects.

While the physical symptoms can be measured objectively, the mental symptoms are less intuitive to measure from data collected using mobile devices. Although measuring the mental symptoms are fairly difficult, we can estimate the mental state indirectly from human behavior. According to Wikipedia, human behavior is "the potential and expressed capacity of human individuals or groups to respond to internal and external stimuli throughout their life" [4]. It is considered as a complex interplay of actions, cognition, and emotions [5]. In other words, it is the interaction between how we think, feel, and act. It is a combination of the physical and mental aspects, and therefore providing clues of one's mental state.

As an example, when people suffering from racing thoughts and find it difficult to concentrate as a result of stress, it reflects on how we interact with the computer. Mark et. al. suggested that people tend to switch computer windows more often as a sign of multitasking [90]. It could be a source of distraction that results in poor performance at work. A recent study suggests that repetitive activity, especially ritual-like behavior, is a stress relief that confers a sense of controllability and predictability [35]. One example is where basketball players bounce the ball exactly six times before a free throw. The repeated high precision and concentration performance of the same act reflects the situation is under control of the person, thereby reducing the feeling of stress and anxiety. This kind of stereotyped ritual-like activity emerged naturally as the phenomena are not only observed in humans, but also in other animals. As a result, those behavioral changes can be objectively measured to estimate the internal mental state.

Although repetitive activity characterizes the behavioral stress response [69], it has a larger overlap with common daily activities such as walking, running, and cycling. Solely monitoring the repetitive activities cannot conclude if a person is stressed or not, especially there is limited knowledge on the repetitive activities that are related to stress. In this thesis, we focus on the repetitive daily activities related to health and fitness such as exercises, heartbeat, and respiration rate. Since these activities account for a large fraction of human behavior, that associated with health [102, 71] and also commonly adopted as a mean for stress relief [49, 60, 109]. Beyond the responses, stress can also affect our performance either positively or negatively, or lead to several behavioral disorders such as substance abused.

1.2.2 Research Challenges

Challenge 1: Variation of the activities and stress response

Each person is a unique individual. We distinct from others in many different ways such as different physique, culture, intelligence, and habits. It leads to the variation of health conditions, symptoms, and impacts of stress among different people. Identifying repetitive activities and monitor the stress of humans is therefore very challenging. The same activity can be performed in many different ways, and different activities can be largely diverse. These variations incur great challenges to existing machine learning based approaches, in which the difficulties stem from the design of features and the acquisition of sufficient annotated data. In order to address these issues, existing work provides solutions on an ad hoc basis, where different models were deployed for a different set of activities as well as personalized models for each individual. Applying multiple models on mobile devices is unfortunately not practical as the devices are typically resource-constrained, and are even not possible on the emerging battery-free devices. More importantly, there are cases where the knowledge of the target activities is limited, especially when the goal is to discover behavioral responses to stress. To address the above issues and support the discovery in the future, a generic and efficient method that can identify general repetitive activities is desired.

Challenge 2: Lack of data for human research

Annotated data of human subjects especially of their mental and biological state is critical to many studies related to humans. Collecting human data is however very costly and sometimes impossible. The participants typically have little incentive to contribute their data. On the contrary, they are more worried about the proper use of the data collected. Although existing work proposed to leverage transfer learning to reduce the number of data needed for data-driven models, it has little application in the real-world scenario as the target of the task and the data required varies.

In addition, data collected from human subjects is usually unreliable based on individual subjective evaluations. For example, a questionnaire is one of the most commonly used techniques to estimate a person's mental state. However, it is difficult to pinpoint their feeling exactly on a numbered scale. Some approaches require recalling the participant's memory of the past experience may suffer from inaccurate information as a result of memory loss. There are significantly inadequate data on human subjects in terms of both quantity and quality. To overcome this challenge approaches that leverage domain knowledge in order to compensate for the lack of data are needed.

1.2.3 Research Framework

Given the above overview, the aim of this research is basically two folds: 1) to detect and recognize repetitive activities; 2) to measure and predict stress and its impact. To this end, the following research framework was proposed as shown in figure 1.3. This research focuses on the physical (physiological) and behavioral measures as they can be quantified objectively. Those activities and responses to stress will be captured by mobile devices such as smartphones, smartwatches, wearable sensors, and wireless devices. Data will be collected from various sensors including Photolethysmogram (PPG), Electrodermal Activity (EDA), wireless signal, and accelerometer, which is already available in wearable devices. The collected data will then be processed for data completion, noise filtering, normalization, and standardization. Specifically, physiological stress symptoms such as heart rate variability, skin conductance, and respiration rate will be measured from the obtained sensor data. On the other hand, repetitive activities will be measured from body motion data captured from the accelerometer. Performance can be measured from smartphone and computer usage. Finally, those measures will facilitate stress monitoring by capturing the daily activities, estimating stress, and predicting its impact on cognitive performance.

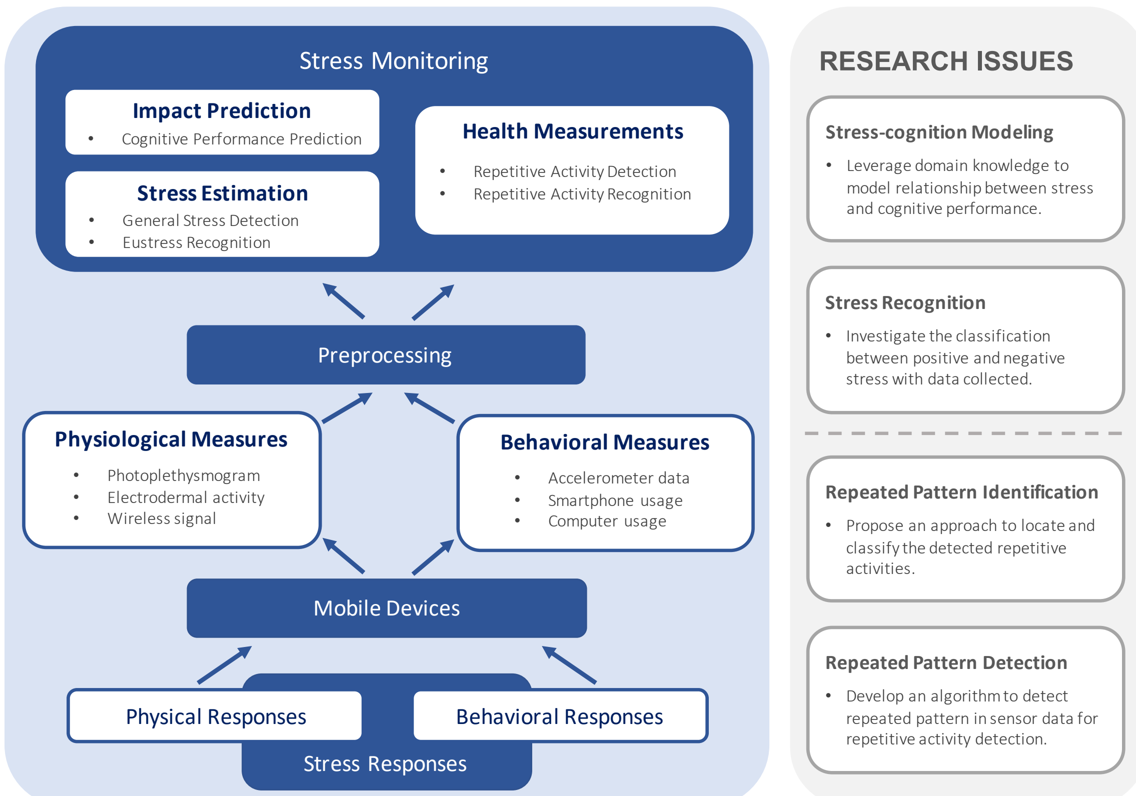


Figure 1.3: The proposed research framework and the corresponding research issues.

The research issues are also listed on the right of the figure. The issues can be simply grouped into two parts. From the bottom, we have repeated pattern detection and identification that aims to provide a method to measure the daily activities that are related to health. As mentioned in section 1.2.2, a generic method is needed to cope with individual differences in repetitive activity. We tackle this challenge in a two-step approach by first solving an easier problem which is to detect the repeated pattern, followed by a more difficult problem that will locate and recognize the repeated pattern. The approaches were applied to detect and recognize daily activities including walking, running, and cycling, that facilitate the measurement of health condition. The upper part of the research issues aims to measure and predict stress and its impact. Stress recognition is different from stress detection and measurement in the literature. Most of the existing work treat stress as general stress that is usually unpleasant. This problem aims to investigate the possibility to distinguish eustress from distress. Then we model the relationship between stress and cognitive performance in a time-aware manner.

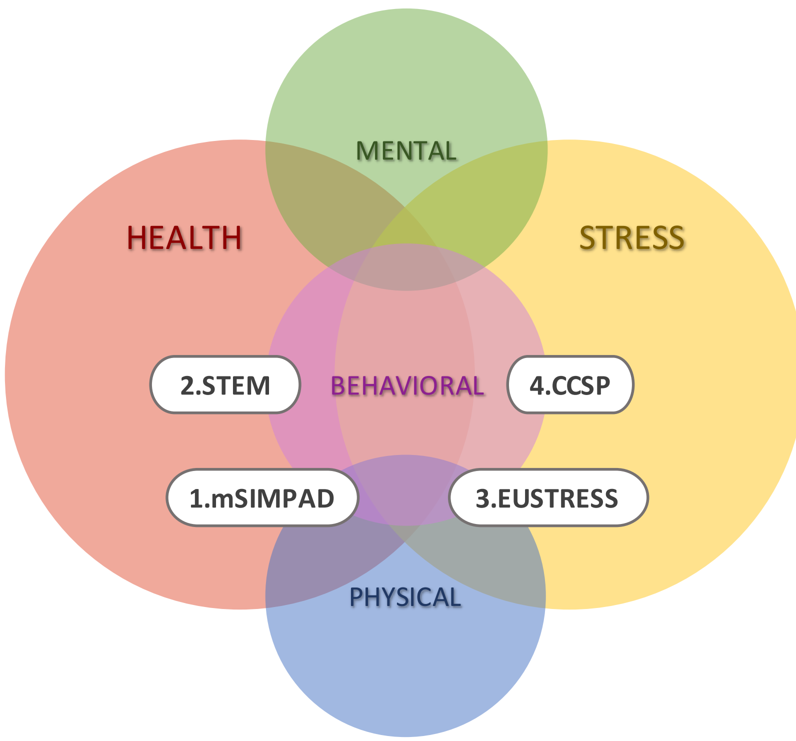


Figure 1.4: The map of our research work in the stress monitoring context.

In figure 1.4, we show the mapping of our research work under the context of stress monitoring based on the focus of each work. The left is related to the measurement of health condition, while the right is related to modeling the impact of stress. The mSIMPAD investigates an efficient and robust algorithm for repeating pattern detection. The algorithm can detect if any real-valued multivariate time series contains any repeating pattern, in which barely any assumption with the pattern such as the shape and interval are unknown. Followed by mSIMPAD, we proposed STEM to locate and recognize the repeating pattern of a time series. Both mSIMPAD and STEM are studied to identify daily activities that are highly repetitive, it can also be applied to many other applications such as respiration monitoring using wireless signals as we will show in chapter 3. The EUSTRESS is a preliminary study that aims to understand the relationship between eustress and stress symptoms. Lastly, the CCSP studied a computational model on the basis of theory about stress and performance to predict cognitive performance by measuring stress symptoms. The reason to leverage domain knowledge for model building is that it provides more accurate prediction using a computational model that requires few annotated data.

1.3 Literature Review

Stress monitoring using mobile devices is an emerging area throughout the last decade. More and more commercial wearable devices started to include stress monitor as one of the basic features like Apple [1], Garmin [3], and Fitbit [2]. It seems a well-studied area but when it comes to research, there are still many open issues that remain unsolved. In view of this, there is an increasing number of studies proposed for stress detection with the aid of mobile devices. Almost every previous work applies machine learning techniques for monitoring stress. Much effort has been devoted to discovering the potential source of data for stress detection. The emphasis is put on the efficiency and robustness of the features extracted from the sensor data.

Henceforth, one natural way to classify the previous work is by the data modality. Each modality measures the stress response in a different part of the mechanism, which can be roughly divided into two categories: biomedical signals and behavioral signals. In this section, we introduce the taxonomy of the literature on the basis of different data modalities and investigate the work being proposed recently to tackle challenges under different application scenarios for stress detection.

1.3.1 Biomedical Signals

Biomedical signals are observations of human body processes which are time-varying measures that are typically obtained from electronic sensors [65]. It is the most widely used measure for stress detection given the solid background knowledge of the biological mechanism. It also enabled objective measures of the subjective internal feeling that makes it the most popular and reliable source for stress detection [124]. There are two types of biosignals, namely physiological signals and physical signals.

Physiological signals are direct measures of the body's vital functions such as cardiac activity, brain function, exocrine activity, and muscle excitability [43]. The common techniques to estimate these body functions including electroencephalography (EEG), electrocardiography (ECG), photoplethysmogram (PPG), blood volume pulse (BVP), electrodermal activity (EDA), electromyography (EMG), etc. Figure 1.5 shows the distribution of the physiological measurement of the human body.

Physical signals are measured of body deformation as the result of muscle activity such as pupil size, blinks, respiration, facial expression, and voice [43]. The physical signals are less directed to the body function and might be influenced by external factors such as environment and human consciousness.

There is no clear boundary applying solely physiological signals or physical signals, instead, most studies combined different measures for a better approximation of

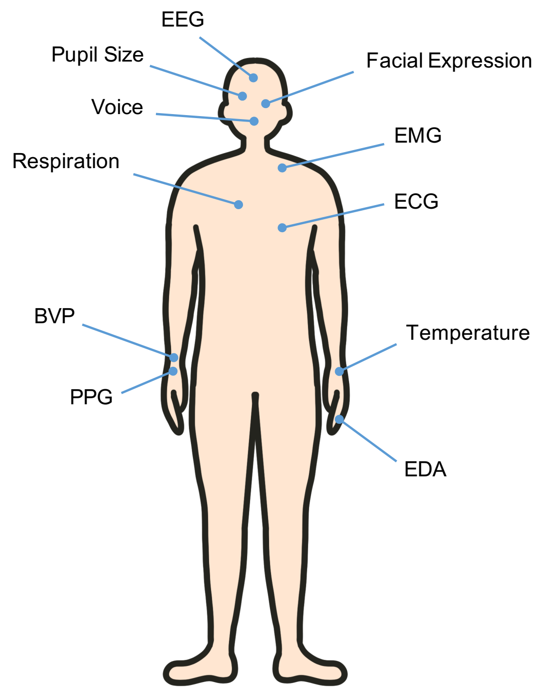


Figure 1.5: The physiological signals that has been considered in the literature.

stress. Healey and Picard [52] are the pioneer in stress detection using biosignals with mobile devices. They developed a sensing platform to capture ECG, EMG, EDA, and respiration measures and showing promising results in stress detection while in real-world driving. The authors in [148] later studied a multi-user system to monitor signals including EDA, BVP, Pupil Diameter, and Skin Temperature for stress detection on computer users. As the biosignal may reveal similar patterns under cognitive load, the authors in [123] illustrated the possibility to differentiate between cognitive load and stress by using a wearable EDA device. The authors also find that non-relative features perform better than relative features, which suggests that calibration procedures are not needed in practical systems. Despite the widely used physiological signal, the physical signals for stress detection were also examined in the literature.

The facial expression has been investigated as a source for stress detection [88, 39, 108]. Face temperature variation is possibly a more privacy-sensitive measure of stress with images [127]. Towards a less invasive measure of stress, authors in [81] proposed a voice-based sensing approach that captures characteristics such as pitch, speaking rate, and vocal energy as features. A study comparing the voice- and EDA-based methods suggested that both measures correlate with the perceived stress, are both effective and less invasive [13].

As more and more studies are published with similar results, it is convinced that the biosignals are efficient and robust features for stress detection [124]. Researchers begin to explore more in-depth issues in stress detection applications. One direction is to introduce non-intrusive sensing approaches for stress detection. In [84], the authors investigated the potential of using a lightweight PPG signal to detect stress and cognitive load. McDuff et. al. [96] investigated an approach to estimate PPG measure at a distance of 3 meters using a digital camera. On the other hand, Sun et. al. [130] proposed an activity-aware stress detection approach given that physical activity could affect biosignal, which should be considered while performing stress detection. cStress was proposed to provide a gold standard for continuous stress measurements with which the experiment was carefully designed and data collected from rigorous lab study [57]. It also considers temporal dependency for the model design.

1.3.2 Behavioral Signal

Compared to the biomedical signal, the literature on behavioral signals is much less studied. While the biomedical signals are more suitable for acute stress detection as the bodily response is more instance; the behavioral signals are more suitable for chronic stress as the changes of behavior are more prominent than the biosignal in the long run.

Several works investigated the possibility of using information collected from mobile devices for stress detection. Behavioral data such as screen on time, call usage, SMS usage, location traces, and co-located Bluetooth devices shows clear modification during stressful situations [41, 116]. Jaques et. al. [63] combined physiological data with behavioral data to predict student's stress and happiness. Similar work has been proposed in [22], that instead of physiological data, it uses information from smartphones, personality traits, and weather data to recognize the stress of a participant. In [37], the authors proposed to leverage smartphone app usage to predict perceived stress at the workplace. Among those smartphone-based approaches, StudentLife [134] initiated a big leap that performed a large-scale study with 48 participants across 10 weeks. It captured physiological, behavioral, and self-reported measures of the students that enabled various applications including stress detection.

Besides the smartphone-based approach, researchers also explore the potential of using social media content to detect the stress of a person [77]. It analyzes the textual information of the content on Twitter to infer the users' psychological stress. There are also studies showing that data collected from the use of a computer mouse can detect if the user is stressed [129]. The authors in [54] proposed to detect stress by measuring the pressure made upon the keyboard and mouse. Mark et. al. studied the relationship between computer use and stress [90]. Stress is associated with time spent on computers and the amount of multitasking, which is evaluated by the number of switches between computer windows. Researchers also investigated using motion data collected from smartphone to detect stress in working environment [40].

1.3.3 Remarks

The related work demonstrated the possibility of stress detection with mobile devices that have been studied extensively. There are effective and robust biosignals features widely used in the literature, and numerous potential measures related to behavior. The literature is clearly moving from laboratory to naturalistic setting, in order to provide a more practical system design for continuous stress measurement.

In contrast to stress detection, predicting the impact of stress using mobile devices is previously unexamined. Although there is work modeling and predicting the change of cognitive performance [12], it is not on the basis of stress and is not in the scope of this study.

1.4 Thesis Organization

The rest of the dissertation is organized as follows. The main body contains two parts: 1) detection and recognition of repetitive activities; 2) measurement and prediction of the impact of stress. The first part included in Chapter 2 and Chapter 3, introduced a generic method to detect and identify repeating patterns in multivariate time series, for the health related repetitive activities. The second part including Chapter 4 - a preliminary study of stress recognition, and Chapter 5 - a computational model to predict cognitive performance over time under stress by measuring the biosignals. The final chapter concludes the whole dissertation.

Chapter 2

mSIMPAD: Efficient and Robust Mining of Successive Similar Patterns of Multiple Lengths in Time Series

A Successive Similar Pattern (SSP) is a series of similar sequences that occur consecutively at non-regular intervals in time series. Mining SSPs could provide valuable information without a priori knowledge, which is crucial in many applications ranging from health monitoring to activity recognition. However, most existing work is computationally expensive, focuses only on periodic patterns occurring in regular time intervals, and is unable to recognize patterns containing multiple periods. Here we investigate a more general problem of finding similar patterns occurring successively, in which the similarity between patterns is measured by the z-normalized Euclidean distance. We propose a linear time, robust method, called *Multiple length Successive sIMilar PAtterns Detector* (mSIMPAD), that mines SSPs of multiple lengths making no assumptions regarding periodicity. We apply our method on the detection of repetitive movement using a wearable inertial measurement unit. The experiments were conducted on three public datasets, two of which contain simple walking and idle data, while the third is more complex and contains multiple activities. mSIMPAD achieved F-Score improvements of 3.2% and 6.5% respectively, over the simple and complex datasets compared to the state-of-the-art walking detector. In addition, mSIMPAD is scalable and applicable to real-time applications since it operates in linear time complexity.

2.1 Introduction

Successive Similar Patterns (SSPs), or series of similar subsequences that occur successively, are prevalent in the physical world in areas such as seasonal weather, biosignals, and human behavior. Mining SSPs in time series means recognizing the appearance of successive recurring patterns, which is a challenging problem of great influence [146]. For example, repetitive physical motions characterize many interesting types of exercise including walking [114, 24], running [50], and free weight training [45, 137], which can be detected as the repeating patterns occurring in wearable sensor data. SSP detection is the enabling technology for exercise tracking using wearable devices that evaluate the activeness of an individual and can provide guidelines for daily activities to promote physical health. Despite exercise tracking, SSP detection can be applied to analyze heartbeat signals from electrocardiography (ECG) by searching unusual patterns within periodical signals for abnormal heartbeat detection [32]. Factory assembly work can be analyzed by estimating the lead time using SSP detected from wearable sensor data [86]. In this work, we focus on the use case of repetitive movement detection - a problem that aims to identify the repeating physical motion of human activities - using wearable inertial measurement units (IMUs) data. Automatic repetitive movement detection is warranted as it is the fundamental building block for human activity recognition. Specifically, representative patterns can be extracted efficiently from long-time series of each of the detected segments to facilitate human behavior studies and wearable healthcare applications [42].

Although numerous methods for repeating patterns finding have been proposed based on periodicity detection, most of them can only handle a single fixed period and fail to detect periodic patterns when disturbances appear or the patterns are misaligned [138, 147]. There are also patterns with multiple periods that may not be present all the time and their re-occurrence may be shifted. For instance, an athlete lifting weights may perform multiple successive repetitions, where the interval between each repetition may vary due to muscle fatigue. Different moves can produce different periods and the shift of each repetition can constitutes asynchronous periodic patterns. Existing methods typically require extensive domain knowledge to determine and learn many parameters [137, 112] and make assumptions to the target patterns such as the fixed periodicity [114, 45]. Therefore, it is desirable to have an SSP detection method which is parameter-light, and robust to unknown patterns with variations. Such general approach can reduce the effort devoted to scenario based repetitive movement detection as it requires barely any domain knowledge.

SSP detection is related to, yet different from periodic pattern mining and periodicity detection. In periodic pattern mining, the focus is on finding the pattern in symbol sequences that is fully or partially matched with other occurrences of the pattern [139]. Although Yang et. al. [138] introduced an efficient method for asynchronous periodic patterns mining, it is not a straightforward process to convert real-valued time series to symbol sequences when prior knowledge is missing [78]. On the other hand, periodicity detection focuses on estimating the period of the recurring patterns [18]. The key difference of SSP is the relaxation of the periodicity assumption. It does not postulate a regular interval among the successive patterns, which is more flexible in covering the general set of repeating patterns in reality.

Mining multiple length, successive similar patterns is not trivial due to the following reasons. Unlike periodic patterns, the concept of SSP is ambiguous and difficult to define without prior knowledge of the target pattern. A high computational cost is associated with this problem due to the relaxation of the periodicity assumption. Searching for variable interval patterns between each repetition is intractable even for a small number of repetitions. Additionally, a time series may contain multiple lengths of repeating patterns which makes it difficult to determine which length a pattern belongs to.

Here, we outline a novel, efficient and robust matrix profile [144] based algorithm that finds SSPs with multiple lengths in multi-dimensional real-valued time series. We first introduce a definition of SSP based on the concept of the Range-constrained Matrix Profile (RCMP) and proposed the Range-Constrained Multi-dimensional Scalable Time series Ordered Matrix Profile (RC-mSTOMP) to compute the RCMP efficiently. We then present the *Successive sIMilar PAtterns Detector* (SIMPAD) on the basis of the RCMP which requires two inputs: the target pattern length l and the maximum displacement of pattern m. SIMPAD has barely any periodicity constraints, and the result can be computed and updated in an online fashion efficiently. We extended the proposed method and introduced the *Multiple length Successive sIMilar PAtterns Detector* (mSIMPAD) for finding SSP with multiple lengths within a time series. It provides an estimation of the pattern length and potentially assists in applications such as representation learning for pattern recognition.

Given their ubiquity and availability in smartphones and wearable devices [67, 126, 30], IMUs are the dominantly used source of data for physical activity assessment. It is a well-established area for evaluating the performance of the proposed method in real-world applications. Experiments were conducted on three public datasets and we achieved promising results compared to the state-of-the-art (SOTA) repeating pattern based walking detectors. It shows that the performance of the proposed method is insensitive to the input parameters. We also examined the robustness of our method on low-quality sensor data, to evaluate its suitability to the emergence of battery-free, low sampling frequency, power consumption optimized wearable devices. Additionally, we examined the empirical computational cost and demonstrated the linear relationships to the length and number of dimensions of the input time series. The code used in this study is freely available to all researchers and can be found at: https://github.com/chuntungli/mSIMPAD. We summarize our contributions here:

- We formally defined an SSP, which makes no assumption regarding the periodicity of the target pattern. On this basis, we introduced the RCMP, a modification of the Matrix Profile that is more efficient and superior in the case of SSP detection.

- We proposed the SIMPAD, a general SSP detection method based on the RCMP that is robust, efficient, and parameter-light, which can facilitate various healthcare applications including exercise tracking and heartbeat monitoring. This method is then extended to capture SSPs with multiple lengths, which we call mSIMPAD.

- We evaluate the performance of SIMPAD and mSIMPAD on three public datasets both empirically and by examining their computational costs. The experiments demonstrated the superior performance of the proposed methods over the SOTA repeating pattern based walking detectors.

- We provide guidelines for parameter settings by investigating the effect of different values. We also examined the influence of low-quality input data and the result affirmed that the proposed methods have practical value in handling battery-free wearable devices.

The rest of this chapter is organized as follows. Section 2.2 summarizes the recent findings on repetitive activity detection and motif discovery. Section 2.3 introduces the preliminaries of the proposed algorithm. In section 2.4, we introduced the RCMP and mSIMPAD in detail. Section 2.5 presents the experimental evaluation on three public datasets. Section 2.6 discusses the potential problems and applications of the mSIMPAD. Finally, Section 2.7 covers the conclusion and future direction of this work.

2.2 Related Work

Our work is closely related to the detection of periodicity in time series, which is an active field of research in the data mining community that has been studied extensively. AutoCorrelation Function (ACF) based methods and Fast Fourier Transform (FFT) based methods are the two major approaches to date for periodicity detection in time series [104, 147, 44, 132, 133]. ACF computes the correlation of a sequence to a previous sequence candidate with varying lags and the period is determined by the lag that maximizes the ACF. FFT converts a sequence from the time domain to the frequency domain and determines the period as the frequency that has the maximum power. Generally, the two methods have the same $O(n \log n)$ computational cost and the major drawback of these methods is that they assume the pattern has the same periodicity. It fails when the periodicity varies overtime and the result is sensitive to the frequency that is being estimated.

In human activity recognition, ACF and FFT based methods have been widely used to detect activities that are composed of repetitive movements. Rai *et al.* [114] proposed a normalized-autocorrelation based approach to identify the repetitive pattern of walking from IMU data. Brajdic & Harle [24] conducted a comprehensive evaluation of walk detection comparing the supervised and unsupervised methods including ACF and FFT based approaches. They show that all of the studied methods achieve comparable results. Physical exercises also consist of a set of repetitions of the same movement, which make them a detection candidate here. Guo *et al.* [45] extracted the magnitudes of the IMU data and computed the ACF to identify and count each repetition per set using data collected from wearable mobile devices. These approaches are especially robust when the period is known a priori. However, the period is usually not known and may vary overtime in many real-world applications.

Xie *et al.* [137] decomposed a movement (*complex-activities*) into a series of small-range movements (*meta-activities*) and used the sequence of meta-activities to recognize a complex-activity. They collect angular information during physical exercise and apply Dynamic Time Warping (DTW) to identify meta-activities to overcome this issue. Maekawa *et al.* [86] is similar to this work, where it also identifies similar patterns in the repetition to evaluate assembly work in a factory. It identifies the motif within the IMU time series of each repetition and uses the interval of motif to estimate the lead time of the operation process.

Motif discovery has been extensively studied, but a breakthrough was made recently by Yeh *et al.* who proposed an efficient algorithm, namely namely the STAMP to compute the matrix profile [144]. Several extensions have been made in the following years for multi-dimensional time series as well as towards the improvement of the efficiency of the algorithm [142, 143]. Mirmomeni *et al.* [99] proposed to leverage the matrix profile for mining SSP by examining the number of nearest neighbor arch crossings at each sample of the time series; however, this method will fail when a similar pattern appears in a faraway region of the series. Gharghabi et. al. [42] introduced a temporal constraint to exclude arches from undesired regions in a different context for time series segmentation, but these methods inherit the same computational costs of the matrix profile that requires $O(n^2)$ time. To overcome these issues, we propose a general SSP detection based on the matrix profile with a time constraint during computation that can remove information from undesired regions, while being efficient enough for real-time applications.

2.3 Preliminary

2.3.1 Successive Similar Patterns Mining

In this chapter, we investigate successive similar patterns mining from sensory data. Since periodicity detection has been studied extensively where different fields define it in different ways. We unite these definitions by starting with the definitions of the useful notations. A time series T is a sequence of real valued numbers, and a subsequence $T_{i,l}$ of T is a continuous subset of the values from T of length l starting from position i. Formally, $T_{i,l} = [t_i, ..., t_{i+l-1}]$. The distance between two subsequences $dist(T_{i,l}, T_{j,l})$ is measured by the z-normalized Euclidean distance:

$$dist(T_{i,l}, T_{j,l}) = \sqrt{\sum_{p=1}^{l} (\frac{t_{i+p-1} - \mu_{i,l}}{\sigma_{i,l}} - \frac{t_{j+p-1} - \mu_{j,l}}{\sigma_{j,l}})^2} \tag{2.1}$$

It is the root squared difference of the z-normalized values of two subsequences, where $\mu_{i,l}$ is the mean of $T_{i,l}$ and $\sigma_{i,l}$ is the standard deviation. This can be simplified as follow:

$$dist(T_{i,l}, T_{j,l}) = \sqrt{2l(1 - \frac{QT_{i,j} - l\mu_i\mu_j}{l\sigma_i\sigma_j})} \tag{2.2}$$

where $QT_{i,j}$ is the dot product of the two subsequences. A *successive similar pattern* is a subsequence $T_{i,l}$ of T where a *similar* subsequence $T_{j,l}$ appears within a *nearby* *range*. We then define similar as a small distance between the two subsequences, and a nearby range refers to a small displacement between two subsequences. Formally, a subsequence $T_{i,l}$ is a successive similar pattern iff $\exists T_{j,l} : dist(T_{i,l}, T_{j,l}) < \alpha$, for $i \neq j$ and $|i - j| < m$ where $\alpha \in \mathbb{R}$ is some threshold in which $\alpha \geq 0$ and $m \in \mathbb{Z}$ is a user-defined window length. Then we define the problem as:

Problem 1 (Successive Similar Patterns Mining) *Given a multi-dimensional data series T, target subsequence length l and the searching range m. We want to identify the subsequences that contain successive similar pattern in T.*

2.3.2 Matrix Profile

Before we introduce the proposed method, a brief introduction to the Matrix profile (MP) is provided as a background. This is a method recently proposed by Yeh *et. al.* [144] for all-pair-similarity-search across a time series. The MP is defined as a vector $MP = [mp_1, ..., mp_{n-l+1}]$ that stores the minimum distance of the subsequence to its nearest neighbor for every subsequence in T. The pair of subsequences that has the minimum distance, namely the motif pair can be easily identified from the valley of the MP. The matrix profile was developed for uni-dimensional time series, and it has been recently extended to process multi-dimensional time series. We suggest interested readers refer to [143].

Efficient algorithms have been proposed to compute the matrix profile including the STOMP [142] and STAMP [144]. The former iterates the time series in sequential order, making it more efficient; while the later is an anytime algorithm that iterates the time series in random order to produce approximated result at any iteration. Theoretically, the computational cost of STAMP is $O(n^2 \log n)$, which was later superseded by SCRIMP++ [149], and both STOMP and SCRIMP++ are $O(n^2)$, where n is the length of the time series T.

We are now ready to introduce our proposed methods given the above definitions. In the next session, we will introduce the SIMPAD to efficiently solve the SSP detection problem. Then, we will present the mSIMPAD, an extension for multiple-length SSP detection.

2.4 Methodology

The detection method has two parts, first we introduce the *Successive sIMilar Patterns Detector* (*SIMPAD*) to identify the segments that potentially contain successive similar patterns namely the set of "valleys" from the Range-Constrained Matrix Profile. Second, we choose a combination of valleys that maximize the likelihood of the segments being repetitive using a maximum weighted independent set algorithm. For simplicity, repeating patterns and successive similar patterns (SSPs) being used interchangeably in the rest of this chapter.

2.4.1 Range-Constrained Matrix Profile

The original matrix profile calculates the distances between every subsequence to the rest of the time series and only preserves the distances and their corresponding indices for its nearest neighbor. However, such an approach allows the nearest neighbor to be located anywhere in the time series, which might not be of our interest as the SSP should appear in a period that is considered "short". Figure 2.1 shows a case where an abnormal heartbeat due to ventricular contractions can hardly be identified by the regular matrix profile because of the coincident matching, but is fairly notable in the RCMP. If similar ventricular contractions appeared multiple times over the entire ECG recording, it may identify the contractions as SSP with the regular matrix profile by accident. Therefore, we introduce the Range-Constrained Matrix Profile (RCMP) where the nearest neighbor is calculated only within a given range. For ease of presentation, we refer to MP as the vector of RCMP in the rest of this chapter.

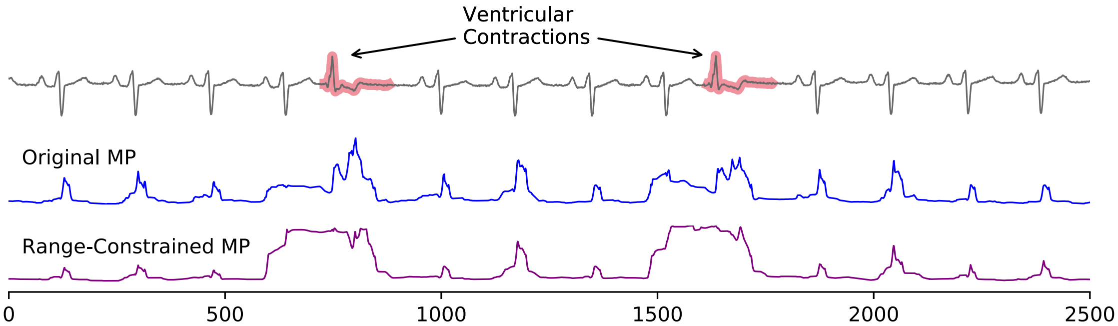


Figure 2.1: Top: a snippet of ECG data in the Beth Israel Deaconess Medical Centre (BIDMC) PPG and Respiration dataset [16]. Middle: the original MP computed from the ECG signal. Bottom: the range-constrained MP that clearly indicate the two ventricular contractions.

The idea of the RCMP is to find the nearest neighbor only within the searching range (length of the search window) m. Instead of calculating the entire distance profile, we calculate a range-constrained distance profile $DP_{i,m}$ for every subsequence $T_{i,l}$ in T. $DP_{i,m}$ is an intermediate vector to store the distances of subsequence $T_{i,l}$ to other subsequences from $T_{i-m,l}$ to $T_{i+m,l}$. Then the minimum value in $DP_{i,m}$ is selected to update the MP. We modified the mSTOMP algorithm in [143] introducing the RC-mSTOMP which produces the MP in $O(nm)$ time. This modification not only ensures a local similarity search but also significantly improves the efficiency of computing the RCMP. The details of this algorithm can be found in algorithm 1.

Let $T_{i,l}$ and $T_{j,l}$ be the motif pair which are two subsequences that have the lowest mutual distance between each other. One may imagine that a subsequence repeated once is a motif pair that forms two valleys in the MP at the location of the pair. However, the distance between $T_{i+1,l}$ and $T_{j+1,l}$ should also be small since most of the distances between the subsequences overlapped with the motif pair. Therefore,

Algorithm 1: Range Constrained Multi-Dimensional Matrix Profile (RC-mSTOMP).

Input: d-dimension time series $\boldsymbol{T}$, int l, int m
Output: MP

double[] MP; int[] IP;
double[] $QT \leftarrow$ slidingDotProduct($\boldsymbol{T}_{1,l}, \boldsymbol{T}_{1,m}$);
int $s \leftarrow$ sum($\boldsymbol{T}_{1,l}$); int $ss \leftarrow$ squaredSum($\boldsymbol{T}_{1,l}$);
int $sr \leftarrow 2m + 1$; int $dv \leftarrow \boldsymbol{t}_1$; int $nv \leftarrow \boldsymbol{t}_{m+1}$;
for $i \leftarrow 1$ *to* $|\boldsymbol{T}|$ **do**
  if $i > 1$ **then**
    $QT \leftarrow QT - dv \times \boldsymbol{T}_{i-m,sr} + nv \times \boldsymbol{T}_{i-m+l,sr}$;
    $s \leftarrow s - dv + nv$;
    $ss \leftarrow ss - dv^2 + nv^2$;
  $DP_{i,m} \leftarrow$ calcDistProfile($QT, \boldsymbol{T}_{i,l}, \boldsymbol{T}_{i-m,sr}, s, ss$);
  $DP_{i,m} \leftarrow$ columnWiseAscendingSort($DP_{i,m}$)
  $DP'_{i,m} \leftarrow$ double$[d, sr] = \{0,...,0\}$
  for $k \leftarrow 1$ *to* d **do**
    $DP'_{i,m} \leftarrow DP'_{i,m} + DP_{i,m}[k, :]$
    $DP''_{i,m} \leftarrow DP'_{i,m} \div k$
    $MP[k, :] \leftarrow$ elementWiseMin($MP[k, :], DP''_{i,m}$)
  end
  $dv \leftarrow \boldsymbol{t}_{i-m}$; $nv \leftarrow \boldsymbol{t}_{i+m+1}$;
end

instead of having two (or the number of repetitions) separated valleys, the MP covering SSP should be a flat valley.

With this observation, we can identify SSPs by finding the valleys in the MP. The details of the SIMPAD can be found in algorithm 2. We first compute the MP providing the time series T and the target subsequence length l as input. Then the key of this algorithm is to decide a suitable threshold to distinguish repeating and non-repeating components. This is a difficult task as the distance of the MP correlates to the d and l. We assume the input series T is a composition of repetitive subsequences and non-repetitive subsequences, so that the MP is either at a distance of SSP or non-SSP segments. Then we apply Otsu's method [106] which is a popular

Algorithm 2: Successive sIMilar PAtterns Detector (SIMPAD)

Input: d-dimensional time series T, int l
Output: RP

$MP \leftarrow$ RC-mSTOMP(T, l);
double $\alpha \leftarrow$ otsu_thresh(MP);
$RP \leftarrow$ boolean$[|MP'|] = [0,...,0]$;
$counter \leftarrow 0$;
for $i \leftarrow t$ *to* $|MP|$ **do**
  if $MP[i] \leq \alpha$ **then**
    if $counter \geq l$ **then**
      $RP[i - l : i] \leftarrow 1$;
    else
      $counter \leftarrow counter + 1$;
    end
  else
    $counter \leftarrow 0$;
  end
end

binarization method in the field of image processing to determine the threshold α. Those subsequences with distance below α are *valid* as the valleys shown in figure 2.2. Finally, to avoid false positives caused by chance, we only accept subsequences that are valid for at least l consecutive time steps. The result is stored in a boolean vector RP of length $l - m + 1$, which indicates if the corresponding subsequence $T_{i,l}$ in T contains a repeated pattern or not.

Note that we could replace the RCMP by the regular matrix profile and perform the same detection pipeline for SSP identification. However, it will generate an MP that includes the nearest neighbor from anywhere of the entire time series and potentially degrades the detection performance. To overcome this issue, we might adopt the windowing approach by letting the window size equals to the search range m while computing the regular matrix profile for each window. We then obtain the full MP by concatenating the matrix profiles of each window to perform SSP detection. This can ensure the range constraint but the windows are assumed to be independent, which results in a loss of information coherency of the pattern as a whole. Instead, the RCMP incorporates the range constraint in the computation, which preserves information coherency and reduces the computational cost. We include the detection results of SIMPAD and mSIMPAD with a regular matrix profile and those with a sliding-window-based regular matrix profile in Section 2.5.3.

2.4.2 Multiple-Length Successive Similar Patterns Detection

In the last section, we have introduced the SIMPAD that can be applied for fixed-length SSP detection. It assumes all the repeated patterns have the same length, so it may fail when patterns with different lengths appear within one time series. To tackle this problem, we present the *Multiple-length Successive sIMilar PAtterns Detector* (*mSIMPAD*) to capture repeated patterns of different lengths automatically. The basic intuition is that for each potential pattern length, we compute the RCMP accordingly and identify the valleys as the candidates of repeated patterns. We then choose a set of valleys that best fits the patterns.

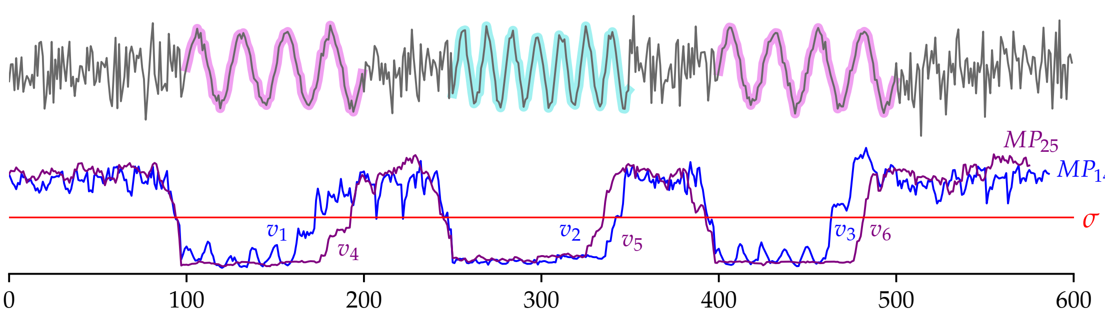


Figure 2.2: An artificial signal contains 3 regions of sine waves with two intervals: From 100 to 200 and 400 to 500, the interval is 25; From 250 to 350, the interval is 14. The bottom shows the MP with length 25 in purple, and 14 in blue.

Let $\hat{l}$ be the *true length* of a repeated pattern that equals to the interval length. The quality of a fit is then defined by the distance between $\hat{l}$ to the detected subsequence length l. In reality, $\hat{l}$ is usually not known and may vary overtime. We assume that *a valley with larger sum of depth is a better fit to the repeated pattern.* The rationale behind is that a larger value in the valley implies a clearer differentiation between repeated pattern and non-repeated patterns. Figure 2.2 shows an example where the area of valleys is larger when l is closer to $\hat{l}$. Finding a better fit to a repeated pattern is similar to searching for a valley with a larger sum.

To better illustrate the problem, we further introduce the notation of MPs and valleys with different target lengths. MP_l is the MP of T with subsequence length l. A valley $V_l \in \mathbb{R}^{|V_l|}$ is a sequence of differences between the subsequence $D \in \mathbb{R}^{|V_l|}$ of MP_l with some real value threshold α such that all the values of V_l are less than α. Formally, $V_l = [\alpha - d_i | d_i < \alpha, \forall i \in [1, 2, ..., |V_l|]]$, where $d_i \in D$.

Identifying patterns with multiple periodicities in T maps to finding the best fit of valleys from multiple length MPs. Given that we have computed all the MPs for different l, there are two key issues when choosing the set of valleys that best fit the patterns. The first issue is that the scale of the distance depends on l. From equation 2.1 we notice that longer subsequences tend to have larger distances, and therefore the difference between α and d_i could be larger. That being said, the distance incurs a strong bias to pairs of subsequences with larger l. To mitigate this effect, we obtain the length normalized distance by factorizing the MP by $sqrt(1/l)$ which is known to be better than simply factorized by l [79]. With the length normalized distance, we can compare the similarity of subsequence pairs with different length as well as the corresponding valleys.

The second issue is that at any point in T, there should be at most one valley chosen as the best fit of a given pattern. While one valley may overlap with other valleys with different l, choosing one valley will reject the others. See figure 2.2 as an example, choosing v_1 will reject v_4, and vice versa. This could become a lot more complicated when $|L|$ is large, and one valley may be overlapped with multiple other valleys either from one length or different lengths. The longest valley is always chosen if we simply find valleys that yield the largest sum. To overcome this issue, we introduce the assumption that *the sum of all selected valleys is maximized when all of the best fits of repeated patterns are found.* Instead of choosing those longest valleys, the overall objective is to find a set of valleys that maximize the total sum. It allows the method to choose several shorter valleys over one long valley if the total sum is larger. Formally, the subproblem is defined as:

Problem 2 (Multiple Length Successive Similar Patterns Mining) *Let* $V = \{V_1, V_2, ..., V_{|V|}\}$ *be the set of valleys found in* MP *at all candidates* l*, and* $idx(V_l) = [x_1, x_2, ..., x_{|V_l|}]$ *be the function mapping the valley to the corresponding index in* T*. The objective is to find the subset* $V_{opt} \subseteq V$ *such that the total sum of valley* $\sum_{V_i \in V_{opt}} \sum_{j=1}^{|V_i|} v_j$ *is maximized where* $idx(V_i) \bigcap idx(V_j) = \emptyset, \forall V_i, V_j \in V_{opt}$ *and* $i \neq j$*.*

This problem is related to the Maximum Weighted Independent Set (MWIS) problem [107], which is an NP hard problem to find a subset of weighted vertices in a graph such that there exists no edge between any pair of the selected vertices while the sum of the weights is maximized. We can generate the graph $G = \{V', E\}$ for vertices $V' = \{v'_1, v'_2, .., v'_{|V|}\}$ as the sum of valleys V as $v'_i = \sum_{j=1}^{|V_i|} v_j$. Then we can generate the set of edges E if two valleys are overlapped, formally $E = \{(v'_i, v'_j) | idx(V_i) \bigcap idx(V_j) \neq \emptyset\}$. The objective is then defined as:

$$\begin{gathered} \max_{V_{opt} \subseteq V'} \sum_{v_i \in V_{opt}} v_i \\ s.t. \forall u, v \in V_{opt} : (u, v) \notin E \end{gathered} \tag{2.3}$$

Notice that the graph generated is sparse with numerous components, since the valleys come from different parts of the time series and are separated by regions that have no repeated patterns. In order to find the solution more efficiently, we can divide the problem into multiple subproblems by the components in G while having the same optimal solution. This can drastically reduce the search space to speed up the computation. Then we apply the branch and bound approach in [107] for each of the subgraphs and finally combine the result to obtain the optimal solution.

Algorithm 3: Multiple-length Successive sIMilar Patterns Detector (mSIMPAD)

Input: d-dimension Time Series $\boldsymbol{T}$, int[] L, int[] M
Output: RP
1 $RP = [False, ..., False]$;
`// Find valleys from` MP `at different` l
2 **for** $l, m \leftarrow L, M$ **do**
3     $MP_l \leftarrow$ RC-mSTOMP(T, l, m);
4     $\alpha \leftarrow$ otsu(MP_l);
5     $V[l] \leftarrow$ findValleys(MP_l, α);
6 **end**
`// Find best match from` V
7 $G \leftarrow$ generateGraphs(V);
8 **for** $G' \leftarrow G$ **do**
9     $V_{opt} \leftarrow$ MWIS(G');
10 **end**
11 $RP[idx(V_{opt})] \leftarrow True$;
12 return RP;

The details of the mSIMPAD can be found in algorithm 3. First, we compute the MP for each potential pattern length and searching range and determine α using Otsu's method to identify the valleys from lines 2 to 6. Note that it is possible to let L be $[2, 3, ..., n/3]$ and M be $2 \times L$ if domain knowledge is missing. We generate graph G and separate it into multiple subgraphs in line 7. Line 8 to line 10 outline the loop for each subgraph to find the MWIS that is the best fit of valleys and it is stored in V_{opt}. Finally, the repeated patterns are annotated as the indexes of selected valleys correspond to T.

2.5 Experimental Evaluation

The experiment aims to answer the following questions: 1) How do different thresholds α, subsequence lengths l and search ranges m affect the detection accuracy? 2) Does the proposed algorithm achieve a comparable result to the SOTA walking detectors? 3) Can it detect different forms of repetitive movement? 4) How does it scale to different sizes of input? We first introduce the metric for performance evaluation.

2.5.1 Performance Metric

For a time series T of length n, we identify if the subsequence $T_{i,l}$ that contains repetitive movement, where $i \in [1, ..., n-l+1]$ and l is the window size of the search. This process produces $n-l+1$ detection result denoted as $RP = [rp_1, rp_2, ..., rp_{n-l+1}]$:

$$rp_i = \begin{cases} 1, & \text{if } t_{i,l} \text{ contains SSP} \\ 0, & \text{otherwise} \end{cases}$$

The ground truth of each trace contains a repetitive segment indicated by t_{start} and t_{end}, and we derive the truth label of each subsequence $T_{i,l}$ as 1 if $t_{start} \geq i \geq t_{end}$, and 0 otherwise. Then we define accuracy (ACC), false positive rate (FPR), false negative rate (FNR), and error rate (ERR) as follow:

$$\begin{cases} \text{ACC} = \frac{TP+TN}{|RP|} \\ \text{FPR} = \frac{FP}{FP+TN} \\ \text{FNR} = \frac{FN}{FN+TP} \\ \text{ERR} = \frac{FP+FN}{|RP|} \end{cases}$$

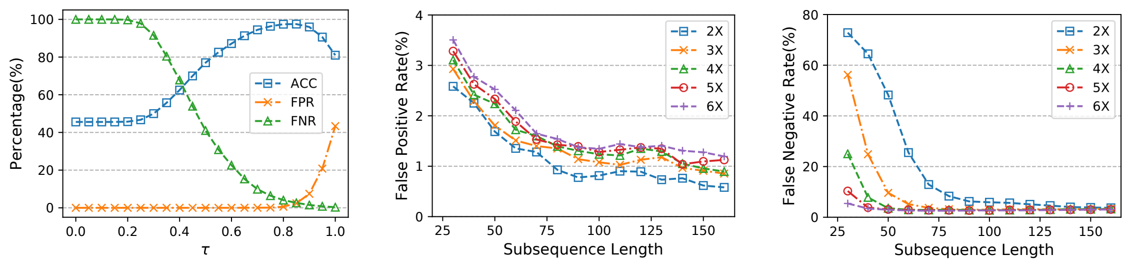
Label \ Prediction	1	0
1	TP	FN
0	FP	TN

2.5.2 Parameter Estimation

First, we study the effect of the parameters of SIMPAD on a dataset [24] collected from 27 participants using a conventional smartphone with an embedded accelerometer sampled at 100Hz. The participants were told to walk at different speeds with different placement of the smartphone - i.e. carry by hand, in a pocket, in a backpack or in a handbag. Then the ground truth was obtained by manually labeling each of the traces from the camera recording, which has the indicated start and end times when participants were walking. The previous study in [24] shows that both supervised and unsupervised approach can accurately detect walking segments in which the median error rate is less than 2%.

The parameters were estimated by evaluating the performance on walk detection using the ground truth provided. We start by examining the effect of the threshold α. It is assumed that no repeating patterns appear in the first e subsequences, so that the first e distances were used to obtain the baseline distance d_{base} for non-repetitive subsequences as $d_{base} = \sum_{i=0}^{e} mp_i$. We let $e = 100$ as the earliest walking session begins at 837. We let $0 \leq \tau \leq 1$ be a user defined ratio to obtain threshold α manually by multiplying the baseline distance such that $\alpha = \tau \times d_{base}$. A range of values $[0, 0.05, 0.1, ..., 0.95, 1.0]$ for τ were examined over all the traces, where we fix the other parameters for $l = 100$ (≈1 sec) which is about to cover a *stride* (two steps) as the average steps per second is about 2 [114]. Let the searching range m be 3 times the subsequence length ($3l$). We excluded the first e subsequences in the evaluation as we are using these distances to determine α.

Figure 2.3a shows the performance of different values of τ, where we found that lower τ results in higher FNR, and higher τ results in higher FPR. It suggests that the distance of the RCMP can effectively differentiate between repetitive and non-



(a) (b) (c)

Figure 2.3: (a) effect on accuracy (ACC), false positive rate (FPR) and false negative rate (FNR) of different τ; (b) and (c) effect on false positive rate and false negative rate of different l and m.

repetitive subsequences. The value of τ was fixed as 0.85 for the following evaluation on different subsequence lengths l and searching ranges m.

Ideally, the subsequence length should be exactly the same as the length of the repeating pattern such that it matches the next cycle. However, repetitive movements in reality vary in every repetition which makes the value difficult to determine. Similarly, the searching range should cover somewhere around $i + l$, as the next cycle should begin right after the current cycle. Unfortunately, the lag between each cycle varies and the shape might be deformed. Therefore, a larger search range provides a better chance to find a more similar nearby cycle, therefore lowering the distances.

We demonstrated the relations between different subsequence lengths and searching ranges in figure 2.3b and 2.3c, in which the performance is calculated from the mean of the detection result over all traces. It shows that FPR is insensitive to l, though relatively higher FPR occurred when l is smaller. This is due to the smaller l providing fewer points to compare in a subsequence so that it is more likely to mismatch the non-repetitive subsequences randomly. On the contrary, FNR seems to be very sensitive to l as it increases drastically when l is small. The reason is that for a repetitive movement with a normal cycle of length $\hat{l}$, 100 in this case, if $l \ll \hat{l}$ (i.e. $l < \hat{l}/2$) then the subsequence contains only a small portion of the complete cycle. While the searching range is small, it cannot cover any portion of the next cycle with the given window. Therefore, the FNR is significantly higher for $l \leq 50$ and $m \leq 3 \times l$ given that the average length of a walking cycle is about 100. Fortunately, the larger m can tolerate the negative effect of a small l better, since it provides a better chance for the small portion of a cycle to be matched with the next cycle given a large window. But it comes with a minor drawback that the FPR is slightly increased.

As discussed earlier, the dataset in [24] is a relatively simple, and contains only idle and walking data. The authors achieved a median of total error of less than 2% with the best parameters. Unfortunately, we were unable to reproduce the result with the reported parameters, so that we compare the performance mentioned in [24]. The SIMPAD achieved comparable result as the median of total error rate is 1.78% using the parameters ($l = 100, m = 6 \times l, \tau = 0.85, e = 100$).

2.5.3 Repetitive Movement Detection

We first evaluate if the proposed methods achieve comparable results to the SOTA walking detectors on the HAPT [15] dataset. Then we study if the mSIMPAD is generic enough for the general repetitive movement detection by evaluating the performance on PAMAP2 [115] that contains various activities.

Evaluation of Robustness

The HAPT is a dataset collected from 30 volunteers using accelerometer and gyroscope on smartphone at a 50Hz sampling rate. It contains different forms of activities including: walking, climbing up or down stairs, sitting, standing, laying, and transitions between activities. We use the precision $= TP/(TP + FP)$, recall

$= TP/(TP + FN)$, and F-Score $= 2 * (p \times r)/(p + r)$ for the evaluation.

We compare the proposed methods to the widely used walk detection algorithms, namely the Normalized Autocorrelation based Step Counting (NASC) [114] and the short-term Fourier transform (STFT) [18] correspond to the ACF based method and the FFT based method. These algorithms were the best performing algorithms as noted in [24]. NASC excluded segments of the time series where the standard deviation was below a threshold σ_{thresh} over a window std_{win}, then it performs normalized auto-correlation over a window of 2 seconds with a range of time span τ_{min} to τ_{max} for those remaining subsequences. Those subsequences are then asserted to be walking if the maximum of the normalized auto-correlation exceeded another threshold R_{thresh}. STFT is a Fourier transform based method that calculates the frequency domain energy of the vertical acceleration signal with consecutive windows of size dft_{win}, and affirms walking if the total energy of the interested frequencies exceeds threshold dft_{thresh}.

Algorithm	**Parameter Value**
NASC	$std_{win} = 40, \sigma_{thresh} = 0.24, R_{thresh} = 0.4, \tau_{min} = 40, \tau_{max} = 100$
STFT	$dft_{win} = 60, dft_{thresh} = 0.25, freq_{min} = 0.01Hz, freq_{max} = 7Hz$
SIMPAD	$l = 50, m = 5 \times l$
mSIMPAD	$L = [40, 50, 60], M = 5 \times l \in L$

Table 2.1: List of the parameter values used of each algorithm for HAPT dataset.

The raw linear accelerations obtained from accelerometer were used for SIMPAD, mSIMPAD and NASC. Since the STFT takes the vertical velocity as input, we apply the Madgwick algorithm [85] in order to estimate the orientation of the smartphone using the gyroscope signal, and then transform the linear acceleration to the coordinate with respect to the earth.

The parameters of the SOTA methods were then selected using a brute force search approach in order to provide the upper bound of performance for each of the methods. The selected values are reported in table 2.1. For the SIMPAD, we are only required to provide the parameters of l and m. We choose the length to be 1 second ($\approx$50 samples) as discussed earlier. From the previous experiment, we notice that 5 times the subsequence length can tolerate with the mismatch between l and l_{true}well. For the mSIMPAD, we choose $L = [40, 50, 60]$ to cover the variations of walking speeds and the same scale for the searching range as $M = 5 \times L$.

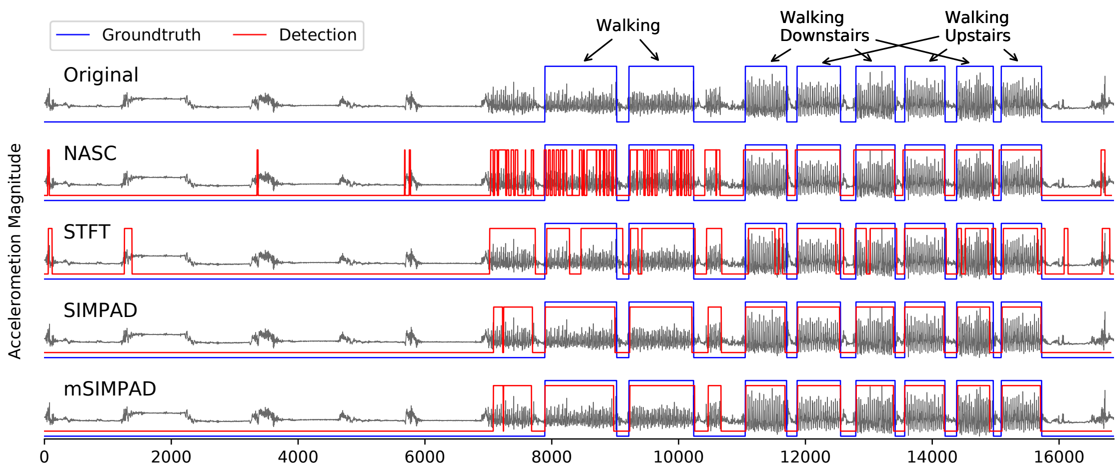


Figure 2.4: Detection result on one of the traces with different algorithms. The top indicate the groundtruth of the trace with blue line, and the detection results are indicated with red line in the lower figures.

An example of a detection result of each algorithm is shown in figure 2.4. The red lines indicate the detection result generated by the methods where 0 is non-repetitive, otherwise it is repetitive. The ground truth were indicated by blue lines. The graph shows that NASC suffers from a higher false negative rate and breaks one walking period into several segments due to the difficulty of defining a global threshold. STFT has a more continuous detection period, but is not sensitive enough to cover the walking period in place that results in higher false positive rate. On the contrary, SIMPAD and mSIMPAD reveal better performance and cover the entire walking period and fit the walking period better compared to the other algorithms.

From the above example, we observed that the data contains two subsequences that have significantly higher acceleration magnitude for every trace (between samples 7050-7900 and 10250-11060 in the above example). We further investigated all the traces and found that there exist repeating patterns that have not been reported in the annotations provided in [15]. It might be due to the relocation of the experiment as it happens when the task changes from lying to walking, and from walking to climbing downstairs. Therefore, we excluded the two suspicious zones of data from the last transition to the first walking part, and from the last walking to the first climbing downstairs part for a more precise evaluation.

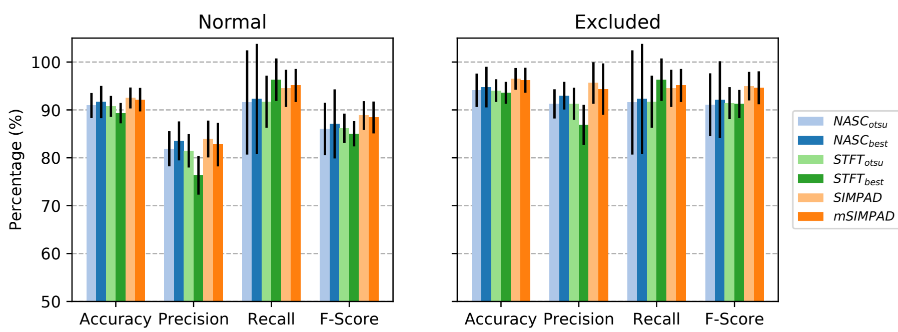


Figure 2.5: Performance on HAPT dataset.

The overall result is reported in figure 2.5 including the normal evaluation where the entire dataset was used, and the excluded evaluation where the suspicious zone has been removed. To provide a fair comparison, we modified NASC and STFT to automatically determine the threshold using Otsu's method as we use in our proposed approach, and selected one second as the window size for those methods. We notice that the modified NASC and STFT achieved a comparable result to the best parameters obtained from exhaustive search. Automatic threshold determination may lead to variation tolerance between different time series, to the point where its incorporation into STFT even outperforms the best parameters. While the NASC is a two-step thresholding approach, we fixed r_{thresh} as 0.4 and determine std_{thresh} by using Otsu's method, which yields similar results to the best parameters. Both SIMPAD and mSIMPAD show improvements by 2.8 and 2.5% respectively, with F-Scores roughly equal to 95%. Also, the high precision and recall of the proposed methods illustrate that the RCMP is a robust indicator for differentiating repeating and non-repeating patterns.

Evaluation on Generality

This section aims to evaluate the ability of the proposed method on general successive similar patterns detection. The evaluation is conducted using the PAMAP2 [115], which is collected from 9 subjects wearing 3 IMUs sampled at 100Hz frequency while performing 18 different activities. The various types of activities aim to provide a range of different repeating frequencies for generality evaluation. We classify the following activities as repetitive: walking, running, cycling, Nordic walking, ascending stairs, descending stairs, and rope jumping. The rest are considered as non-repetitive activities. We leverage the IMU data on the subject's ankle for activity detection. We down-sample the data to 50Hz and the same transformation method mentioned in the previous section was applied for vertical acceleration estimation.

We compare the results to the modified NASC and STFT with the same parameters that were used in the previous section, since it is difficult to define a global threshold for various activities. For the mSIMPAD, we choose $L = 40, 70, 100$ accordingly to the 0.8, 1.4 and 2 seconds in order to capture repeating patterns with different lengths. The lengths were selected as a reference to τ_{min} and τ_{max} where NASC searches within this time lag range.

The overall results are reported in table 2.2 in which the best values are in bold. The

Dataset	Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
HAPT	NASC	94.10 ± 3.49	91.26 ± 3.05	91.56 ± 10.88	91.07 ± 6.57
	STFT	94.03 ± 2.37	91.30 ± 3.35	91.73 ± 5.44	91.41 ± 3.34
	SIMPAD	**96.44 ± 2.26**	**95.63 ± 4.33**	94.50 ± 3.89	**94.96 ± 2.98**
	mSIMPAD	96.16 ± 2.60	94.35 ± 5.36	**95.10 ± 3.45**	94.62 ± 3.44
PAMAP2	NASC	81.70 ± 12.32	**99.39 ± 0.85**	66.47 ± 21.95	77.12 ± 22.26
	STFT	78.79 ± 9.12	99.31 ± 0.90	62.45 ± 6.75	76.50 ± 4.76
	SIMPAD	84.11 ± 5.59	99.12 ± 1.14	71.24 ± 5.66	82.78 ± 3.74
	mSIMPAD	**84.62 ± 5.65**	98.28 ± 1.78	**72.90 ± 5.58**	**83.59 ± 3.62**

Table 2.2: Performance on HAPT dataset where the values given as mean ± SD.

proposed methods outperform NASC and STFT by 5.7% in F-Score. It shows that the RCMP works well with different types of repetitive movements as well as different lengths of repeating pattern. The similar results of SIMPAD and mSIMPAD suggest that if the target patterns have similar lengths, SIMPAD can capture most of the repeating patterns with ease. The performance of mSIMPAD still higher than the SIMPAD as it can find better match within the patterns of different lengths. We expect the improvement would be much larger when the variability of the pattern lengths is huge.

Note that both SIMPAD and mSIMPAD are matrix profile based methods, in which the input can be replaced by the original matrix profile. However, the regular MP has a higher probability of finding the nearest neighbor which is not a repeating pattern just by chance without the range constraint. We can see that from its performance on the HAPT dataset where the F-Scores are 92.15% and 91.23% for the SIMPAD and mSIMPAD respectively. We notice a significantly lower precision (87.78% and 85.89%) which complies with the inference when the regular matrix profile is used. Alternatively, we could perform windowing on the time series by letting m be the window size to satisfy the range constraint. However, the sliding window fails to capture coherent information, which results in poor performance compared to the RCMP. The SIMPAD and mSIMPAD with sliding-window-based regular matrix profile achieved F-Scores 92.27% and 92.86% respectively on HAPT, 79.96% and 81.28% on PAMAP2, which shows that both are worse than the proposed RCMP-based approach. The RCMP satisfied the range constraint while preserving the coherent information and greatly reduced the computational cost, which is a more suitable solution for the problem at hand.

Robustness on Low-Quality Data

Battery-free wearable devices rely only on harvested kinetic energy from the user, which has emerged as an alternative to power sensor nodes [128]. The wireless communication and sensing units consume much more power than a typical microcontroller, so the transmission and sampling rate of such devices is reduced to optimize power consumption [59]. We investigate the influence of low-quality accelerometer data by downsampling the traces in the HAPT dataset to 20Hz and adjusted the parameters l and m by the downsampling ratio. As the performance of SIMPAD and mSIMPAD are similar in the HAPT dataset, we report the mSIMPAD result for simplicity. The performance of mSIMPAD recorded a decrease by 2.11% to 92.51% while NASC and STFT also decreased by 2.2% and 0.12% to 88.87% and 91.29% respectively. It shows that the existing approaches perform fairly well on sensor data having a low sampling frequency, as most human activities are lower than 10Hz.

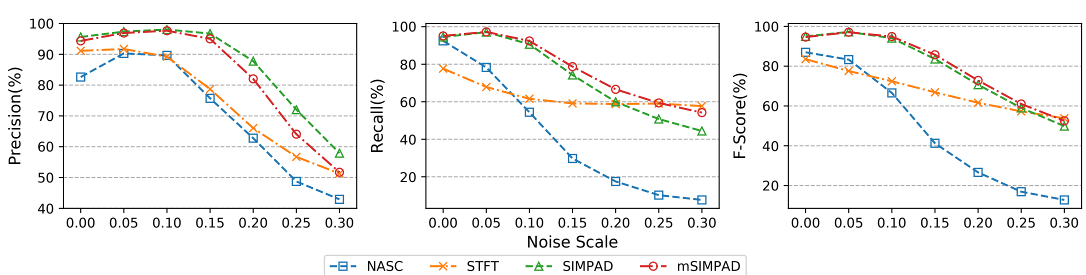


Figure 2.6: Effect of sensor noise to performance.

We also study how sensor quality influences performance by adding Gaussian noise to the traces. We first normalize the traces to the scale of 0 to 1, then inserted random noise of a normal distribution having 0 mean and scale as a standard deviation between 0 to 0.3. Surprisingly, the performance of mSIMPAD improves slightly when a small level of noise is inserted into the signal (scale=0.05) as shown in figure 2.6. The reason is that segments of relatively idle data, could be identified as similar patterns (e.g. a slightly upward trend or slight downward trend), which actually should not be detected at all. The added noise increases the distance between these drifts, and can help distinguish such data up to a certain noise level (10% of the maximum sensor value in this case), which suggests a future direction towards improving the performance of mSIMPAD by simply adding random noise. The performance of STFT is slightly higher than the proposed approach under very serious noise where the noise scale is 0.3 - that is 30% of the maximum value of the data which is rather unrealistic. In general, the proposed approach is more robust than the existing methods, while the STFT potentially performs better in the situation of extremely low-quality data.

2.5.4 Comparison of Execution Times

The above evaluation demonstrated that mSIMPAD is a robust method for repetitive movement detection even on datasets that contains multiple activities. In this section, the empirical computational cost to obtain the MP is examined. Theoretically, the time complexity of RC-mSTOMP is $O(nm)$ in which m can be neglected as $m \ll n$, while mSTOMP is $O(n^2)$ and both ACF and FFT are $O(n \log n)$, where n is the sequence length, and m is the searching range (note that both algorithms increases linearly with respect to the number of dimensions). The time complexity of RC-mSTOMP is significantly lower than the other methods since it scales linearly to the sequence length. We evaluate this property by comparing the RC-mSTOMP to its parent algorithms (mSTOMP and SCRIMP++), and also examine the effect on the execution time with different parameters. All the experiments were performed on a conventional PC with Intel Core(TM) i7-8850H CPU @ 2.60GHz x 12 and 16GB RAM. The default values of the parameters are as follow when not specified: $n = 2^{14}$, $l = 100$, $m = 200$, and $d = 1$.

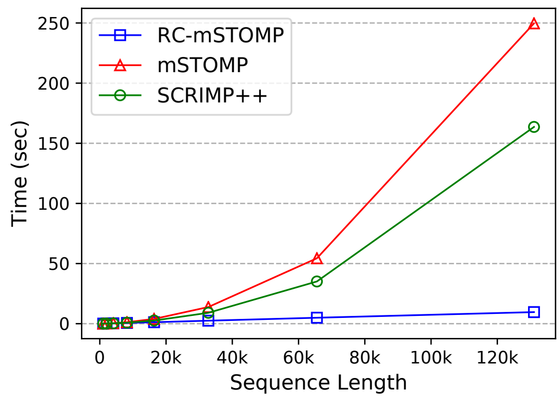


Figure 2.7: A comparison on execution time of different sequence lengths.

First, we examined the computational cost on different sequence length empirically to compare RC-mSTOMP to its parent algorithms mSTOMP and SCRIMP++. It is not intended to conclude that the proposed algorithms are superior than its parent algorithms as the goal of these algorithms are different. Instead, we aim to demonstrate that the proposed method inherits the important properties of the parent algorithms while scaling linearly with respect to sequence length, so that it can support real-time applications. In this evaluation, different lengths of sinusoidal signals were generated as the input sequences. The resulting execution times are shown in figure 2.7. They coincided with our expectations where RC-mSTOMP produces the lowest execution time among all the other algorithms and scales linearly as we will show in the following. mSTOMP and SCRIMP++ are roughly scaling at $O(n^2)$ but still very scalable to large time series. The SCRIMP++ completes slightly faster than mSTOMP that might due to the sinusoidal data in this particular case.

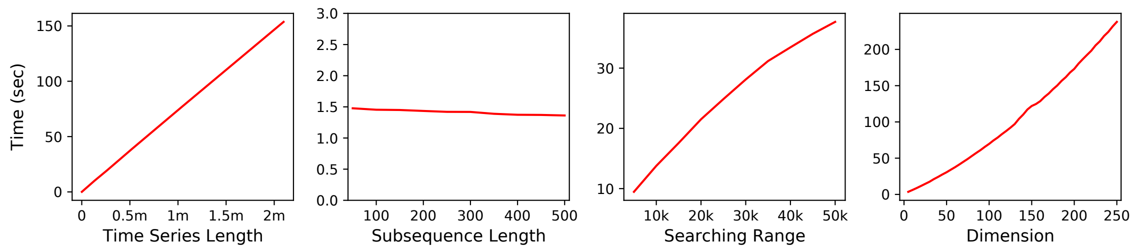


Figure 2.8: An evaluation on execution time of different parameters.

Then, we examined the effect on execution time over different parameters: n, l, m, and d. The resulting execution times are shown in figure 2.8. It shows that the proposed algorithm inherits the same property as discussed in [143] on subsequence lengths and dimensionality. Subsequence length has no effect on execution time when m is fixed, and it follows a linear relationship to dimensionality. Then we examined the effect of the length n of the input sequence by fixing the other parameters as default values, and execute RC-mSTOMP on the time series with increasing length. As expected, we found that the execution time increases linearly with respect to sequence length. Finally, we evaluate the effect of the length of the searching range m on execution time. We found that for small n, the effect of different size of m is negligible. Therefore, we increase n to 2^{16} and $l = 500$. The result at the right of the figure 2.8 shows that it also follow a linearithmic relationship with respect to m.

In this section, we demonstrated that RC-mSTOMP is capable of supporting real-time applications as the execution time has no effect on l and linearly correlates to n, m, and d. In the next section, we will discuss the potential problem and application of this work.

2.6 Discussion

We proposed the method for multiple length successive similar patterns mining using the Matrix Profile, and we evaluated the proposed methods in the use case of repetitive movement detection on three public datasets. Result shows that the proposed method is efficient and robust for general repeated pattern mining without prior knowledge of the pattern, except the expected lengths of the target patterns. In this section, we discuss the potential problems as well as the applications of the proposed algorithm.

The underlying technique of the proposed method is related to matching the patterns of a time series on its own. It seems to coincide with ACF and FFT that are the commonly used techniques in the previous work, but there are two major differences that explained the superiority of the proposed method. Firstly, the ACF and FFT have more restricted constraints on the time span so that the repeated patterns occur with regular intervals, while our proposed method is capable of detecting repeated patterns with a variable time span since we are searching for a local motif within a given range m. Secondly, efficient algorithms for calculating the ACF and FFT are $O(n \log n)$ while our method achieved $O(nm)$. To the best of our knowledge, it is by far the fastest, deterministic and exact algorithm for successive similar patterns detection.

The key limitation of the proposed method is to determine the parameters. The pattern length l and the displacement m could be designated based on the sampling frequency of the sensor data. We suggest larger l and m would be more favorable to highly repeating patterns such as walking as shown in section 2.5.2. Also, experiment shows that SIMPAD can work well even if l is quite different from the actual pattern length. In addition, the threshold σ is determined by Otsu's method that assumed the time series is a composition of repeated and non-repeated components. If such assumption is not met, choosing a global threshold would be an option. It is not difficult as the distance is normalized both by the signal and the length of the pattern.

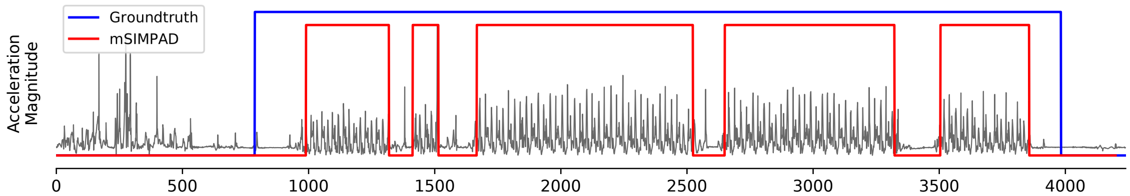


Figure 2.9: An example of rope jumping data in PAMAP2 [115]. The magnitude of the acceleration signal shows that several pause exists during the activity. The blue line indicates the ground truth of the repetitive movements, and the red line indicates the detection result of mSIMPAD: 0 as non-repeating; and > 0 as repeating.

The experiments might not show an enormous improvement of our method compared to the SOTA as reported in 2.5.3. However, considering the modest number of parameters to set, and its robustness to novel situations as well as to poor quality sensor data, the proposed method has great potential as a general approach that is devoid the effort devoted to studying specific scenarios. Also, we investigated the detection results over different series and found two key reasons that degrade the detection performance. First, the dataset is generated mainly for activity recognition purposes, so the quality of the ground truth labels is rough. There are offsets on almost every repeated activity that result in many false negatives that actually are non-repeated components. Also, the repeated activities are not always continuous nor contiguous, but the ground truth labeling annotated the entire segment as one activity. For instance, participants may fail while rope jumping. An example can be found in figure 2.9 where the participant has stopped several times during rope jumping. Those regions constitute idle data as no body motion is captured whereas the ground truth data was not handled to that level of detail.

Second, we only used the raw accelerometer data as input which is a noisy signal. Various techniques would be applied to obtain quality data including filtering and sensor fusion in order to improve the detection performance. As an example, we included the gyroscope data as input to the mSIMPAD, and the average F-Score has increased from 83.6 to 87.5 for the PAMAP2 dataset. However, we aim to compare the result to the SOTA walking detectors within a consistent setting. We therefore apply the same configuration over the three datasets in order to demonstrate the proposed methods are superior than the SOTA in general, but not by manipulating the data input nor the parameters of the algorithm. In addition, our method can better extract those repeated activities in place as shown in the example in figure 2.4 and figure 2.9. It suggests that the proposed method is desired, and it can be applied as a subroutine of human activity recognition and analysis.

The potential applications of the proposed algorithm is two-fold. For non-periodical time series, it can identify where repeated patterns occurred, especially for asynchronous periodic patterns and slowly changing patterns. Applications such as exercise tracking, which rely on repeated pattern detection can utilize the proposed method for better performance. On the other hand, it can identify abnormalities for periodical time series such as electrocardiograms of heartbeats as shown in figure 2.1. The intuition is that the MP computed from periodical data is the composition of regular patterns and abnormal patterns. For the regular patterns, the distance of MP should be close to 0 where the abnormal patterns are the discords from the MP. It can also be applied as a subroutine of other data mining tasks such as activity recognition, activity segmentation, and routine discovery. For instance, RCMP can efficiently identify those segments containing repetitive movements, and machine learning techniques can then be applied only to those segments in order to eliminate unnecessary computation.

2.7 Conclusion

A Successive Similar Pattern (SSP) is a key feature of many kinds of interesting data. The detection of these repeating patterns without prior knowledge comes with significant challenges. In this study, we proposed the mSIMPAD, an efficient algorithm for multiple length SSP detection based on the matrix profile. We formally defined SSP based on the distance to the nearby subsequences and introduced the Range-constrained Matrix Profile - a modification of the original matrix profile - to compute the distances efficiently. Then, the SIMPAD was proposed and we further extended it to handle multiple length successive similar patterns automatically, namely the mSIMPAD. We studied the repetitive movement detection problem as a use case, and we conducted experiments on three public datasets to evaluate the proposed method in terms of robustness and time efficiency. The experimental evaluation shows that the mSIMPAD achieved on par (or even better) results compared to the state-of-the-art repeating pattern based walking detectors on all three public datasets. Finally, we discussed the potential problems and applications arising from the proposed method. In the future, we plan to extract the template of the SSPs using the RCMP, and apply it to activity recognition and abnormality detection.

Chapter 3

Repetitive Activity Monitoring Using Multivariate Time Series: A Generic and Efficient Approach

Repetitive activities like breathing and walking account for a large fraction of human behavior. Repetitive activity monitoring aims to measure and distinguish between different repeating activities using multivariate time series data collected from IoT devices. It not only plays a vital role in understanding human behavior but also enables numerous applications ranging from healthcare monitoring to manufacturing management. Most existing approaches process multivariate time series on a sliding window basis. However, these approaches are mostly scenario dependent, computationally expensive, and require extensive domain knowledge. Moreover, real-world repetitive activities may have varying time intervals between them, which invalidate existing sliding window methods. In this chapter, we propose STEM, a Scalable Template Extraction Method for scenario independent monitoring of repetitive activities with varying intervals. Instead of using sliding windows, we detect and locate the appearance of repeating patterns based on the Matrix Profile. Distributional features are then extracted from the identified patterns such that domain knowledge can be avoided. The model is validated on both three public datasets and a synthetic dataset. The results demonstrate the proposed method can eliminate around 95% of the computation on undesired subsequences while achieving a recognition improvement of over 20% on the synthetic dataset and 4% on the public datasets. It also shows superior performance on a use case of respiration rate estimation using wireless signals.

3.1 Introduction

Repetitive activities are building blocks of our daily lives including fundamental bodily functions like respiration and heartbeat, and common human activities like exercise and assembly line manufacturing. The capacity of accurately monitoring repetitive activities could enable numerous applications ranging from healthcare monitoring [24, 45] to manufacturing management [86]. For example, premature ventricular contractions as a risk factor to many heart diseases could be detected via monitoring the repeating heartbeats with commodity wearable devices [74]. Therefore, there are growing interests of repetitive activity monitoring which refers to detecting whether successively repeated physical motions exist and identifying the category of those motions. The recent advancement in IoT technologies allows efficient collection of sensor data, providing a great opportunity for continuous monitoring of repetitive activities. The data collected typically appear as multivariate time series data. The focus of this chapter is to achieve efficient and effective repetitive activity monitoring with multivariate time series data.

Due to the significance of repetitive activity monitoring, various approaches have been proposed. However, those approaches have three main limitations. First, they have a strong assumption that repetitive activities contain fixed periodicity which is unrealistic in many real-world scenarios. For instance, people who suffer from sleep apnea stop and start breathing repeatedly at an irregular rate. At the gym, the time intervals between consecutive movements may change with the levels of energy. The presence of irregular intervals of the repetitive activity invalidates the existing approaches. Second, existing approaches are mostly sliding-window-based methods. It typically requires considerable domain knowledge to determine the size and step of the window as well as to design the features, which remains a major challenge in general activity recognition [75]. Third, the sliding window methods are too computationally demanding for IoT devices with limited resources. It divides the time series into segments of equal length and extracts features from every segment for further activity recognition with supervised learning models. The feature extraction and recognition pipeline performed on every window induce a prohibitive computational cost.

In this chapter, we focus on monitoring repetitive activities using multivariate time series data derived from IoT devices. The main idea is based on the Successive Similar Pattern (SSP) [74] - the recurring pattern with irregular intervals - generated by the repeating physical motions within the time series. However, the vision of applying SSP for repetitive activity monitoring entails the following challenges.

- The SSPs could have variable lengths, shapes, and intervals making it difficult to identify the segments where SSPs occur.
- The start and end positions are ambiguous for recurrent patterns. As a result, identifying each SSP from the time series incur a high computational cost that becomes intractable even for small data.
- The SSPs may have various lengths and could be misaligned, making it difficult to compare the distances between different SSPs.

To address those challenges, we firstly proposed an algorithm called *R-mSIMPAD* to detect time series segments containing SSP and estimate the length of the pattern which is more robust and has fewer assumptions than the existing method [74]. Next, we introduced the concept of the *templates* that is the underlying "true" patterns being repeated with variations, in order to formally define a set of SSPs in the same segment without any assumption regarding the pattern. On this basis, we proposed a *Scalable Template Extraction Method* (STEM) to identify and extract the set of SSPs from the detected segment. Finally, we examined two approaches to classify the detected segments. On one hand, we investigated an elastic-measurement-based method to compute the pairwise distance between the extracted template and apply the Nearest Neighbor algorithm for the classification. On the other, we combined STEM with the Empirical Cumulative Distribution Function (ECDF) to extract distributional features from the segments to mitigate the computational cost incurred by the elastic measure.

We conducted extensive experimental evaluations on both public datasets and a synthetic dataset. The results suggest that our approach can efficiently reduce 95% unnecessary computation by ignoring time series that do not contain any repeating patterns. Also, the extracted template can effectively distinguish among different SSPs achieving an improvement of up to 23% on synthetic data. The features extracted by the combination of STEM and ECDF show superior performance on the public datasets recorded a 3.3% improvement on average. We found that STEM can better identify the patterns being repeated and exclude those patterns with abnormalities and variations that lead to a better result. Finally, we study a use case of respiration monitoring based on wireless signals. Without any modification, STEM easily achieved at least 3 times better performance compared to the baseline method. The use case illustrated that the proposed method has great potential as a general method for many other applications.

The main contributions of this work are as follows:

- We investigated the problem of repetitive activity monitoring and proposed STEM, an efficient and effective method to identify and recognize SSPs from multivariate time series.
- We performed extensive evaluations on both public datasets and synthetic data to validate the performance of the proposed method and achieving on par or even superior performance.
- We demonstrated that the proposed approach is a general method that can be applied to many applications as we illustrated on the two use cases of repetitive activity recognition and respiration rate monitoring.

The rest of this chapter is organized as follows. Section 3.2 summarize the related literature. Section 3.3 presents the detailed design and rationale of the method. Section 3.4 illustrate the result of the experiments. Section 3.5 discussed some issues that might be uncleared. Finally, we conclude this article in Section 3.6.

3.2 Related Work

Repetitive activity monitoring is closely related to activity recognition, which aims to classify different activities using the collected data. Although the previous work does not explicitly target repetitive activities, most of them are studied and evaluated mainly on repetitive activities [27, 82, 45, 141, 101]. Some studies included recognition of non-repetitive activities but the performance there usually suffers due to the complex nature of the activity [51], making it out of the scope of our work here.

A large body of literature adopted time series classification techniques for activity recognition [11], in which unrealistic assumptions were made. They assumed that the start and end positions of a pattern can be accurately identified and that the lengths are equal for patterns of the same class [58]. Therefore, considerable work adopted the sliding window approach combined with machine learning models to perform activity recognition given its simplicity and robustness. One of the key contributions of the prior work focuses on extracting distinctive features from the data. Popular statistical features such as [68] and distributional features [47] achieved promising results on many activity recognition tasks even compared to state-of-the-art deep-learning-based approaches [51].

There is also existing work that focuses on a particular set of highly repetitive activities. Xia et al. [110] proposed an unsupervised method to recognize assembly work in a factory by finding the motif in the sensor data. [101, 45, 137] investigated the recognition of different gym exercises that are highly repetitive, and count the repetitions of each exercise for performance evaluation. The auto-correlation function that computes the self-similarity at different lags, is the most commonly used approach for repetition counting. The major drawback is that auto-correlation cannot handle repeating patterns with irregular intervals as shown recently in [74]. Although repetitive activity recognition has been widely studied, the existing approaches are either scenario dependant and require extensive domain knowledge to determine many parameter settings or make unrealistic assumptions that are not practical for real-world applications. We aim to propose a general method for repetitive activity monitoring that has barely any pattern assumptions regarding shape and periodicity and is efficient and robust to novel situations.

3.3 Methodology

In this section, we first introduce the notations and definitions essential to understanding the problem. Then we provide the general problem statement of repetitive activity monitoring. We then present the general idea and rationale of the proposed method and give examples.

3.3.1 Definitions

A multivariate time series $\mathcal{T}$ is a sequence of d-dimensional real-valued numbers. A subsequence $\mathcal{T}_{i,l}$ of $\mathcal{T}$ is a continuous subset of the values from $\mathcal{T}$ of length l starting from position i. Formally, $\mathcal{T}_{i,l} = [\mathbf{T}_i, ..., \mathbf{T}_{i+l-1}]$, where $\mathbf{T}_i$ is a d-dimensional vector. A Successive Similar Pattern (SSP) is a more general definition of a repeating pattern, which is a subsequence that occurs consecutively at non-regular intervals in time series [74]. It is defined as a subsequence $\mathcal{T}_{i,l}$ of $\mathcal{T}$ where a *similar* subsequence $\mathcal{T}_{j,l}$ appears within a *nearby range*. The range is a user-defined constraint of the displacement of the SSP and the similarity is defined by the z-normalized Euclidean distance as:

$$D(\mathcal{T}_{i,l}, \mathcal{T}_{j,l}) = \sqrt{\sum_{p=0}^{l}\sum_{k=0}^{d}(\frac{t_{i+p}^{(k)} - \mu_{i,l}^{(k)}}{\sigma_{i,l}^{(k)}} - \frac{t_{j+p}^{(k)} - \mu_{j,l}^{(k)}}{\sigma_{j,l}^{(k)}})^2} \tag{3.1}$$

where $t_i^{(k)}$ is the value of $\mathbf{T}_i$ at k-th dimension, $\mu_{i,l}^{(k)}$ and $\sigma_{i,l}^{(k)}$ are the mean and standard deviation of $[\mathbf{T}_i^{(k)}, \ldots, \mathbf{T}_{i+l}^{(k)}]$. The pair of $\mathcal{T}_{i,l}$ and $\mathcal{T}_{j,l}$ is considered in the same class if the above condition is satisfied. A segment is a subsequence $\hat{\mathcal{T}}$ of $\mathcal{T}$ contains either none or exactly one class of SSPs. Each segment belongs to either one class in the set of all possible classes $Y = [y_1, \ldots, y_m]$.

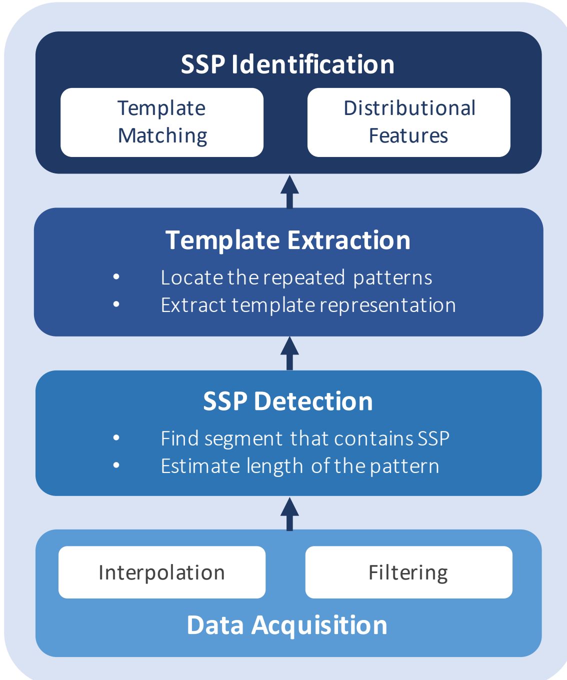


Figure 3.1: Overall framework of the Repetitive Activity Recognition System.

3.3.2 Problem Statement

Repetitive activity monitoring aims to classify and measure different repetitions of the same physical motion using data collected from IoT devices. In this chapter, we focus on the identification and classification of SSPs in multivariate time series. Since the SSPs within a segment is ill-defined, we introduce the concept of templates to help formulate the problem.

We assume there is a *template*, a d-dimensional sequence $\mathcal{T}_l$ of length l that is being repeated with variations at non-regular interval within a segment $\hat{\mathcal{T}}$. Given a multivariate time series $\mathcal{T}$, our objective is to find a set of non-overlapping subsequences $\mathbb{S} = \{\mathcal{T}_{i,l}\}$ as SSPs that minimize $\sum_{\mathcal{T}_{i,l} \in \mathbb{S}} D(\mathcal{T}_l, \mathcal{T}_{i,l})$, and predict $\mathbb{S}$ as $y' \in Y$ that minimize the error between y' and y.

3.3.3 Method Overview

The proposed method has four major components as shown in Figure 3.1. Firstly, data is collected with IoT devices and preprocessing is performed on the acquired data. The preprocessing is simply an interpolation of missing data and low-pass filtering with 20Hz as the cutoff frequency, that can be computed in most of the light-weight IoT devices. Then the SSP detection method will find subsequences that contain SSPs and estimates the pattern length. SSP template extraction will then finds the patterns that are being repeated only on the detected segments and therefore significantly reduced unnecessary computation. Finally, SSP identification can be achieved by either matching the extracted template or combining it with existing distributional features.

3.3.4 Successive Similar Pattern Extraction

SSP Detection

Mining SSPs is computationally expensive as it covers a more general set of repeating patterns without assuming a fixed periodicity of the recurring interval. In this regard, mSIMPAD has been proposed recently for mining SSPs of multiple lengths in time series [74]. It scales linearly to the size of the input series and is robust to novel situations as well as to poor quality data. This method is developed based on the Matrix Profile [144], a method for all-pair-similarity-search across a time series. It modified the original matrix profile by introducing a temporal constraint, namely a Range-constrained Matrix Profile (RCMP). The intuition is that the distances between SSPs are comparatively lower than those of non-SSPs. A set of SSP candidates can then be identified from the RCMP as *valleys*, which is a continuous segment that has a lower distance to some threshold. With the SSP candidates obtained from different target lengths, mSIMPAD chooses a non-overlapping subset of candidates that maximize the sum of depths of the selected valleys as the SSP. This detects SSPs and provides a rough estimation of the pattern length.

The key limitation of mSIMPAD is that it assumes the input series must contain both repeating and non-repeating subsequences, such that it can apply the Otsu method [106] to determine the threshold θ. However, this might not be the case in some scenarios where data contains only repeating segments. Also, the detected segments might simply be rejected or separated by unexpected spikes due to random noise. To overcome these problems, we further improved mSIMPAD by learning the threshold θ and introducing the method for merging time series segments to avoid spikes or false rejections.

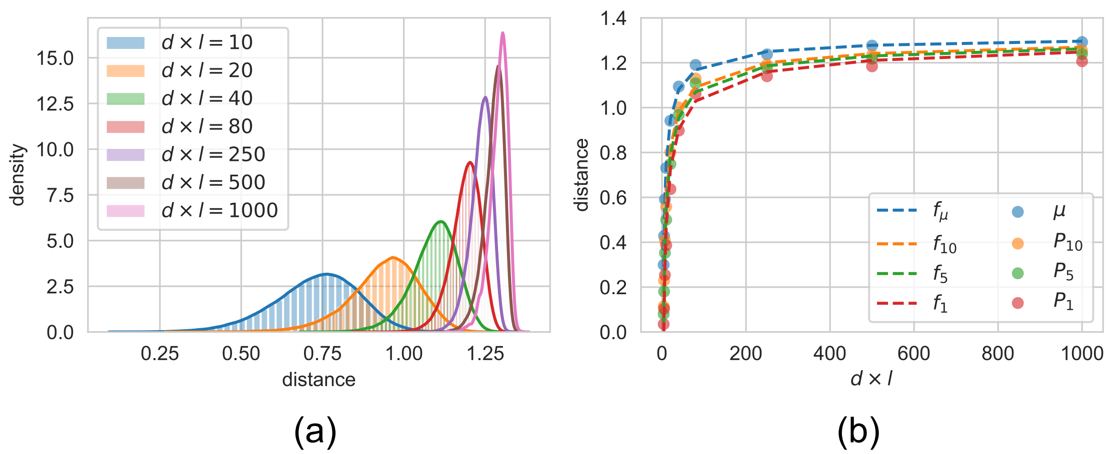


Figure 3.2: The effect of sequence dimension and length to the distance of random signals. (a) shows the probability distribution of the distance of random noise with different d x l combinations. (b) shows the dotted lines are the estimation of power-law function and the dots are true values of mean, 10-percentile, 5-percentile, and 1-percentile respectively.

To learn the threshold θ, we study how the random signal contributes to Z-normalized Euclidean distances and determine a threshold that can eliminate most of the random signal. From equation 1, we notice that apart from the signal itself, the distance is attributed to the subsequence length l and dimension d of the series. Adding l by

1 will increase the elements by a factor of d, and adding d by 1 will increase the elements by a factor of l. Therefore, we model the relationship by varying the value of $d \times l$ from a set of candidates from [4, ..., 1000] and choosing d and l arbitrarily. For each candidate, we generate 100 time-series using the approach proposed in [64] and compute the RCMP of the series to estimate the distance of a random signal at different $d \times l$. The result is shown in Figure 3.2 which reveals that the distance follows the power-law distribution in terms of the rise of the average values, as well as the scale of the variation. The reason is that when there are more elements to compare, it is less likely to find two similar subsequences just by chance. Therefore, the distances will converge if the number of points is large enough. We could estimate the distance of a random signal given different $d \times l$ with the following equation, setting $\alpha = -2.46, k = -0.62; \epsilon = 1.33$:

$$\theta(d \times l) = \alpha(d \times l)^k + \epsilon \tag{3.2}$$

In this work, we aim at a 95% confidence interval in eliminating random signals by computing the threshold as the estimated mean subtracted by 1.645 times the estimated standard deviation. Then, we could identify regions that contain SSPs by choosing the subsequences where the distance values are less than θ. Data from periods of idle activity may however contain drifts that have lower mutual distances causing false-positive just by chance. To overcome this issue, we modified the mSIMPAD to standardize the input series with 0 mean and standard deviation as 1. Then we insert a Gaussian noise with a scale of 0.1 to mitigate the effect of idle data. On the other hand, to avoid splitting the desired subsequence into smaller parts due to an abnormal spike or valley, we introduced a greedy, iterative merging approach to combine a split with its surrounding subsequences if the length of the split is less

Table 3.1: Performance comparison on repeating pattern detection.

	Method	Accuracy	Precision	Recall	F_1 Score
HAPT	SIMPAD	0.970	**0.994**	0.945	0.968
	mSIMPAD	**0.971**	0.991	0.951	0.970
	R-SIMPAD	**0.971**	0.945	**1.000**	**0.972**
	R-mSIMPAD	0.966	0.935	**1.000**	0.966
mHealth	SIMPAD	0.692	**1.000**	0.579	0.731
	mSIMPAD	0.777	**1.000**	0.696	0.819
	R-SIMPAD	0.891	**1.000**	0.852	0.920
	R-mSIMPAD	**0.915**	**1.000**	**0.884**	**0.938**
PAMAP2	SIMPAD	0.808	**0.994**	0.712	0.829
	mSIMPAD	0.816	0.990	0.729	0.839
	R-SIMPAD	**0.933**	0.970	0.923	**0.946**
	R-mSIMPAD	0.928	0.952	**0.935**	0.943

than l. It starts from the split with the smallest length and iteratively merging those splits until all of the splits have at least length l. Then the subsequence is verified for containing SSPs by majority voting.

The improved method relaxes the assumption that the input time series must contain both repeating and non-repeating patterns. It also provides more accurate detection results as the inserted Gaussian noise can better differentiate the idle and non-idle components in sensor data. We employed the same evaluation metric in [74] and we further discard irrelevant data such as subsequences labeled as a transition since they may contain any activities including repetitive activity (e.g. walking to another location) during the transition phase. The improved methods are denoted as R-SIMPAD and R-mSIMPAD respectively, that are more robust and have fewer assumptions on the input signal compared to their original forms. The results are reported in Table 3.3.4. We notice that our algorithm performs similarly to existing work on the HAPT dataset but significant improvements can be observed on both the mHealth and PAMAP2 datasets. The number of false negatives has been drastically reduced as we can see from the much higher recall rate, resulting in an over 11% improvement in the F_1 score.

Scalable Template Extraction Method (STEM)

With the above-given method, we identified subsequences that contain SSPs and provided a rough estimation of pattern lengths. Then for each subsequence, we compute the distance profile DP_i, and select the nearest neighbor $T_(j, l)$ iteratively if the distance is less than θ. The distances from $j - pr$ to $j + pr$ are discarded for the selected nearest neighbor $T_(j, l)$ with a pruning range $pr = \gamma \times l$, in which γ is the pruning factor of the pattern length. This process will repeat until all of the distances are discarded or are greater than θ. Then we estimate the quality of the match by averaging all of the distances of the chosen subsequences that are denoted as *template candidates* and select the set of candidates with the minimum average distances as the best match of the SSP. However, the estimated length of the pattern is relatively rough, which may over or underestimate the true length of the pattern. We proposed a two-step approach to better refines the length of the extracted template. The SSP should either be continuous or vary at different intervals. We could evaluate the variation of the positions around the start and end points of the candidates. If the SSP is continuous, we could refine the template by minimizing the distance between the start and end positions. The intuition is that the pattern should appear one after another such that the variation should remain relatively low as we can see from Figure 3.3; otherwise, the variation should be relatively high such that we can refine the length by minimizing the averaged variations. From the experiment, we notice a huge improvement in terms of recognition accuracy with this template refinement approach as we will show in section 3.4.

The recognition is then performed by comparing the templates. One can imagine that

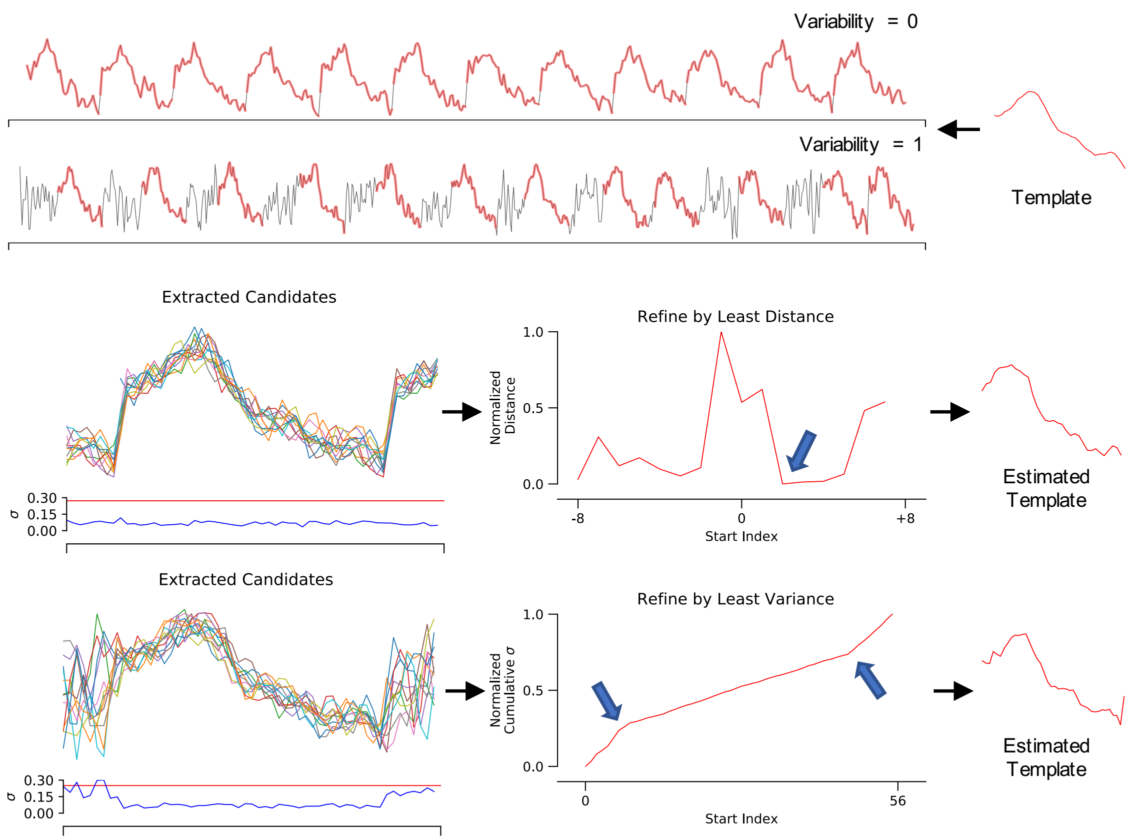


Figure 3.3: Example of a repeating pattern with different variability and the feature extraction and refinement procedure. It finds a set of SSP candidates, then estimates the point-wise variation among them. It then refines the start and end position by either the least variance or the least distance, based on the maximum point-wise variation that is smaller or larger than some threshold σ accordingly.

for continuous patterns, it is difficult to determine the start and end positions. The extracted template may be misaligned where the start and end positions lie around the middle. When comparing two templates, we align them by padding one by itself to cover all possible extracted cycles and compute the cross-correlation between the padded template with the other. Then, we roll the template by maximizing the cross-correlation.

On the other hand, the template generated from one sequence might be slightly different from another in terms of length and shape. We apply an elastic distance

Algorithm 4: Scalable Template Extraction Method

Require: T, int l, double θ, double γ=1
Ensure: TP
$C_{best} \leftarrow []$;
for $i \leftarrow 1 : n - l + 1$ **do**
  $DP \leftarrow$ computeDP($T_{i,l}, T$);
  $C \leftarrow$ findCandidates(DP);
  if $D(C) < D(C_{best}$ **then**
    $C_{best} \leftarrow C$;
  end if
end for
$\sigma_C \leftarrow$ pointwiseSTD(C_{best});
if $\sigma_C < std(T)$ **then**
  $C_{best} \leftarrow$ refineByNearestPoint(C_{best});
else
  $C_{best} \leftarrow$ refineByVariability(C_{best});
end if
$TP \leftarrow$ median(C_{best});
return TP;

metric to handle these kinds of small differences. Dynamic Time Wrapping (DTW) has been the most widely used measure for time series. However, [94] suggested that the Time Warp Edit Distance (TWED) consistently achieves the best performance in their study. Compared to other distances like DTW, the TWED is a metric that can potentially speed up computation such as clustering and retrieval. To reduce the computational cost of TWED, we adopted the window constraint as discussed in [103] to the TWED to limit the maximum warping of the TWED.

The above-mentioned method, denoted as STEM-TWED, aims to identify the subsequences which minimize the internal distance as a representation and recognize the subsequence by comparing the TWED with the labeled templates. It can distinguish tiny differences among time series which is especially suitable for differentiating fairly similar, low dimensional series. However, it relies on accurate length estimation as the differences in lengths between time series incur higher costs. mSIMPAD only offers a rough estimation of the pattern length, and it depends on the input of the length candidates. Moreover, time series distance measures such as DTW and TWED are computationally expensive. Therefore, we introduce a variation of STEM we call STEM-ECDF that incorporates the existing feature extraction method, namely the Empirical Cumulative Distribution Function (ECDF) which is simple, yet very robust even compared with state-of-the-art deep-learning-based features [51].

We employ the same detection and candidates extraction approach as mentioned above. Instead of using a single template as a representation, we compute the ECDF of all the template candidates as a representation. Each detected subsequence will then be represented by the ECDF vector that preserves the distributional information of the template. Since the ECDF features have the same number of dimensions, we can leverage traditional machine learning models for recognizing the template. This allows recognition with much lower computational cost, while still being capable of handling repeating patterns with irregular intervals. We delayed the discussion on the merit of this approach until section 3.5.

3.4 Experimental Evaluation

In this section, we report the experimental evaluation of the proposed approach to SSP recognition and compare it to other baseline activity recognition methods. We used one synthetic dataset and three sensor-based activity datasets. With the ground truth being available in the synthetic data, we specifically measured the performance of template extraction from three aspects: pattern length estimation, template candidate selection, and the similarity between the extracted template and the ground truth pattern. We then measure the performance of activity recognition on the three public datasets to illustrate the robustness of real-world applications. Finally, a use case of wireless-sensing based respiration monitoring is provided to demonstrate that the proposed method as a general approach has a broad range of applications with great impact.

3.4.1 Experiment Setup

The baseline methods include one of the most widely used statistical features [68] and the Empirical Cumulative Distribution Function (ECDF) [47] for IMU-based activity recognition. mSIMPAD and STEM perform directly on the input series, while the baseline methods are sliding window based. To provide a fair comparison, we divide the recognition result into equal-length segments, which is done in the baseline methods. Then we perform majority voting within each segment to decide whether it contains an SSP and choose the extracted template accordingly. To mitigate the effect of recognition model parameters, we apply the Nearest Neighbor (NN) classifier for both the STEM and baseline methods.

Evaluation Metric

We adopted the weighted F_1 score as the evaluation metrics defined as $\frac{1}{|X|}(\sum_{i\in C} 2|X_i| \times \frac{precision_i \times recall_i}{precision_i + recall_i})$ where C is the set of given classes, $|X|$ is the number of all testing instances, $|X_i|$, $precision_i$ and $recall_i$ refer to the number of testing instances, precision, and recall of a particular class i respectively. The $precision_i$ is defined as $\frac{TP_i}{TP_i+FP_i}$ and the $recall_i$ is defined as $\frac{TP_i}{TP_i+FN_i}$, where TP_i, FP_i, and FN_i refer to the true positive, false positive, and false negative of a particular class i respectively.

Synthetic Data

To get a better grasp of the performance of the proposed approach, we need a dataset we have full control over, including knowledge of the shape of the pattern, the number of repetitions, and the interval variability. Therefore, synthetic data is required for evaluation purposes, as well as to help determine the proper set of parameters for

Table 3.2: List of activities.

	Repetitive Activities	Non-repetitive Activities
HAPT	walking, walking upstairs, walking downstairs	sitting, standing, lying
MHEALTH	walking, climbing stairs, jogging, running, cycling	sitting, standing, lying
PAMAP2	walking, walking upstairs, walking downstairs, running, cycling, Nordic walking	sitting, standing, lying, watching TV, computer work

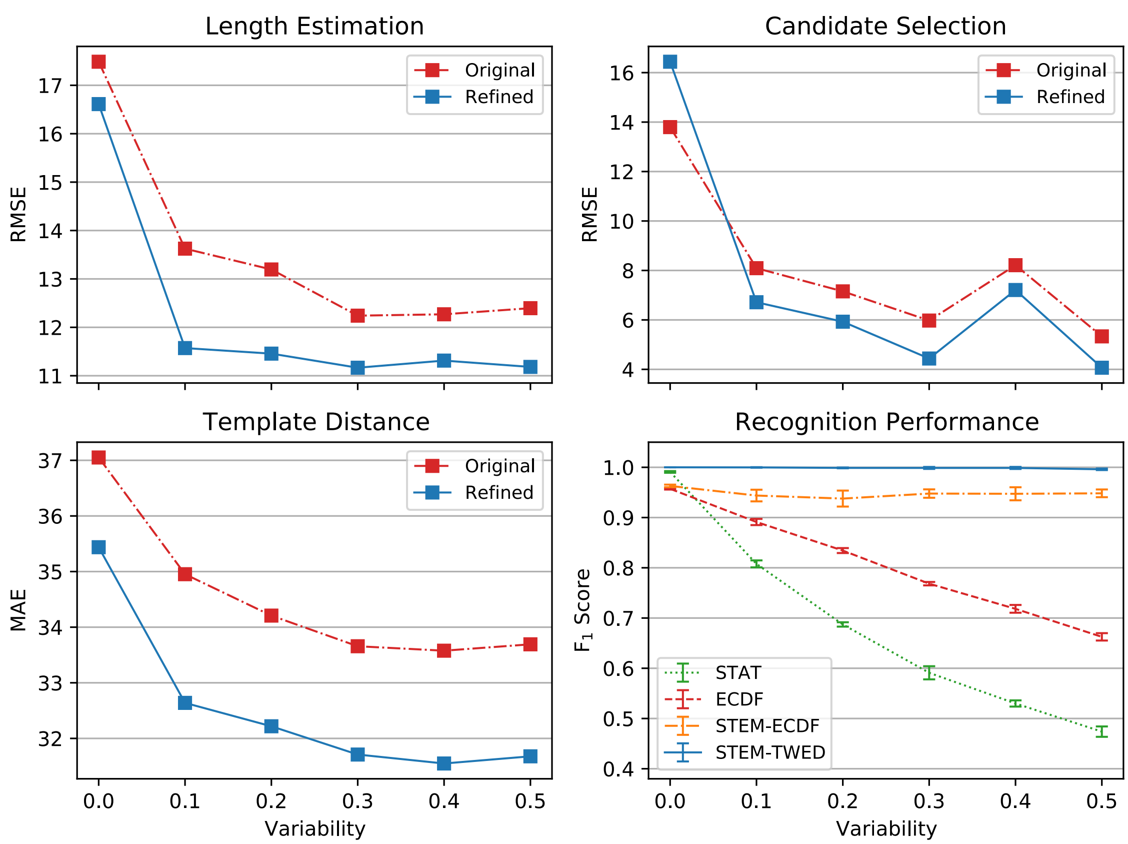


Figure 3.4: Evaluation result on the synthetic dataset in terms of length estimation, candidate selection, template extraction, and recognition performance.

the algorithm itself. We first randomly generated 19 random walk time series with lengths between [40,. . . ,80]. The 19 time-series are treated as templates, and for

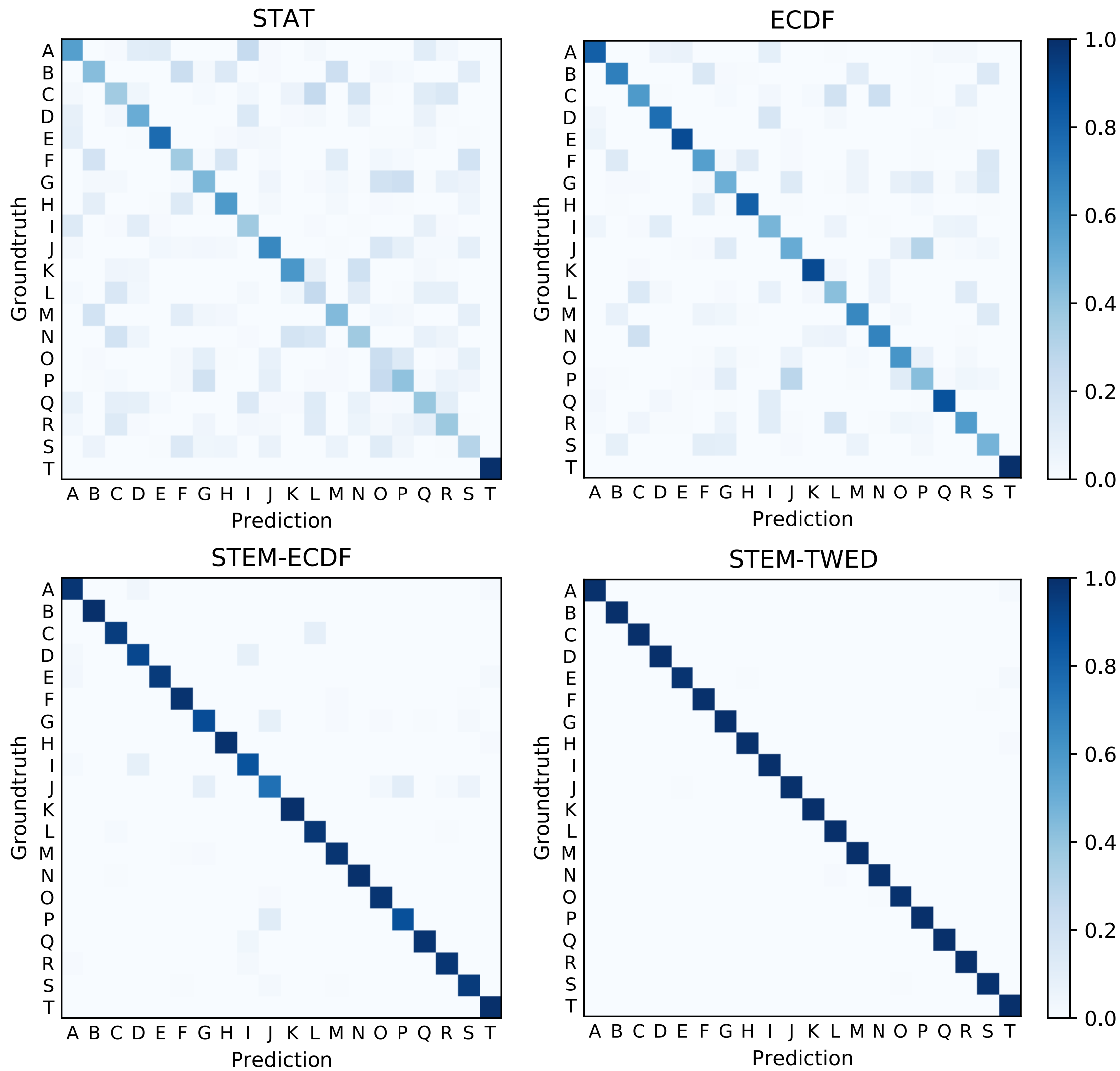


Figure 3.5: Confusion matrix of different methods on the synthetic dataset.

each template, we generate 100 time-series repeating the template from 5 to 20 times randomly. We also generated 100 random time series with no repeating patterns as one negative class. Gaussian noise was then added to each of the generated series, resulting in 2000 synthetic time series.

We produced six datasets following the above-mentioned procedures with the same 19 templates, in which we introduce different variable intervals between [0,0.1,...,0.5].

The variable interval v is a factor of the pattern length l, which determines the standard deviation of the interval equal to $v \times l$ and the interval between patterns following the normal distribution. Figure 3.3 shows an example where variability equals 0 means the templates appear one after another, and variability equals 1 means the templates appear at varying intervals in which the variation follows a normal distribution with standard deviation as $1 \times$ the pattern length.

Public Datasets

We choose three publicly available activity datasets for the evaluation. The HAPT [15] collects data from 30 volunteers wearing a waist-mounted smartphone while performing various activates in laboratory conditions: walking, walking upstairs, walking downstairs, sitting, standing, and Lying down. MHEALTH [17] is composed of 12 activities in an out-of-lab environment, performed by 10 volunteers with 3 sensors placed on the subject's chest, right wrist, and left ankle. PAMAP2 [115] includes 18 activities performed by 9 subjects wearing 3 sensors on the subject's chest, and the dominant side's hand and ankle.

Table 3.3: List of activities ID.

ID	Activity
A0	Non-repetitive Activities
A1	Walking
A2	Walking Upstairs
A3	Walking Downstairs
A4	Jogging
A5	Running
A6	Cycling
A7	Nordic Walking

We manually classify activities as repetitive activities and non-repetitive activities for each dataset, where non-repetitive activities are treated as one class. The details of the classification are mentioned in Table 3.3. We choose only two IMUs (one from the hand and one from the ankle) from the MHEALTH and PAMAP2 datasets, as the chest data does not contribute much information on the listed activities. For each of the detected segments, we extract the template (denoted as STEM-TWED) and the ECDF features (denoted as STEM-ECDF) from the template candidates as mentioned in section 3.3. Then the sliding window is applied to extract statistical and ECDF features directly from the window data. The STEM-TWED and STEM-ECDF features are also selected on the same window by majority voting. The size of the window is defined as 5 seconds with a step size of 2.5 seconds.

3.4.2 Evaluation of Repetitive Activity Recognition

Synthetic Data

We assess the quality of the extracted template by evaluating its performance on *length estimation*, *candidate selection*, and *template extraction* measured by the similarity between the template and the ground truth. The results can be found in Figure 3.4, and detailed evaluations are outlined in the following section.

Length estimation evaluates the ability to choose the correct pattern length by measuring the length differences between extracted templates against the ground truth. We measure the Root Mean Squared Error (RMSE) for the differences and the result shows that the RMSE decreases with the increasing variability in general. Also, the refined template recorded consistently lower RMSE, suggesting that the refinement technique can provide a more accurate length estimation.

We measure the differences between the location of the selected template candidates and the locations of ground truth patterns to assess the quality of selected candidates. To mitigate the effects of the length variations, we compute the differences using the center point of the template candidate and the ground truth. Again, we measure the RMSE of the differences for the evaluation. Noticing that the number of selected candidates might be different from the ground truth number, we match the largest common indices with a greedy approach. We start from matching the pair of indices with a minimal difference and remove those pairs from the list iteratively. This process repeats until either the list of candidate indices or ground truth indices are empty. It is reasonable to ignore patterns that have not been covered, since the goal of the STEM is not to identify all patterns but to discover the most internally similar subsequences. subsequences that are covered are distinct from the selected candidates, and including such abnormal patterns might degrade the performance of template extraction. The results show that the error generally decreases with increasing variability. The performance of the refined template is also consistently better than the original template except for when variability is close or equal to 0. This happens for patterns without a variable interval, since it is possible to identify any snippet of the series as the template itself while remaining at a low averaged distance, which is still correct as it identifies the pattern as long as the length is also estimated properly.

We examine the distance between the template and ground truth, where we align the two sequences and measure their mutual distance with TWED. As expected, the distance also decreases with increasing variability as we achieved better length estimation and candidate selection. Moreover, the performance of the refined template is consistently better than the original template. The above evaluation suggests that the proposed template extraction method is satisfactory for patterns with regular periodicity, while still very robust when the pattern contains variable intervals. The template refinement is also needed as it provides more accurate length estimation and candidate selection which leads to a more precise template, and therefore the distance is much lower than the original template.

The analysis above demonstrated how the proposed method performs on template

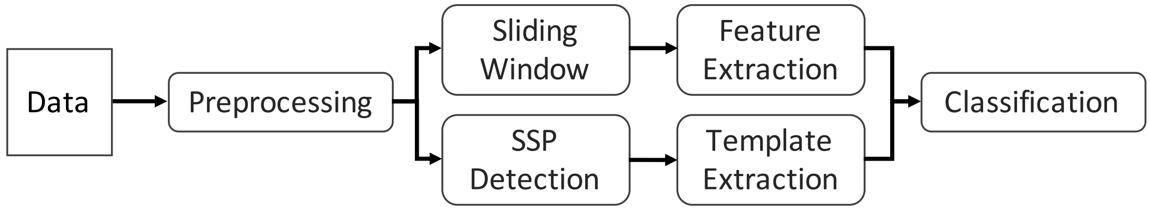


Figure 3.6: An illustration of the efficiency achieved over the traditional approach. SSP detection reduces most of the unnecessary computation by eliminating segments without any SSP.

extraction under different variability conditions of the repeating pattern. We then examine the performance of recognizing the pattern with the extracted templates. For each dataset of different variability, we perform 5-fold cross-validation to divide the dataset into 5 equal size partitions randomly. We then perform evaluations using each partition as the testing set and the remaining partitions as the training set. The performance is then calculated as the average over the 5 partitions as shown in Figure 3.5.

We can see that the proposed STEM-based method achieved superior performance compared to the baseline methods. Although the RMSE of length estimation seems to be quite high, the distance computed by TWED can still efficiently distinguish series with tiny differences. The results show that STEM-TWED achieved a 23% improvement compared to the best baseline method when the variability is up to 0.5.

Public Datasets

We evaluate the performance of repetitive activity recognition from two aspects: *efficiency* and *classification accuracy*. Efficiency is evaluated by measuring the degree of reduction of unnecessary computation as shown in Figure 3.6, whereas accuracy, is the degree to which repetitive and non-repetitive activities are correctly identified. Successfully classified subsequences of both the repetitive and non-repetitive activi-

Table 3.4: Comparison of recognition performance for different methods on the public datasets.

Dataset	features	A0	A1	A2	A3	A4	A5	A6	A7	Weighted Average
HAPT	STAT	**0.999**	0.864	0.893	0.914	—	—	—	—	0.952
	ECDF	**0.999**	**0.987**	0.991	**0.992**	—	—	—	—	**0.995**
	STEM-TWED	0.989	**0.987**	0.978	0.962	—	—	—	—	0.984
	STEM-ECDF	0.996	0.984	**0.999**	0.964	—	—	—	—	0.990
mHealth	STAT	0.999	0.929	0.930	—	0.645	0.648	0.966	—	0.900
	ECDF	0.965	0.991	0.993	—	0.784	0.741	0.915	—	0.921
	STEM-TWED	**1.000**	0.624	0.764	—	0.467	0.492	0.819	—	0.767
	STEM-ECDF	**1.000**	**1.000**	**1.000**	—	**0.867**	**0.841**	**1.000**	—	**0.969**
PAMAP2	STAT	**0.962**	0.711	0.818	0.815	—	0.891	0.937	0.682	0.859
	ECDF	0.921	**0.746**	0.831	0.790	—	0.886	**0.940**	0.699	0.855
	STEM-TWED	0.923	0.498	0.654	0.693	—	0.579	0.777	0.516	0.705
	STEM-ECDF	0.948	0.649	**0.890**	**0.859**	—	**0.983**	0.886	**0.717**	**0.882**

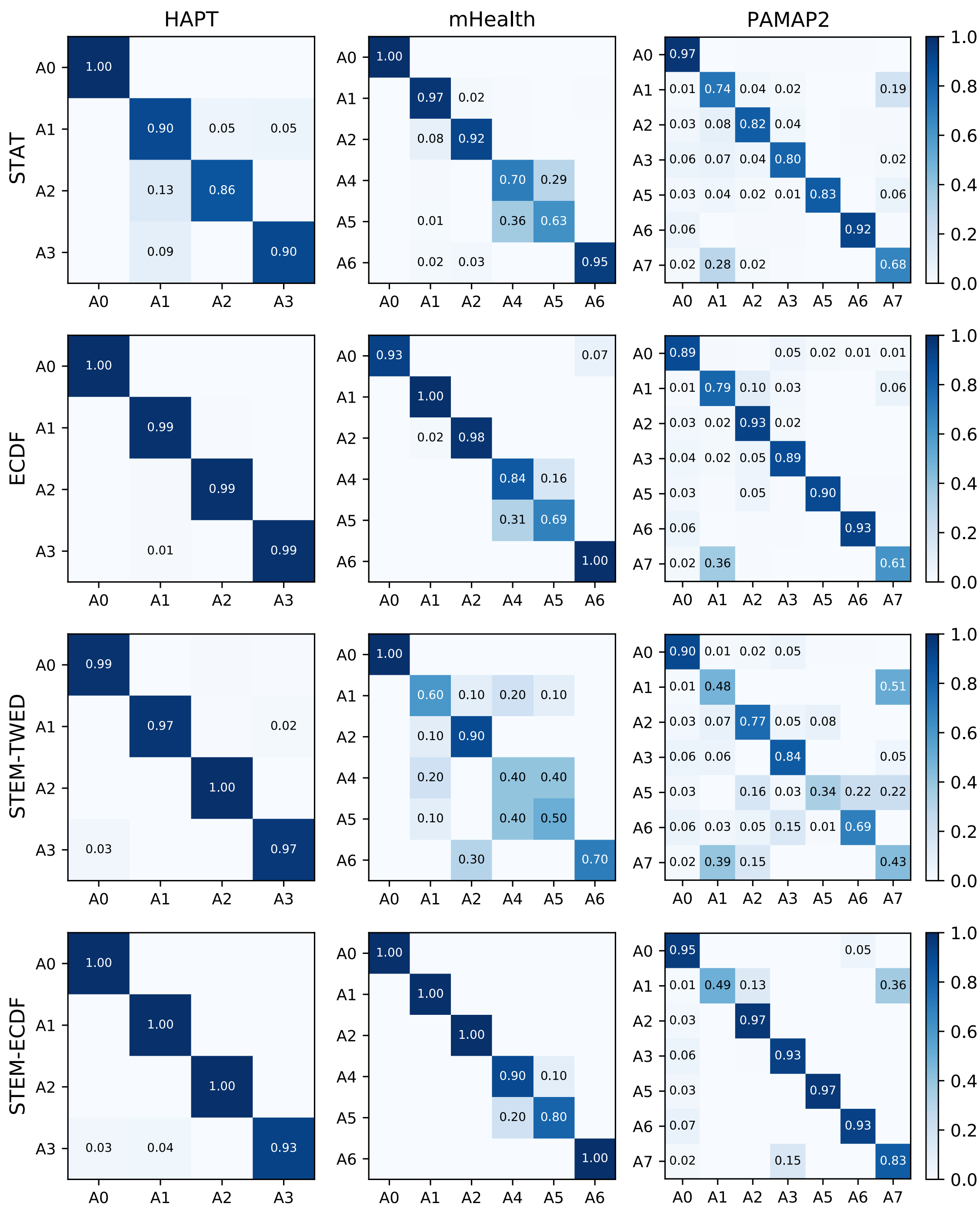


Figure 3.7: Confusion matrix of different methods on the public datasets.

ties are the true positives from both sets. We recorded an average true positive rate of repetitive activity classification of 92.4% and 93.9% respectively for SIMPAD and mSIMPAD over the three datasets; and 96.6% and 95% for non-repetitive activity classification. This suggests that the mSIMPAD is better at identifying repetitive activities at 93.9% while eliminating 95% of the non-repetitive segments.

Note that for the public datasets, some of the classes only have a few samples so that we only perform 3-fold cross-validation for the evaluation on recognition accuracy, and the results are reported in Table 3.4. The two baseline methods produce similar results while the ECDF features perform slightly better in general. Surprisingly, the STEM-TWED had the worst performance, which shows that the individual differences can largely degrade the recognition performance with an elastic distance measure approach. For instance, one's pattern of jogging might be more similar to another's, and as we can see from Figure 3.7, the STEM-TWED has difficulty distinguishing between jogging and running.

With the abstraction of the pattern using statistical features, STEM-ECDF achieved the best performance in two out of three datasets. This is because the STEM-ECDF can better identify internally similar patterns and ignore abnormalities to form better quality features out of the repetitive activity. However, as we have illustrated the superior performance of STEM-TWED on the synthetic dataset, it suggests the STEM-TWED is more suitable for repetitive activity monitoring where the patterns are very similar in most cases. We delay the discussion in choosing the STEM-ECDF and STEM-TWED, and the reason that STEM-ECDF outperforms the original ECDF in section 3.5. In general, the proposed approach achieved superior performance compared to the baseline methods, recording a 3.3% improvement on average for the STEM-ECDF.

The above evaluation suggests that for repeating patterns with irregular intervals, the proposed approach performs significantly better than the baseline methods. However, measuring the distance between two sequences with time-warping techniques are known to be computationally expensive. The superior performance comes with a trade-off of computational efficiency. By combining the template extraction with the distributional features, the efficiency can be largely improved while achieving on par or even better result on repetitive activity recognition in reality.

3.4.3 Use case: Respiration Monitoring

Wireless sensing is an emerging area in the IoT community. Numerous publications have shown the potential of wireless-sensing based vital sign detection, which enables various applications in healthcare as well as activity recognition. To illustrate the potential of the proposed method on other repeating pattern extraction problems, we adopted respiration monitoring using wireless signals as a use case. Specifically, we identify the repeating patterns within a wireless signal captured from an RFID transceiver to estimate the respiration rate of a subject with an RFID tag on their chest. We randomly selected 10 signals collected in [140] in which half of them contain normal breathing, and the other half contain periods where the participants were instructed to hold their breath to simulate the condition of Sleep Apnea.

Respiration can be identified by measuring the phase of the wireless signal [80]. Intuitively, the physical motion of the chest affects the signal strength of the tag as it expands and contracts while inhaling and exhaling. These miniature changes constitute periodic patterns in the phase values of the signal. These patterns can be detected by STEM and provide the estimated pattern length if it exists. The located template candidates are considered as a signal of breathing in which we can estimate the breathing rate by counting the number of candidates. We compare the performance with the baseline method introduced in [140]. It assumed that the breathing pattern is a simple waveform signal that can be identified by peak detection. It computes a threshold, determined by the mean of the phase value to eliminate false positives caused by small variations. A normal breathing rate is roughly 30 times per minute and is considered as a physical limitation. This avoids peaks that are closer than 2 seconds apart. Then, we compute the number of breaths within each signal and calculate the RMSE for each of the methods. The RMSE of STEM is 0.89, where the RMSE of the baseline method is 3.66, which shows that STEM achieved much better performance compared to the baseline method.

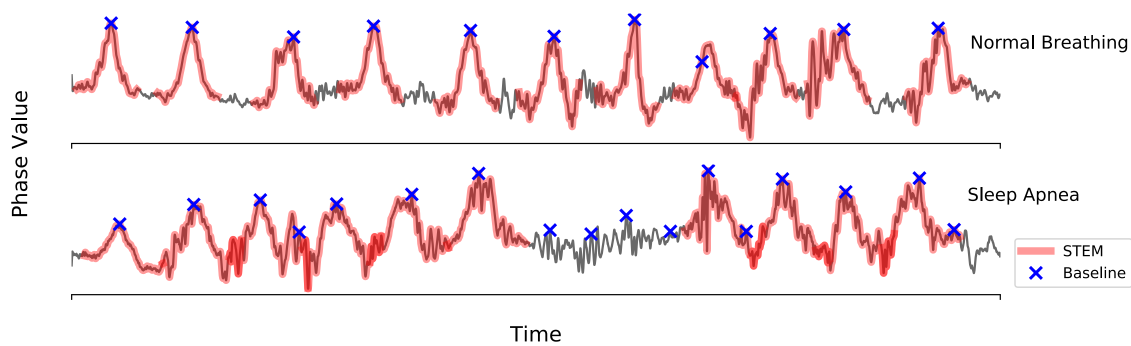


Figure 3.8: Example of respiration estimation. The red line denotes the detected pattern using STEM and the blue cross marks denote the detected respiration using the baseline method.

Figure 3.8 shows an example of normal breathing and simulated sleep apnea by holding one's breath. As we can see, STEM can accurately identify the waveform generated by breathing even with irregular intervals, periods of pause, and shape variations. In contrast, although the baseline method achieved similar results on normal breathing datasets, it fails if the signal contains periods of pause. The minimum distance between peaks simply does not work when the pattern contains irregular intervals as the breathing rate might vary from the average value. The drift of the signal can invalidate the threshold approach. In contrast, STEM first detects if the repeating pattern occurs by comparing the z-normalized distance between the subsequences, which can better handle data drift and shape variation. The inserted noise can better differentiate the truly repeating patterns from the non-repeating subsequences. Therefore, STEM achieved much better performance in estimating the breathing rate from the wireless signal. Note that this is just an example application of our approach. It can also facilitate many other applications such as heart rate detection, blink rate detection, and many repetitive motions such as hand-flapping, rocking, spinning, just to name a few.

3.5 Discussion

In this work, we focus on the classification of multivariate time series that may contain repeating patterns. These kinds of time series are prevalent in day to day life, and are especially interesting when considering repetitive activities [24, 45, 137, 86, 110], physiological signals [80], or the audio signals of music [113], just to name a few. We focus on the application of repetitive activities given their importance for physical health monitoring. The presented method is however general enough for other time series classification tasks with repeating patterns, as the proposed method is scenario independent where the only required parameter is the length of the target pattern.

Based on STEM, we proposed a recognition method using the nearest neighbor algorithm with time warp edit distance namely the STEM-TWED. It shows an ability for accurate recognition that however has a few drawbacks. First, the method compares against the z-normalized template that could misclassify similar patterns with totally different magnitudes. Second, the TWED relies on accurate length estimation as the differences between lengths incur a higher distance due to the warping penalty. Also, it is difficult to design a proper penalty if the templates are not normalized. Third, the computational cost increases with the number of training samples due to the distance measure for each sample, and calculating the TWED is much slower than calculating the Euclidean distance for the traditional features. To balance the recognition accuracy with the computational cost, we combine the STEM with the Empirical-Cumulative-Distribution-Function based features namely the STEM-ECDF. The advantage of the STEM-ECDF is that it can avoid unnecessary computation on non-repetitive series, while efficiently extracting template candidates and ignoring noisy data within the subsequences.

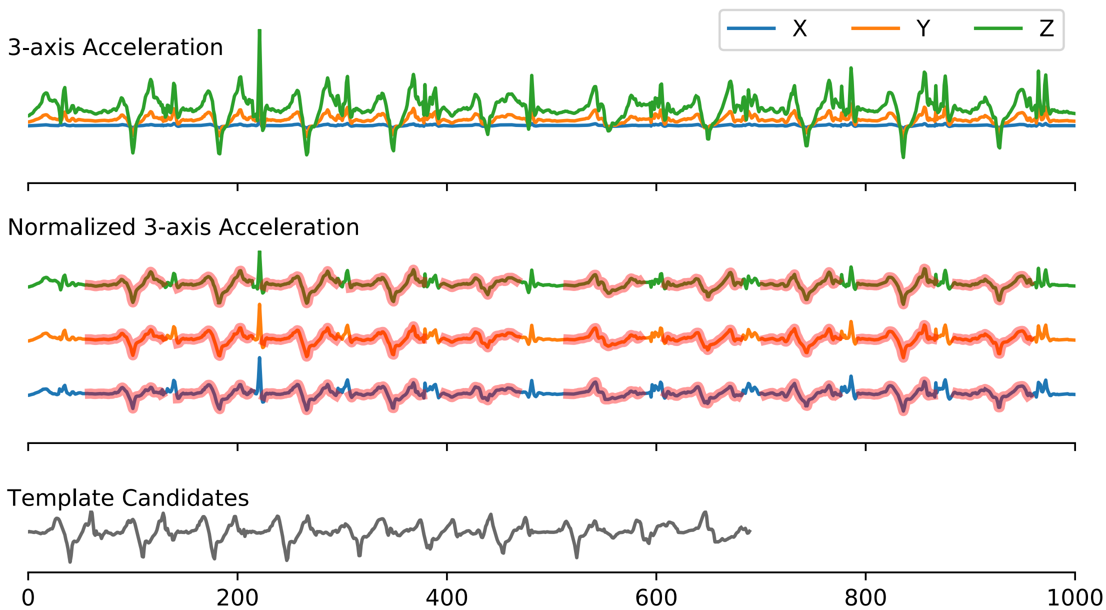


Figure 3.9: Example of candidate selection in a snippet of mHealth dataset.

Figure 3.9 shows a snippet of data from the mHealth dataset, obtained from the accelerometer on a person's ankle while climbing stairs. We notice that from the raw accelerometer signal, the X-axis is almost flat compared to the Z- and Y-axis data. If we normalize the data on each axis, very similar patterns can be observed among the three axes as shown in the middle of the graph. We observe slight differences between each repetition, but by using STEM we effectively extract internally similar subsequences within the series, such as the subsequences highlighted in red color. Since the three axes have almost identical patterns, we only show the candidates extracted from the X-axis in the lower part of the figure.

3.6 Conclusion

In this work, we proposed STEM, a template extraction method to identify repeating patterns in multivariate time series and apply it to repetitive activity identification. STEM aims to substitute the commonly used sliding window technique, which is computationally expensive and usually requires extensive domain knowledge to work. STEM leverages the recently proposed successive similar pattern detection method to determine whether repeating patterns occur within a time series. For these detected subsequences, it identifies a template, which is a pattern that minimizes the internal distances within the subsequence as a representation. We evaluated our approach on synthetic data, as well as on three publicly available datasets. The experiment shows that the proposed method can efficiently avoid unnecessary computation on non-repeating series. It also provides more accurate recognition results especially when the periodicity of the repeating pattern is variable. By combining the STEM with the distributional feature, it achieved a more balanced trade-off between computational cost and recognition accuracy. The proposed method shows superior performance compared to the baseline methods on repetitive activity recognition, and can additionally be applied to other time series classification tasks with repeating patterns.

Chapter 4

Eustress or Distress: An Empirical Study of Perceived Stress in Everyday College Life

Eustress is literally the "good stress" associated with positive feelings and health benefits. Previous studies focused on general stress, where the concept of eustress has been overlooked. This chapter presents a novel approach towards stress recognition using data collected from wearable sensors, smartphones, and computers. The main goal is to determine if behavioral factors can help differentiate eustress from other kinds of stress. We conducted a natural experiment to collect user smartphone and computer usage, heart rate, and survey data in situ. By correlation and principal component analysis, a set of features could then be constructed. The performance was evaluated under leave-one-subject-out cross-validation, where the combined behavioral and physiological features enabled us to achieve 84.85% accuracy for general stress, 71.33% one kind of eustress as an urge for better performance, and 57.34% for eustress as a state of a better mood. This work provided encouraging results as an initial study for measuring eustress.

4.1 Introduction

Stress is one of the major attributes of mental health has received growing interest from both industry and academia. Numerous studies suggest that stress is a health crisis, which associated with several diseases such as cardiovascular diseases, anxiety, and depression. A recent survey found that about half of the Americans experienced a major stressful event in the last year [105]. Many of them reported they suffer from stress-related behavioral responses including lack of sleep, losing appetite, and desire to exercise. Nowadays, the term stress is generally referred to as negative stress (distress) in our daily conversation. The adverse impact of stress has been studied extensively, whereas the positive aspect of stress has also attracted rising attention. For example, the business and management community aims at maximizing individual productivity by managing work stress. However, the concept of positive stress (eustress) is incomplete. Lacking knowledge about eustress obstructed the development of positive stress.

Typically, stress was assessed through a questionnaire or clinical assessment by a psychiatrist. In the last two decades, researchers tried to measure stress through physiological markers including heart rate, blood pressure, galvanic skin response, etc. The result of these methods is promising in a rigorous laboratory environment, however, not applicable to detect stress in daily life. Moreover, the concept of eustress has been overlooked in the past decades. In light of advanced mobile and wearable technology, data can be collected ubiquitously and less obtrusively, which enabled continuous stress assessment using ubiquitous sensing technology. To address these problems, we conducted a natural experiment and evaluated the classification result on the features extracted. We showed that ubiquitous computing is a potential method for evaluating eustress.

4.2 Background

The word stress was coined by Selye back in 1965, who defined stress as "the non-specific responses of the body to any demand for change" [119]. In general, it refers to the physiological responses caused by any stressful event (stressor). These responses are triggered by the Autonomic Nervous System (ANS), which influences internal organs and regulating heart rate, respiratory rate, blood vessel, galvanic skin response, and so on. ANS is divided into two subsystems, namely the Sympathetic Nervous System (SNS) and the Parasympathetic Nervous System (PNS). When stressful events arise, a higher activity rate in SNS, which signals the adrenal glands to release stress hormones (e.g. adrenaline and cortisol). These hormones led to physiological changes, also known as the "*fight or flight*" response. Alternatively, activity in PNS increases during the restful event.

Selye introduced the concept of positive stress, namely eustress in 1976 [122]. He extended his work in stress to distinguish eustress and distress in terms of adaptiveness toward stress response, where eustress is "healthy, positive, constructive results of stressful events and stress response" [66]. Lazarus considers eustress as a positive cognitive response to a stressor, which is associated with positive feelings and a healthy physical state [70].

Another dominating approach to understand eustress was developed on the Yerkes Dodson Law [20]. It suggests that stress is beneficial to the performance until some optimal level is reached, after which performance will decline, which follows the inverted U shape diagram.

4.3 Related Work

Owing to the unclear criteria to distinguish eustress from others, the existing analysis focused on general stress. Various stress measurement methods using computational techniques have been proposed in the last two decades [124]. These methods can be classified into two categories: *physiological measures* and *physical measures*. The former one evaluates mental stress by monitoring different physiological responses including skin conductivity, heart activity, brain activity, blood pressure, etc. The later one collects physical characteristics (e.g. body gesture, facial expression, voice, etc.) that is sensitive to stress and using machine learning methods to develop a computational model for stress recognition. Among all different types of input, Sharma & Gedeom suggested that heart rate variability (HRV) rank the top among different primary measure for assessing mental stress in terms of accuracy and non-intrusiveness.

Sun et al. consider stress assessment as a detection problem, which takes accelerometer data into account to filter the effect of motion artifact [130]. In [116], the authors collected data from wearable sensors and mobile phones in situ, which accuracies range from 75-87.5% for 2-class classification problems with different feature sets. Their work was extended in [36] with a larger population and longer period, in which achieving classification accuracies range from 67-92%, showing that behavioral features are possible to recognize mental stress on a daily basis.

Existing methods investigated the pattern of physical and physiological sensory data under general stress. In our work, we study the feasibility of measuring eustress by HRV, smartphone, and computer usage data. To the best of our knowledge, we are the first who proposing a classification model toward eustress.

4.4 Research Questions

On the basis of previous work, general stress can be recognized by a physiological signal with high accuracy and suggested that stress is related to a number of behavioral factors such as multitasking, applications used, and physical activity. Recall one of the explanation of eustress regarding performance, multitasking lead to task-switching cost which associated with a decrement in performance [98]. It is obvious that smartphone and computer use are the major source of interruption, and closely related to multitasking. Therefore, we designed the experiment to investigate the possibility of using physiological and behavioral signals together to build an accurate classifier of eustress recognition. Since there has no single domination definition towards eustress, we assess eustress is twofold: 1) Higher self-reported performance along with moderate stress level [20]; 2) Higher self-reported mood along with moderate stress level [70]

It is not difficult to realize that too much or too little stress might not trigger "eustress" in terms of the definitions mentioned in the previous section. Therefore, we assume eustress must be under a moderate physiological stress level. To answer these questions, we study the pattern of whether these behavioral features are able to correlate to this situation.

4.5 Study Protocol

We designed an in situ study and recruited 7 physically healthy subjects (5 male and 2 female) with ages ranging from 22 to 28, in which all of them are either research students and staff. We collected data from each participant on 5 days during their waking hours. During the study, three sources of data are collected from (1) sensor and application on smartphone, (2) application on personal computer, and

(3) wearable heart rate sensor. These data can be categorized into heart rate, usage, and survey measure respectively.

We developed StressSurvey, which is an application for Android smartphones to collect smartphone activities and other sensory data. It connects the heart rate sensor automatically in the background, and recording heart rate data transmitted. It also captures smartphone screen and call activities. Every hour in between 8 AM to 12 PM, the application reminds the participant to report the survey by notification. The details of the data acquisition process are described in the following section.

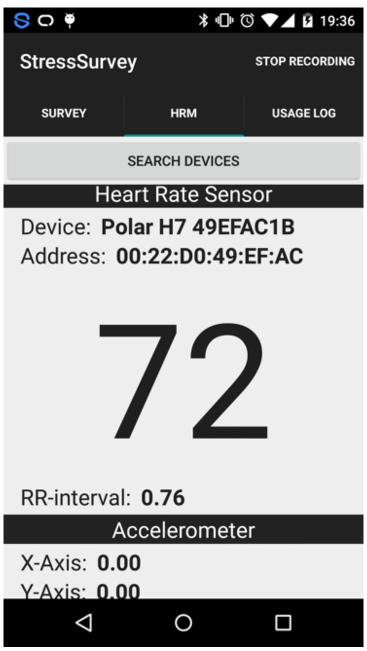


Figure 4.1: Control panel for heart rate measurement.

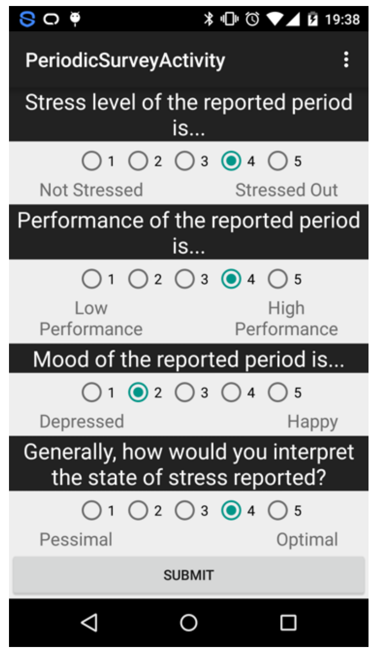


Figure 4.2: Example of periodic survey.

Heart rate measure. Heart rate variability is collected using the Polar H7 heart rate sensor [6], wearing a chest band to record beat-to-beat interval and average heart rate. The heart rate data is measured by the ECG sensor and preprocessed within the H7 device. Then it transmits the record in 1000ms via Bluetooth to the Android smartphone. Since the connection is using Bluetooth 4.0 (BLE), the smartphone is required at least Android version 4.3 with BLE enabled (e.g. Nexus 5, Galaxy S3). The data transmitted complies with BLE specification, where the characteristic specified the format of the record. Each record is either 8- or 16-bit int format, indicated by the first bit of data (0 for 8-bit int, 1 for 16-bit int). Bit 1 and 2 indicate whether the sensor contact feature supported and the sensor contact status. Bit 3 is the indicator of energy status that indicates if energy expended data is presented. Bit 4 indicates if RR-interval data is presented, and the interval is represented in 1/1024 sec. We shift the reading byte by checking the flag data. Each record is stored with UNIX timestamp on the smartphone in common separated values (CSV) format.

In order to eliminate the effect of heart rate due to human artifact, motion data were collected along with heart rate measure, obtained from the accelerometer on the android smartphone. Each motion data contains a three-dimensional vector, which was calculated after removing the influence of the force of gravity.

Smartphone and computer usage measure. Usage log is collected via commercial application RescueTime [7]. Participants are asked to install the RescueTime client application on both computers and smartphones, each of them is assigned with seven prepared user accounts: hkpu.stresssurvey.#@gmail.com where # is an integer id from one to seven. Data can be downloaded through the public API, each row contains the timestamp, application name, category, duration, and estimated productive index ranging from -2 to 2. We collected the most fine-grained record in five minutes interval for each participant. The screen on and off events and the state of smartphone calls are collected directly by StressSurvey. Each record comes with an event indicator and timestamp and is stored locally in CSV format.

Survey measure. This study using the experience sampling method (ESM) to capture self-reported survey from time to time. During the daytime, the application sends out the notification to remind the participant to complete a survey every hour. The survey consists of several questions and provided an integer scale ranging from one to five, asking the perceived stress, performance, and mood. Participants completed the end-of-day survey rated the same scale according to a daily basis.

Table 4.1: Statistic of the data for each participant.

Subj.	Survey	# of reports (1-5)					Total
	Stress	11	1	3	1	0	
1	Mood	1	1	3	3	8	16
	Performance	2	5	5	1	3	
	Stress	10	3	2	2	0	
3	Mood	0	1	6	8	2	17
	Performance	4	7	4	2	0	
	Stress	1	6	6	1	0	
4	Mood	0	1	7	6	0	14
	Performance	0	4	9	1	0	
	Stress	3	9	22	4	2	
5	Mood	0	5	19	14	2	40
	Performance	0	10	20	9	1	
	Stress	7	17	5	1	0	
6	Mood	0	2	17	11	0	30
	Performance	4	9	9	8	0	
	Stress	2	3	1	9	11	
7	Mood	9	13	1	3	0	26
	Performance	5	8	12	0	1	

4.6 Data Overview

Over 7 participants, one was excluded from the analysis because the heart rate sensor was disconnected most of the time. We collected 5,058,233 accelerometer data, 1,410,109 heart rate data, 10,851 screen activity data, 878 call activity, 14,746 smartphones and computers usage, and 252 self-reported survey data in raw format.

By removing incomplete data, there are 143 survey data combined with sensory, usage, and survey data aggregated on an hourly basis. Statistic of the reported survey is shown as table 4.1, where we found that each participant has their own preference for reporting their values.

In general, perceived stress is positively associated with performance and inversely for mood as shown in figure 4.3. The average use of smartphone and computer increase starting from 6 AM to 11 AM and reach the first peak in the morning. After this, it slightly drops from 12 PM to 1 PM. The use of computers and smartphones at night decreased significantly.

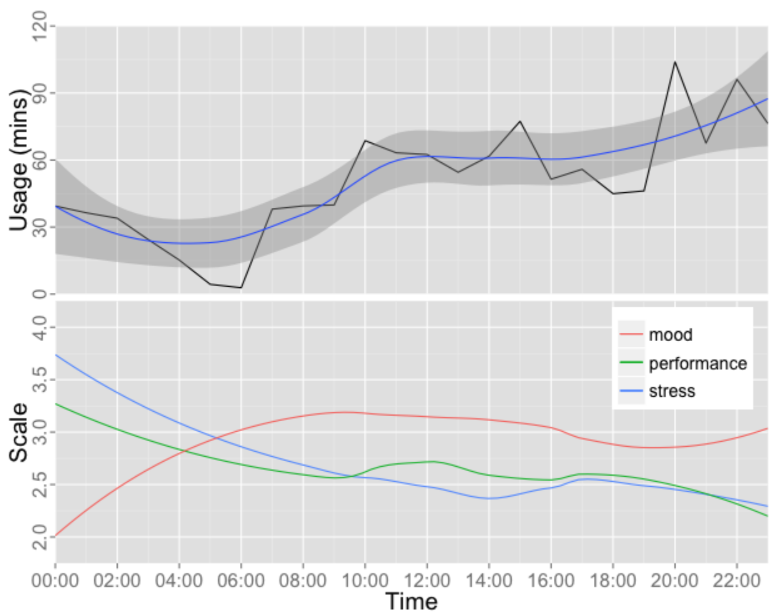


Figure 4.3: Average of inter-subject computer and smartphone usage (duration) and survey value.

4.7 Feature Extraction

Data especially heart rate measure requires cleaning and transformation prior to classification. First of all, we remove obvious errors (e.g. heart rate < 40), and RR-interval that is more than 20% different from the previous one. Then, the value is interpolated by the moving average. The summary of features extracted is shown as table 4.2.

4.7.1 Heart Rate Measure

Heart rate measure (HRM) including average heart rate data and actual R-R interval obtained from the heart rate sensor. The average heart rate data were aggregating in 60-minute windows, in which the standard deviation (SDHR) and the average (AVHR) of heart rate were derived. Heart rate variability features can also be extracted from the windows including the standard deviation of NN-interval (SDNN), an average of NN-interval (AVNN), percentage of adjacent NN-intervals differing by more than 50ms (pNN50), and root-mean-square differences of successive R-R intervals (RMSSD). For frequency-domain features, since the sampling rate deviates because of the system operation, and the number of samples is not necessarily the product of two. Therefore, we employ the Lomb-Scargle Periodogram [62] that is capable to analyze unevenly sampled time-series and data sets with missing values.

Then the power spectrum obtained is sum up to three separate bins, grouped by very low frequency (VLF) $<$ 0.04 Hz, low frequency (LF) 0.04 - 0.15 Hz, and high frequency (HF) 0.15 - 0.4 Hz respectively. In addition, the accelerometer data were collected during heart rate measurement is available. Then the motion intensity (MI) was defined by $\frac{1}{3}(|ACC_x|+|ACC_y|+|ACC_z|)$, where average and standard deviation of MI were calculated in the 60-minute windows aligned to the HRM features.

4.7.2 Smartphone and Computer Usage

Usage log including smartphone screen, call state, and application used are captured from smartphones and computers. For screen and call activities, duration and fre-

Table 4.2: Summary of extracted features for stress classification.

Modality	**Features**
Heart rate measure	AVHR, SDHR, AVNN, SDNN, RMSSD, PNN50, VLF, LF, HF, LF/HF
Motion	AVMI, SDMI
Screen	Duration of screen on time (secs), frequency of screen on event
Call	Number of call, answered call; Duration of off-hook
Application	Duration of each category: social, entertainment, internet, communication, study, email

quency are extracted from raw data. For application usage, records are aggregated on an hourly basis. Each record consist of the name of the application, time of the usage recording, and duration of each application. Some other information such as category and estimated productivity provided by RescueTime were not used. The usage record is then labeled manually into the following categories: internet, email, social, communication, study, and entertainment. Then the sum of the duration of applications used from the same category was calculated. In order to eliminate the individual difference among different participants, the categorized data was used to derive three ratios namely: social, productive, and non-productive ratio. Then we perform the dimension reduction by using Principal Component Analysis (PCA), to further eliminate linearly dependent features.

4.8 Classification Result

In this section, we present the process of training the classifier, and the result of different approaches. We use the R (programming language) to build various classifiers using well-known learning methods: Multinomial Logistic Regression (MLR), Support Vector Machine (SVM), and Random Forest (RF), to evaluate the predictive power on the linear classifier, non-linear classifier, and ensemble classifier respec-

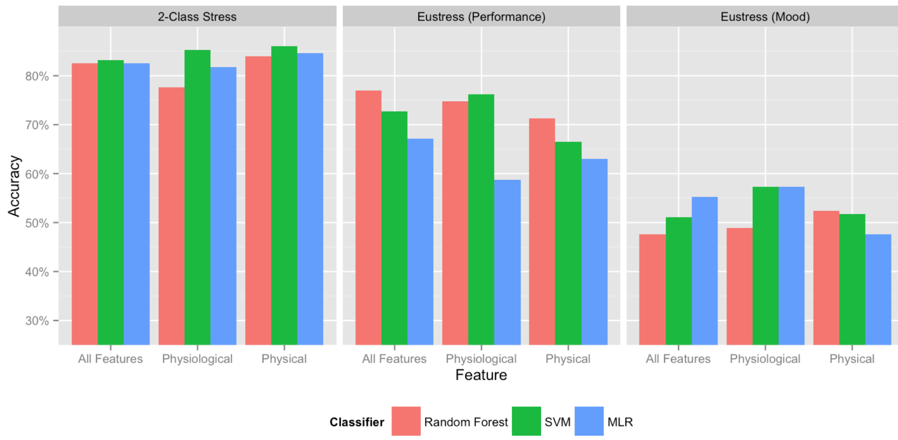


Figure 4.4: Classification Result

tively.

First of all, we perform inter-subject z-score normalization on the features in order to increase the generality of the model. Then we calculate the correlation matrix to eliminate redundant features, which has a coefficient greater than 0.75. Then the features were selected by exhaustive search with 10-fold cross-validation using Random Forest. Then we apply the Synthetic Minority Over-sampling Technique (SMOTE) [26] to the training data set to avoid over-fitting and deal with unbalanced data distribution.

For each classification problem, we partition all features into two subsets of features: physiological features, physical features. We tested every problem with any set of features before and after dimension reduction using PCA. The performance was evaluated under leave-one-subject-out cross-validation. For each learning method, the model was built using repeated cross-validation. We also fine-tuning the parameters of the model using a greedy approach in terms of accuracy.

4.8.1 General Stress Recognition

Prior to the eustress recognition, we tested our features on two-class general stress recognition with the above setting. Whereas the self-reported survey collected during the study was ranging perceived stress from one to five, then the value was normalized within the subject and the class "stressed" is defined by z-score ¿ 0 where the alternative is "not stressed".

On average, we achieved 82.75% accuracy and 96.93% recall for a two-class stress recognition problem using all features by applying PCA; More specifically, the best result was obtained by Support Vector Machine with 83.22% accuracy and 97.9% recall. For physiological features alone the accuracy reached 81.59% and 96.27% recall, where behavioral features obtained 84.85% accuracy and 99.03% recall. Our results show that we achieved competitive classification accuracy comparing to the state of the art.

4.8.2 Eustress Recognition

In this study, we have several assumptions: 1) eustress is the "right" amount of stress that improves performance [20]; 2) eustress associated with a positive feeling. Therefore, we define eustress in two ways: Eustress is the combination of moderate stress with high performance; and eustress is the combination of moderate stress with high mood. We consider moderate stress as 1 standard deviation away from 0 (both positive and negative direction) for z-score normalized stress. Mood as a subjective measure as stress was applied the same normalization technique as stress, where the distribution of performance is more consistent over different subjects, we considered high performance strictly greater than 3.

For eustress in terms of perceived performance, the accuracy achieved 67.13% with recall only 42.75% using all PCA features. For eustress defined by perceived mood, the accuracy has only 55.25% and recall 56.22% using physiological PCA features. It shows that the highly unbalanced data result in a poor recall rate on eustress classification.

4.9 Limitation

This work as a preliminary study of eustress has several limitations. Firstly, the sample size is limited to 6, where a larger-scale study is required for further study. Secondly, self-report surveys are considered as ground truth in this work, where it may suffer from inconsistent between different subjects. Lastly, the concept of eustress is unclear, where a more accurate model can be achieved by introducing a more concrete definition of eustress.

4.10 Conclusion

Existing work studied general stress in both laboratory and natural environments. However, there are only a few works that contributed to eustress since the concept has been proposed in the '70s. Our work-study the possibility of using ubiquitous sensing technologies for eustress recognition. We conducted a natural experiment and recruited 7 participants over 5 days. With an Android-based application developed, heart rate and smartphone usage data were collected to constructed a set of features using correlation and principal component analysis. We estimated the robustness of the features by three standard learning algorithms.

The result showed that heart rate variability, computer, and smartphone usage can be used for general stress classification as the literature suggested. The recognition accuracy also remains consistent over different learning algorithms. On the other hand, the accuracy of eustress in terms of performance is higher than mood, since perceived performance is highly related to the application used on smartphones and computers. However, the recall rates are low for both cases showing that the generality of the model still requires further study. The gap between general stress and eustress mainly due to the solid background of general stress that facilitated the feature engineering process and results in better classification performance.

Notice that the accuracy comparing to the existing work may seem quite low, however is reasonable since the previous studies assess mental stress in a rigorous laboratory or aggregated the data by days. In contrast, our natural experiment approach and finer granularity of time-series result in more noisy data which leads to a decrement of performance. We agreed that there is room for improvement, further study is required to achieve better recognition accuracy and recall rate.

To conclude, eustress is a widely accepted psychological phenomenon, should receive more interest from academia. As an initial study, our work provided encouraging results of eustress recognition, which can facilitate research on this problem in the near future.

Chapter 5

How Continuous Stress Affects Cognitive Performance: Towards a Computational Model

Cognition can be viewed as dynamical systems that are affected by various factors including mental stress and mental fatigue. Computationally modeling the dynamics of cognitive performance over time is crucial for improving productivity, and preventing accidents by varying external stressors. Previous work focused on descriptively identifying and discovering the general trends between predictive factors and performance. However, existing models remain at the theoretical level and cannot be applied for cognitive performance prediction on a prolonged cognitive task. We proposed a data-driven cognitive performance prediction model (called *CCSP model*) by leveraging psychological knowledge to computationally model the dynamic relationship between stress, fatigue, and cognitive performance over time. The proposed model was then trained and evaluated on the data collected in a rigorous laboratory experiment. It outperforms the state-of-the-art machine learning models in both Leave-one-participant-out and N-step ahead prediction settings. We discuss the implication of our work on improving learning and working productivity and preventing the occurrence of driving and occupational accidents.

5.1 Introduction

Stress is ubiquitous, and it has become part of our daily life. In 2016, the overall stress level in America was increased for the first time in nearly a decade, and there were 80% of Americans experienced at least one stress symptoms according to a recent survey [14]. As a public health crisis, stress has been shown as a risk factor to several diseases including cardiovascular disease, high blood pressure, diabetes, anxiety, and depression [29, 97, 33]. Apart from mental and physical health consequences, stress is also an important factor affecting cognitive performance. According to the definition given by the Oxford dictionary, cognition is "the mental process of acquiring knowledge and understanding including perception, intuition, and reasoning through experience and sense". It is usually assessed by measuring performance at specific tasks, which is *cognitive performance*. Being able to model the impact of stress on cognitive performance can provide insight into improving productivity, and preventing occupational accidents from happening by varying external stressors. Hence, monitoring stress to promote mental health and improve cognitive performance is a vital question in many stressful occupations such as nursing and the military.

One of the most widely accepted definitions of stress was introduced by J. Rabkin in 1976, where the stress is defined as "the organism's response to stressful conditions or stressors, consisting of a pattern of physiological and psychological reactions, both immediate and delayed" [111]. As a survival mechanism, stress hormones including cortisol and adrenaline are released under stressful situations. It improves the bodily and cognitive function to contend with threats and challenges. After that, it will resume to normal homeostasis being the equilibrium within the organism [121]. The relationship between stress and performance had been described as the inverted-U curve [145], which is a widely accepted law of psychology. Furthermore, the law has been extensively applied to identify positive stress, namely "eustress", which is the stress that improves performance. The law suggested that stress is beneficial to the performance until some optimal level is reached, after which performance will decline.

Apart from stress, excessive hours of work often leads to *mental fatigue* - "a psychobiological state caused by prolonged periods of demanding cognitive activity and characterized by subjective feelings of tiredness and lack of energy" [89] - that impairs performance. Previous work suggested that supervisory executive attentional functions are impaired during mental fatigue such that performance is affected more by other stressors [95]. In such a case, the interactive development of stress and performance in prolonged cognitive task is accompanied by mental fatigue. Unfortunately, the inverted-U relation only considers stress as an instantaneous effect on performance, which ignored its development over time. It is crucial to model stress together with mental fatigue to overcome this limitation. In [56], the authors proposed the cognitive-energetic framework that considers cognition as a dynamic process by regulating goals and actions as well as resource allocation. It remains at the theoretical level that is not applicable for performance prediction because of the inability to quantify those factors. Existing frameworks shared the same limitation since those works aim to identify the predictive factors and discover the general trends on performance, which is insufficient to predict the fluctuation of performance over time. As a dynamic process, a data-driven computational model is required to predict cognitive performance in daily life.

This study focuses on modeling the impact of continuous stress and mental fatigue on cognitive performance while performing prolonged cognitive tasks. By leveraging psychophysiological knowledge, we propose a data-driven Computational Continuous Stress Performance (CCSP) model to investigate the dynamic relationship between stress, mental fatigue, and cognitive performance over time. The model is trained and validated by data collected from a rigorous laboratory experiment carried out by 15 healthy participants. The proposed model outperforms the state-of-the-art machine learning models in both leave-one-participant-out and N-step ahead prediction setting. Although recent trends in machine learning lie on black-box approaches such as deep neural networks, we showed that a simple but interpretable model combines with domain knowledge could better predict psychological states.

There are several challenges in developing and validating the CCSP model. First, mental fatigue as an internal state cannot be measured objectively. Typical approaches measure mental fatigue by subjective self-report which suffered from individual bias. Another alternative approach is to approximate mental fatigue by the change of cognitive performance, which assumes mental fatigue is the only factor influence performance. In many cases, one is performing prolonged cognitive tasks under stress, and cognitive performance is affected by both mental fatigue and stress. Therefore, it is not practical to measure mental fatigue directly from the cognitive performance. To address this challenge, we model mental fatigue as a latent state of the Hidden Markov Model, in which the impact of fatigue is embedded and modeled indirectly by the change of cognitive performance.

The second major challenge is to design a suitable stressor to induce stress continuously without changing the operation or the difficulty of the cognitive task. Stress is usually induced by varying the operation or difficulty of the task in psychological studies. In our study, the task has to remain the same difficulty throughout the experiment such that performance at different points of time is comparable. For this sake, we adopt ambient traffic noise as an external stressor in the experiment, which is independent of the cognitive task.

The third major challenge is to handle the learning effect leading to an incomparable performance issue. When participants perform a cognitive task for a period of time, the learning process with experience accumulated tends to increase performance due to improved skills. Delivering task training to the participant may eliminate this effect, however, it is time-consuming and it may also introduce other biases in measuring cognitive performance because of the individual difference of learning ability. We applied the Advanced Trail Making Test (ATMT) - a standard cognitive performance evaluation task, to evaluate cognitive performance, which is robust enough with almost no learning effect as shown in the previous study [100].

To the best of our knowledge, our work is the first one to build a data-driven computational model on the basis of psychological theories, that measures the impact of continuous stress on performance taking mental fatigue into consideration. It fills the gap between psychological theory and computational modeling approaches, and pave the way towards the goal of applying stress detection on modeling the impact on human behavior.

The main contributions of this work are as follows:

- We leverage the knowledge of psychology to study a theoretical Continuous Stress Performance (CSP) model that comprehensively describes the dynamic relationship between stress, mental fatigue, and cognitive performance in a prolonged cognitive task.

- We build a Computational Continuous Stress Performance (CCSP) model based on the theoretical CSP model to quantitatively investigate the impact of stress and mental fatigue on cognitive performance over time. Each component in the CCSP model has the physical meaning corresponding to the psychological concepts and theories.

- The proposed CCSP model can accurately predict cognitive performance in real-time that could provide insights on improving productivity, stress management, and risk management for prolonged cognitive tasks. We also show that affective detection problems with limited training examples should apply domain knowledge in the model building phase. Therefore, simple but robust models could outperform the state-of-the-art machine learning models.

The rest of this chapter is organized as follows. In section 5.2, we review the related work of stress and cognitive performance. In section 5.6, we study the psychological CSP model and computerize it to a CCSP model. In section 5.5, we introduce data collection from a laboratory experiment and stress measurement methods. In section 5.7, we analyze the data obtained from the experiment and evaluated the performance of the proposed CCSP model. In the last section, we discuss the implication and limitations of our work, followed by future research directions.

5.2 Related Work

5.2.1 Stress Measurement and Detection

In recent years, computer scientists are interested in detecting and quantifying stress. By measuring stress intensity automatically, it helps better understand the stressors and manage stress levels in daily life. Either in the laboratory environment or in the natural environment, there is still a lack of a gold standard for stress measurement. Existing approaches assess mental stress levels based on the bodily response to the psychological state. Under the stimulation of stressor, the hypothalamic-pituitary-adrenal (HPA) axis increases the secretion of stress hormones including cortisol and catecholamine, which triggers a series of physiological changes in our body [118]. Typically, psychologists and physiologists detected the concentration of cortisol collected in blood, saliva, or urine samples to measure stress [124]. The detection of cortisol concentration takes place in professional institutions is very expensive and time-consuming such that it is inapplicable to monitor stress on a daily basis. As an alternative approach, recent studies rely on detecting the change of physiological signals caused by a sympathovagal imbalance in our body such as altered heart rate variability (HRV) [25, 87] and electrodermal activity (EDA) [31].

On this basis, computer scientists have made great contributions toward ubiquitous sensing of stress-related physiological responses. There are basically three types of contributions. First, the development of new sensors that allow non-intrusive and ubiquitous sensing of physiological signs. Second, the investigation on feature descriptors to better represent and distinguish the stressful physiological state from others. Lastly, building a supervised machine learning model to accurately predict stress. Currently, stress has been well detected in both laboratory and wild via physiological signals collected by wearable sensors [57, 84, 117, 136, 130], digital cameras [96] and hyperspectral imaging technique [28] and achieved over 90% reported prediction accuracy.

Apart from measuring the physiological signal, stress could also be detected by behavioral changes such as gesture and body posture [38], facial expression [125, 83], semantics of speech [72], mobile phone usage [22], social interaction [77], and hybrid approach using both physiological and behavioral data [73]. In a word, physiological and behavioral measures collected from ubiquitous sensing can be reliably applied for approximating mental stress. On top of stress detection, there are several works that studied the relationship between stress and human behavior such as typing pressure [54], email usage style [92], and sleep debt [93] by descriptive analytics. However, there is no existing work computationally modeling the impact of stress on human behavior for performance forecasting. In this work, we propose a computational cognitive performance prediction model to fill in this research gap.

5.2.2 Stress and Cognitive Performance

The relation between stress and cognitive performance has been studied by psychologists for a long time. There are numerous studies indicated that stress intensity significantly influence cognitive performance [21, 46, 53, 135]. In [131], the authors suggested that cognitive performance can be predicted under stress. The response of stress predicted better performance in the case of resources outweighing demands than the case of demands outweighing resources [131]. Alternatively, stress is associated with better performance if one is energetic enough to adapt to stress. Otherwise, mental fatigue will diminish the ability of stress tolerance leading to the decrement of performance [23, 55]. Psychologists focused on descriptive analytics that identifies and discovers significant predictors to describe the effect of stress and fatigue on performance as a trend or phenomenon. It is rare to find a computational model of such a trend or phenomenon to make it applicable for performance prediction in daily life.

In computer science, research on cognitive performance now is focusing on continuously assessing the cognitive function and finding out the condition of achieving the best cognitive performance through modeling and understanding the rhythms of cognitive function such as attention and alertness. By monitoring the usage of online activities including productivity software, Internet surfing, window switching, mail checking, and Facebook, etc. in the workplace environment, G. Mark et al. found that the peak of focus attention appears in mid-afternoon while the bored feeling is strongest in early afternoon [91]. S. Abdullah et al. adopted the response time of the Psychomotor Vigilance Task (PVT) as alertness measurement and predicted it by given smartphone usage behaviors to continuously and automatically monitor the cognitive performance [12]. However, existing studies fail to explain the mechanism of the observed trends and phenomena and usually ignored fatigue induced by the prolonged cognitive load which introduced bias in measuring cognitive performance. In this work, we go beyond the previous findings to investigate the effect of continuous stress on cognitive performance over time and computationally model it for cognitive performance prediction.

5.3 Continuous Stress Performance Model

In this section, we unify the theories regarding stress, fatigue, and cognitive performance and proposed a psychological Continuous Stress Performance (CSP) model that theoretically study the impact of continuous stress and mental fatigue on cognitive performance over time.

The relation between stress and performance can be described by the inverted-U hypothesis, also known as the Yerkes-Dodson Law [145]. As one of the oldest laws of psychology, it has been widely accepted and referenced. However, R. Hockey had criticized its correctness, suggested that the Yerkes-Dodson Law has oversimplified the relation between stress and performance [55, 48]. He argued that the original study was describing the link between discrimination learning and aversive reinforcement, it then being adopted to relate stress and performance which was misinterpreted. The decrement of learning performance under high-stress situations might be caused by the distracting effect of stress-inducing tasks or secondary anxiety effects instead of stress itself [55]. Moreover, noise, heat, and sleep loss made performance worse when those stressors enforced alone, but the joint effect of two stressors was opposite in some cases [55]. For example, sleep loss reduced body sensitivity to noise but the noise increased concentration when sleep loss, which eliminated performance decreasing. Therefore, there was no direct support to the proposed relation, and it persisted due to the failure of quantifying stress and performance respectively to the inverted-U axis.

Another limitation of the Yerkes-Dodson Law is that it is not considering the temporal effect of stress on performance. In the Yerkes-Dodson law, each stress intensity is associated with a constant predicted performance value. By performing the same cognitive task multiple times under different stress intensities, the relation between stress and performance forms the inverted-U shape [46]. The inverted-U relation keeps constant if stress remains at the same level regardless of the duration of the task. In the reality, cognitive performance fluctuated over time because of continuous cognitive demand even if stress remains at the same level. Therefore, the Yerkes-Dodson Law is not applicable for modeling the impact of stress on performance over time. Cognitive performance would change under prolonged stress and cognitive demand. Predicting cognitive performance by stress intensity without noticing the dynamic change under stress over time leads to imprecise results. In such a dynamic cognitive process, the relation between stress and performance is much more complicated than the inverted-U.

Continuously performing cognitive tasks induces mental fatigue as a result of prolonged stress and cognitive demand, which is a symptom of insufficient mental energy. This psychological state affects the ability to sustain stress and performance [23, 55]. J. Huxley defined mental energy as “the driving forces of the psyche, emotional as well as intellectual” [61], where H. Lieberman defined in a similar way as “the ability and willingness to engage in cognitive work”. It is one of the major attributes of performance [76]. In view of this, we adopt the concept of mental energy to describe the temporal effect of mental fatigue towards prolonged cognitive demand and stress in the following section. Prolonged cognitive tasks consume mental energy over time and gradually affect cognitive performance.

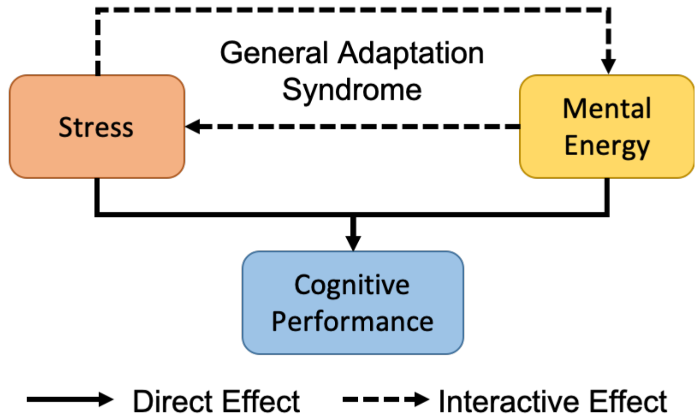


Figure 5.1: Continuous Stress Performance Model.

Under continuous stress, the human body consumes mental energy in order to adapt to the demand for change. While the stressor persists for long period until the mental energy drain, this reciprocal relationship triggers the state of exhaustion such that severe stress responses arose [34]. According to the General Adaptation Syndrome theory, our body consumes "adaptation energy" to resist cortisol secretion resulting in lower stress responses [23, 121]. In other words, "energy" as a limited resource is required during the process of stress adaptation. Lower stress responses can be observed while energy is sufficient to manage the stressor, vice versa. Therefore, continuous stress and mental energy have an interactive effect on each other and jointly affect cognitive performance over time. On the other hand, prolonged cognitive demand also affected the state of mental energy. We named the dynamic impact of continuous stress and mental energy on cognitive performance as CSP model as shown in Fig. 5.1. Prolonged cognitive tasks induced cognitive demands that consumed mental energy over time. Continuous stress and the amount of mental energy interactively affect each other, which jointly leads to the fluctuation of cognitive performance over time. In CSP model, mental energy serves as an intermediate variable carrying the physical meaning of the temporal effect of stress on cognitive performance.

5.4 Data Set

In this section, we will introduce our laboratory experiment design for physiological stress response data and cognitive performance data collection.

5.4.1 Design of Laboratory Experiment

A laboratory experiment was conducted to capture the temporal relation between stress, mental energy, and cognitive performance. Fifteen healthy non-smoking adult participants of both genders (three females), different ages (22 to 30, average 26.53, variance 5.41), multiple races (one Indian, one Italian, and thirteen Chinese) participated. The experiment was conducted in a completely soundproofed enclosed room, and the temperature was kept constant at 22 degrees Celsius. Participants were sitting in front of a laptop computer and wearing the Empatica E4 wristband on the wrist of their dominant hand, which continuously collected their electrodermal activity (EDA), blood volume pulse (BVP), inter-beat interval (IBI), and 3-axis accelerometer data to measure stress response during the experiment. The physiological stress response data and cognitive performance data would be collected during the experiment for training and testing the CCSP model later.

Laboratory Experimental Tasks

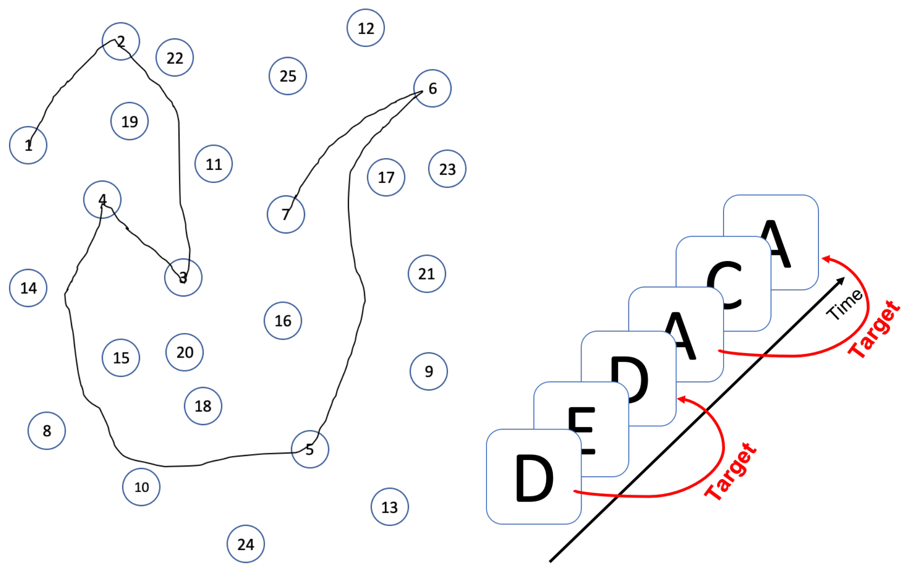
We alternately conduct the Advanced Trail Making Test (ATMT) and the N-back test in the experiment. The ATMT is a cognitive test to "evaluate the level of selective attention and spatial working memory regardless of the subject's intelligence quotient or experience" [100]. The test begins with 25 dots with numbers (1 to 25) on the screen. Participants were asked to connect the dots in sequential order as quickly as possible while avoiding to connect the wrong numbers. The dot will disappear when the participant connects it in the right order. Then a new dot with a new number will be placed on the screen at a random position in order to ensure that there are 25 dots at any time. The new number is generated in sequential order starting from 26. For instance, when the participant clicks the dot with number 1, it will disappear and a new dot with number 26 will be generated at the same time. An example of the ATMT shows in Fig. 5.2a. To measure cognitive performance, the number of corrected and uncorrected connection, mouse movement distance, and the time interval between connecting two dots were recorded.

The N-back test is one of the most popular experimental paradigms to evaluate working memory. It is a continuous task in which a series of stimuli present consecutively, and the participant determines whether the current stimuli are the same as the stimuli n trial before. In the 2-back test, a series of stimuli in one of six characters ('A', 'B', 'C', 'D', 'E', and 'F') will be displayed to participants one by one. Participants are asked to press key H on the keyboard when the current appearing character is the same as the one in N trials before as shown in Fig. 5.2b. Every character appears 800 ms followed by a 400 ms blink page to determine the result (correct, incorrect, or missing). There is 50 stimulus presented in every minute with 17 triggered stimuli that participants need to respond. The sequence of stimulus is generated randomly and then fixes to everyone. Therefore, every participant has the same series of stimuli to assure the difficulty among different participants is the same. The number of correct, incorrect, and miss are recorded as well as the response time of correct and incorrect.

In our experiment, we adopt participants' performance in ATMT to quantify their cognitive performance instead of the performance in the 2-back test because the performance in the 2-back test is much easier to be influenced by the increase of learning with experience than ATMT. In our pilot study, nine participants are hired to perform the 2-back test for uninterrupted 20 minutes. Though they had trained before the experiments, their performance kept on decreasing in the first twelve minutes because of 2-back test successfully consumed their mental energy and induce fatigue, but their performance increased in the last 5 to 7 minutes. This phenomenon is because of the learning process with experience accumulated from the test. Even though participants hardly remember some characters at the end of the test, they could still make correct responses under conditioned reflexes of their memory, which is the limitation of using the n-back test to evaluate cognitive performance. Comparing with the N-back test, the ATMT is more robust to learning with experience so that the performance in ATMT would keep on decreasing under mental fatigue [100]. The limitation of ATMT is that the performance decreases slowly under fatigue and the number of incorrect connections is very small. Therefore, we adopt the 2-back test to consume participants' mental energy and induce fatigue, then measure their cognitive performance in ATMT so that we could get rid of the learning effect and the performance incomparable issue.

Laboratory Experimental Protocol

The experiment consists of two sessions: neutral testing and stress testing. In neutral testing, the cognitive workload would induce mental fatigue, and it is unavoidable that a small amount of stress would be induced by the cognitive demand simultaneously. The neutral testing aims to collect data for exposing the effect of mental fatigue on cognitive performance. In stress testing, an external stressor, baby crying noise, would be applied to the subject to induce extra stress during performing the same tasks as in neutral testing. Stress will be not only induced by cognitive demand but also induced by the sensory load. Because the cognitive tasks in both neutral



(a) (b)

Figure 5.2: Examples of the experimental tasks. (a) is the example of the Advanced Trail Making Test and (b) is the example of the 2-back test.

testing and stress testing are exactly the same, ideally the mental fatigue induced by the cognitive workload in both testings for a given subject would be the same too. The stress testing phase aims to collect data for exploring the changes in cognitive performance over time under external stressors comparing with the neutral phase. Participants are required to visit the lab twice for the two sessions. We randomly selected 7 participants to take the stress testing first and the rest of 8 participants would take the neutral testing first. There is at least 3 hours gap between the two sessions in order to ensure the participant has recovered from both stress and fatigue. All participants were instructed to avoid any kind of caffeine within twelve hours before the experiment.

The participants will wait outside the experimental room upon arrival, where the observer explains the experiment protocol and the experimental tasks to them. Then each participant will have 7 minutes of training of the tasks before the experiment begin (2 minutes ATMT; 5 minutes 2-back test). After training, the participants wear the E4 wristband and following the observer to the experimental room. Physiological data collection will launch once the E4 wristband has been worn. Participants are asked to sit in front of a laptop computer and the experiment begins once the observer leave and closes the door of the experimental room. The first part is a 5-minute rest, where the physiological data were recorded to capture the baseline for each participant. After the rest session, the participant fills in his demography information, and then start performing three cognitive tasks sessions continuously.

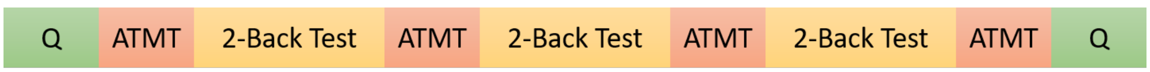


Figure 5.3: The schedule of cognitive tasks session. Q is questionnaire session.

Each cognitive task session contains 4 rounds of ATMT (1 minute per round) and 3 rounds of the 2-back test (2 minutes per round) alternately as shown in Fig. 5.3. Before and after each cognitive task session, participants are asked to self-report their stress level and fatigue level on a scale of 1 to 5 from lowest to highest. In order to make the performance measurement comparable between stress testing and neutral testing, we fix the appearing position of each dot for all participants, which means that all cognitive tasks including ATMT and 2-back tests are exactly the same in stress testing and neutral testing for all participants. The only difference between neutral testing and stress testing is that baby crying noise is played during stress testing. Using baby crying as an external stressor aims to increase the sensory load, and thus increasing psychological stress without changing the operation or difficulty of the task.

5.5 Data Observation

We collected data from 28 participants where two were excluded due to software issues that causing incomplete data. From the remaining 26 participants (M:17;

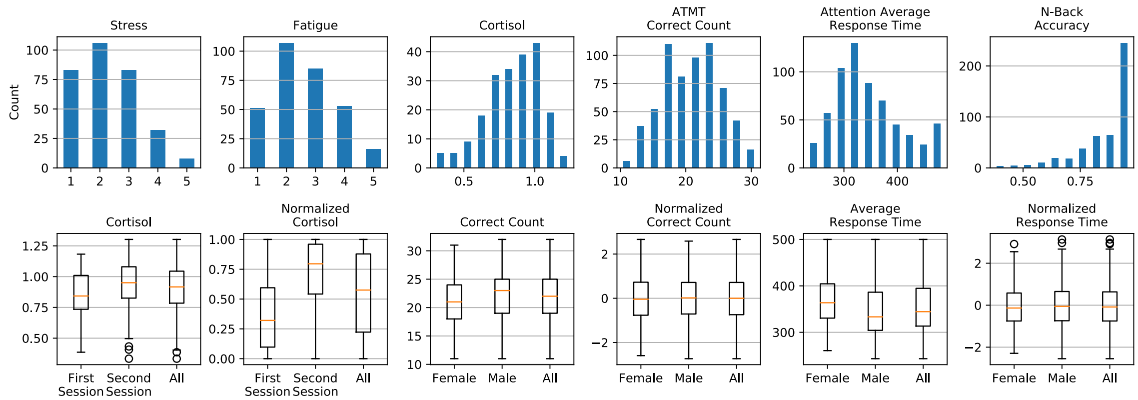


Figure 5.4: An overview of the data distribution.

F:9), we collected 208 self-reports and Cortisol samples, 624 ATMT and Attention task records, and 468 N-back task records.

An overview of the data is shown in figure 5.4. We notice that both self-reported stress and fatigue have skewed distribution towards the lower values, while 62.5% and 58.7% of the surveys report their level of stress and fatigue were less than 3 accordingly. There are only 13.5% and 22.1% of the surveys report their level of stress and fatigue was above 3, and those were reported by only 15 participants. For the Cortisol readings, we can see that the normalized Cortisol shows a more significant difference between the first and the second sessions with a p-value < 0.001. It suggests that the experiment successfully induce stress and fatigue over time that results in higher Cortisol. We also find that individual differences, as well as gender differences of performance, exist among the participants. Therefore, we normalized the self-reports, Cortisol, and performance data for the following analysis to mitigate the effect of the individual differences.

First, we study the correlation between self-report and Cortisol. There exists a high correlation between self-reported stress and fatigue where the coefficient is 0.61 with a p-value < 0.001. Surprisingly, there is no significant correlation for either stress and fatigue to Cortisol. It might be caused by the fact that perceived stress and the physiological response to stress might be different, which is potentially affected by mental energy as discussed earlier. In general, Cortisol increases over time even across sessions but the self-report stress and fatigue are decreased after the rest session. Then we investigated the effect of the session order (whether participant listening to noise in the first session) and the result shows that participants having the stress session first will drastically increase the Cortisol in the later neutral session even after 15 minutes recovery, the difference comparing to neutral session first is significant with p-value < 0.05. Therefore, we applied the linear regression model on time and the order of the session to predict the normalized Cortisol. The R-square of the model reaches 0.219 which is comparatively good to the self-reports (stress: 0.142; fatigue: 0.156).

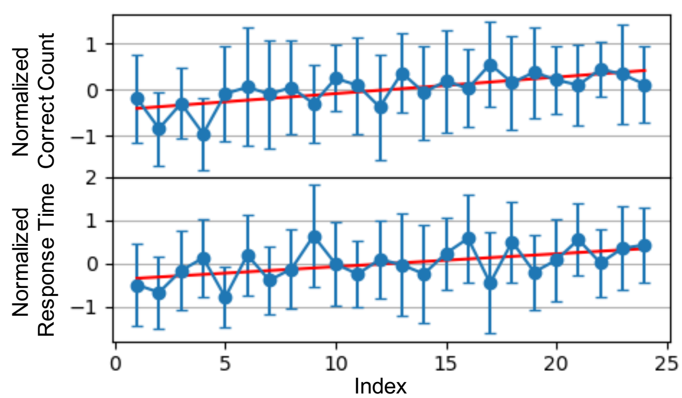


Figure 5.5: Trend of the cognitive performance in terms of correct count and response time.

For the performance, we notice that both normalized correct count and normalized response time increase overtime. It shows that the participants are less responsive when fatigue accumulated, but the performances on cognitive tasks are increasing which might be caused by more concentration as time goes. We found correlation of

0.2 between Cortisol and correct count with p-value < 0.005, 0.15 between Cortisol and response time with p-value < 0.005, 0.14 between self-report stress and response time with p-value < 0.001, and 0.098 between self-reported fatigue and response time with p-value < 0.05. This finding in line with our expectation where performance is related to mental state, which can be partially expressed by stress and fatigue.

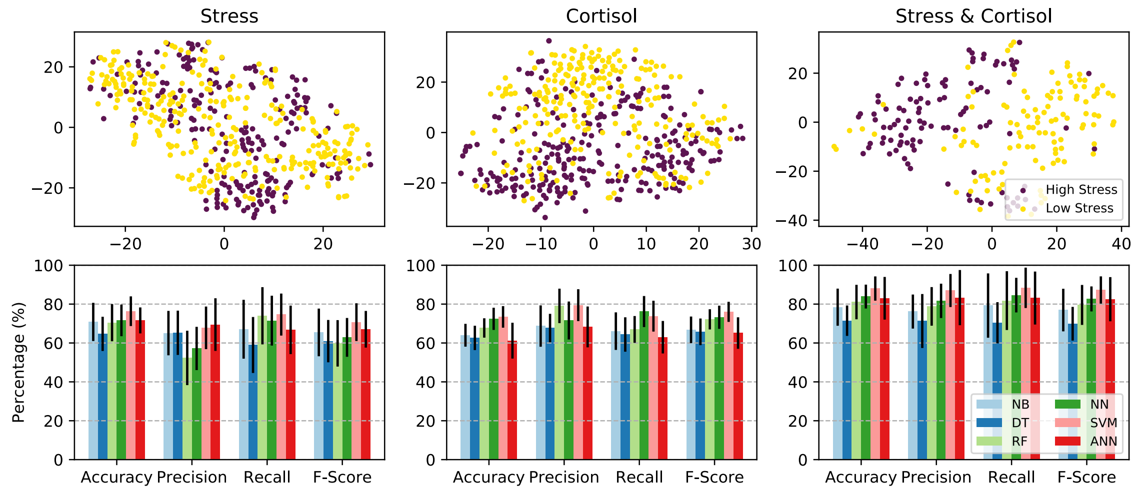


Figure 5.6: T-SNE of stress defined by self-reported stress, Cortisol, and the combination of the two. High stress is where the normalized value of the parameter is greater or equal to 0.5 and values that lower than 0.5 are low stress.

We then study if the collected physiological data can estimate mental stress. We extracted statistical features including minimum, maximum, median, and standard deviation for each of the physiological data. There are more on heart rate variability as suggested by previous work [73, 124, 57]. The full list of features can be found in table 5.1. Figure 5.6 shows T-SNE and the detection result of the extracted features. We defined stress according to three criteria: self-report stress, Cortisol, and the combination of the two. The self-report stress and Cortisol were scale between 0 and 1 for each participant, and we define *high stress* as the value ≥ 0.6 where *low stress*

Table 5.1: Summary of extracted features for stress estimation.

Modality	Features
Blood volume pulse	mean, standard deviation, minimum, maximum, range, first percentile, third percentile, and median of blood volume pulose
Electrodemo activity	mean, standard deviation, minimum, maximum, range, first percentile, third percentile, and median of skin conductance
Motion	mean, standard deviation, minimum, maximum, range, first percentile, third percentile, and median of motion intensity
Body temperature	mean, standard deviation, minimum, maximum, range, first percentile, third percentile, and median of body temperature
Heart rate measure	mean, standard deviation, minimum, maximum, range, first percentile, third percentile, and median of heart rate and heart rate variability; RMSSD, PNN20, PNN50, SD1, SD2, SD1SD2, VLF, LF, HF, total power, LF ratio, HF Ratio, and the ratio of LF and HF

as the value ≤ 0.4 in order to avoid fuzzy data. This results in some data that do not fall into any of the class and those have been discarded in the classification process. For self-report, it retains 81.25% (507) of the data, 81.41% (508) for cortisol, and 34.78% (217) for the combination of the two. We can see that for both self-report stress and Cortisol, the high stress are more concentrated but still mixed with those instances of low stress. We notice a clearer differentiation using the combination of both self-report and Cortisol.

We then applied conventional classification models to recognize low and high stress defined by different criteria. First of all, the features were normalized between 0 and 1 for each participant. Then we applied the Principle Component Analysis (PCA) to reduce the number of dimensions of the features. We choose 51 as the number of dimensions that retain more than 99% of the variance. Then we evaluate the classification result on the conventional classification models including Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), Nearest Neighbor (NN), Support Vector Machine (SVM), and Artificial Neural Network (ANN). The result was evaluated by 10-fold cross-validation in the lower part of figure 5.6. It shows that the performance of classifying stress is way better than a random guess. For the combination of both self-report and Cortisol, the F-score achieved up to 85.8%. It complies with the expectation that physiological data can be applied for mental stress estimation. However, as shown above, the relationship between mental stress and cognitive performance is non-linear and vastly affected by the duration of the task. Therefore, a model that takes time effect into consideration is needed, and the effect to responsive and cognitive ability are different as shown from the above study.

In this section, we found that in general, there are correlations between stress and cognitive performance. There is also a significant temporal effect of accumulated stress on cognitive performance. However, it is unable to predict the change of performance for a specific individual using these factors since the effect of stress and duration to mental energy is unclear, and it is difficult to measure mental energy objectively. While we show that physiological signals can estimate mental stress fairly well, stress and duration of the cognitive task play an important role in mental energy. A model that takes the stress and the temporal effect of stress on cognitive performance is needed. In the next section, we introduce a computational model that considers the mental state as a hidden state, which takes the physiological signal as input to predict cognitive performance by taking the temporal effect into consideration.

5.6 Computational Continuous Stress Performance Model

On the basis of the proposed CSP model, we developed the Computational Continuous Stress Performance (CCSP) model which quantifies the dynamic impact of continuous stress and mental energy on cognitive performance over time. CCSP model could be used to predict cognitive performance by given a time series of stress responses. The following definitions are provided to formulate the problem formally. In the following session, performance is regarded as computed cognitive performance, which is an objective measure of effectiveness and efficiency in performing a cognitive task, in order to provide objective measure towards the vague concept.

Problem 3 (Successive Similar Patterns Mining) *Given a Time series of stress measurement $S = \{s_t | t = 1, 2, ..., n\}$. Assume that the mental energy e_t is determined by e_{t-1} and s_t: $e_t = g(e_{t-1}, s_t)$, and performance y_{t_1} and y_{t_2} are independent for $t_1 = 1, 2, ..., n$ and $t_2 = 1, 2, ..., n$ but $t_1 \neq t_2$. The objective is to build a computational model to predict performance y_t respected to s_t and e_t*

The formulation begins with the segmentation of the time series, which was discretized into T time interval with the same length. In CSP model, mental energy is solely affected by stress and the duration of cognitive demand. After discretization, mental energy e_t at time $t = 1, 2, ..., T$ is the consequence of current stress s_t acting on surplus energy e_{t-1} at $t-1$ so that e_t is determined by e_{t-1} and s_t. The initial mental energy is represented by e_0. As mentioned in CSP model, performance y_t at time t is affected by both stress response s_t and the amount of mental energy e_t, which means that the performance at different time frames is independent. Then, our research problem can be formulated as shown in the following box. In our study, mental energy is one of the dependent variables to performance, so it cannot be measured

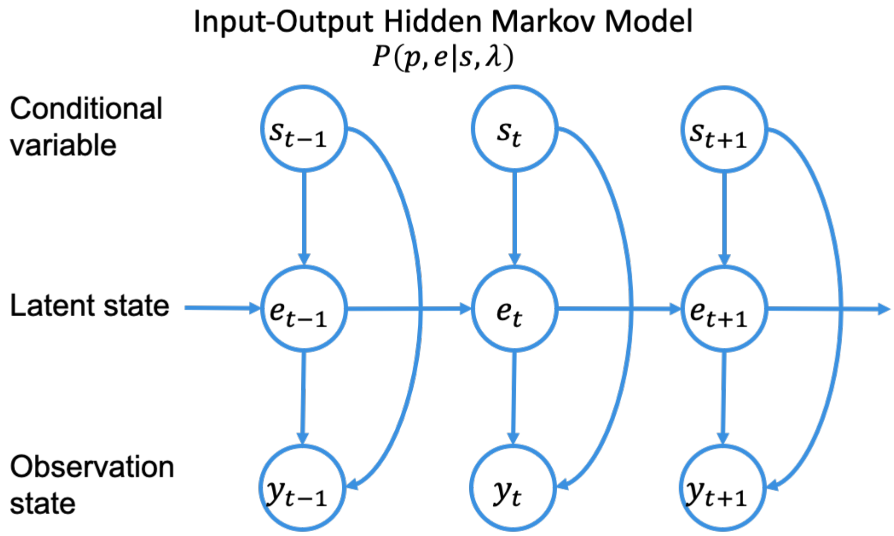


Figure 5.7: Computational Continuous Stress Performance Model.

by performance indicators directly as what psychologists had done before.

The most direct approach to solve the problem is to first collect large amounts of stress response data and performance data, then adopt an existing machine learning algorithm to predict the performance by using features extracted from stress response data. The machine learning algorithm here just serves as a black box, which fails to explain the psychological relation between continuous stress, mental energy, and performance, but the prediction results show that the classical classifiers such as random forest, k-nearest neighbors, and deep neural network cannot work well (please refer section 5.7 for more details). One alternative approach is that using the collected stress response, performance data at each time frames to fit a surface representing the function of performance respected to stress responses and time. However, it is very difficult to collect large amounts of stress response data from a single person in fine-grained, which covered a wide range of stress intensity. The individual differences in stress responses will introduce lots of errors further increasing the difficulty of model fitting.

Although the real value data of mental energy cannot be collected directly, we seek to build a computational model that can still retain the joint impact of stress and mental energy on performance over time. Therefore, we formulate the CCSP model in the view of probability. Our goal shifts to estimate y_t which maximizes the posterior likelihood $P(y_t|s_t, e_t)$. Since e_t is dependent upon e_{t-1} but independent of e_{t-l}, $l = 2, 3, ..., t-2$ and current performance y_t is independent of the previous performance, the problem satisfies the Markov assumption. Better yet, the mental energy e_t can be modeled as the latent state of Markov chain, which can only be partially observed by cognitive performance y_t under input condition s_t. It perfectly solves the problem that lacks objective mental energy data but still retains its impact on performance. Integrating the interaction between y_t, s_t and e_t at time $t = 1, 2, ..., T$, we get a directed graph model showed in Fig. 5.7, which could be quantitatively modeled by the Input-Output Hidden-Markov Model (IOHMM). The IOHMM first estimates the probability distribution of mental energy e_t at current time frame t by given current input probability of stress s_t and the probability distribution of mental energy e_{t-1} at last time frame, and then calculates the probability of current performance y_t under the condition of s_t and e_t. In each time step t, the three vertices s_t, e_t, and p_t form a Bayesian Network that determine the dependent probability distribution between them.

5.6.1 CCSP Model Training and Prediction Algorithms

CCSP model could be considered as a combination of the standard Hidden Markov Model and the Bayesian network. It represents the condition probability distribution $P(y_1^T|s_1^T)$ of observation sequences $y_1^T = y_1, y_2, ..., y_T$ on the condition of input sequences $s_1^T = s_1, s_2, ..., s_T$, where T is the length of sequences. In this chapter, we adopt the method proposed in [19] to train the optimal parameters of CCSP model including initial state probability distribution π, state transition probability distribution A, and observation probability distribution B for both latent states and output observation, which is the expectation-maximization (EM) based algorithm. It maximizes the condition likelihood $P(y_1^T|s_1^T)$, which is similar to the Baum-Welsh algorithm used to train a standard hidden Markov model.

We first introduce the EM algorithm for training CCSP model. Let us define $e_1^T = e_1, e_2, ..., e_T$ as the hidden state mental energy sequences and $\lambda(\pi, A, B)$ is the parameters of CCSP model that need to be trained, where $\pi = [\pi_1, \pi_2, ..., \pi_i, ..., \pi_n]^T$ is initial state probability vector and $\pi_i = P(e_1 = i|y_1), i = 1, 2, ..., n$ represents the probability of being in state i at time $t = 1$ with output observation y_1; $A = [a_{ij,t}]_{n\times n}$ is state-transition probability matrix at time t and $a_{ij,t} = P(e_t = i|e_{t-1} = j, s_t), i = 1, 2, ..., n, j = 1, 2, ..., n$ represents the transition probability from state i to state j with input s_t; $B = [b_{j,t}(k)]_{n\times m}$ is the observation probability matrix where $b_{j,t}(k) = P(y_t = k|e_t = j, s_t), k = 1, 2, ..., m$ is the probability of obtaining observation $y_t = k$ in state j at time t.

Similar to the Baum-Welsh algorithm, we first calculate the expectation of parameter $\hat{\lambda}$ under current hidden state and get the Q function, then we maximize the Q function to estimate the optimal parameter λ using maximum likelihood estimate (MLE). Let $\mathbf{E}$ is the set of hidden state e_1^T, $\mathbf{Y}$ is the set of output observation y_1^T, and $\mathbf{S}$ is the set of input data s_1^T, then the Q function can be calculated by Eq. 5.1 where $\hat{g}_{i,j} = P(e_t = i|s_1^T, y_1^T, \hat{\lambda})$, $\hat{h}_{ij,t} = P(e_t = i, e_{t-1} = j|s_1^T, y_1^T, \hat{\lambda})$, and $z_t = [z_{1,t}, z_{2,t}, ..., z_{i,t}, ..., z_{n,t}]^T$ is a vector of indicator variables that $z_{i,t} = 1$ if $i = t$, otherwise $z_{i,t} = 0$.

$$
\begin{aligned}
Q(\lambda, \hat{\lambda}) &= E[\log P(\mathbf{E}, \mathbf{Y}|\mathbf{S}, \lambda)|\mathbf{Y}, \mathbf{S}, \hat{\lambda}] \\
&= \sum_{t=1}^{T} \sum_{i=1}^{n} E[z_{i,t}|s_1^T, y_1^T, \hat{\lambda}] \cdot \log P(y_t|e_t = i, s_t, \lambda) \\
&+ E[z_{i,t}, z_{j,t-1}|s_1^T, y_1^T, \hat{\lambda}] \cdot \log P(e_t = i|e_{t-1} = j, s_t, \lambda) \\
&= \sum_{t=1}^{T} \sum_{i=1}^{n} \hat{g}_{i,j} \log P(y_t|e_t = i, s_t, \lambda) \\
&+ \sum_{j=1}^{n} \hat{h}_{ij,t} \log P(e_t = i|e_{t-1} = j, s_t, \lambda)
\end{aligned} \tag{5.1}
$$

$\hat{g}_{i,j}$ and $\hat{h}_{ij,t}$ can be easily calculated by the forward-backward algorithm with parameter $\hat{\lambda}$ using Eq. 5.2 to Eq. 5.6 where $\alpha_{i,t}$ is obtained by the forward pass algorithm and $\beta_{i,t}$ is computed by the backward pass algorithm.

$$
\hat{g}_{i,j} = P(e_t = i|s_1^T, y_1^T) = \frac{1}{L}\alpha_{i,t}\beta_{i,t} \tag{5.2}
$$

$$
\hat{h}_{ij,t} = \frac{1}{L} P(y_t|e_t = i, s_t)\alpha_{j,t-1}\beta_{i,j} P(e_t = i|e_{t-1} = j, s_t) \tag{5.3}
$$

$$
\begin{aligned}
\alpha_{i,t} &= P(y_1^T, e_t = i|s_1^T) \\
&= P(y_t|e_t = i, s_t) \sum_j P(e_t = i|e_{t-1} = j, s_t)\alpha_{j,t-1}
\end{aligned} \tag{5.4}
$$

$$
\begin{aligned}
\beta_{i,t} &= P(y_1^T|e_t = i, s_1^T) \\
&= \sum_j P(y_{t+1}|e_{t+1} = j, s_t) P(e_t = j|e_{t-1} = i, s_{t+1})\beta_{j,t+1}
\end{aligned} \tag{5.5}
$$

$$L = P(y_1^T|s_1^T) = \sum_i P(y_1^T, e_T = i|s_1^T) = \sum_i \alpha_{i,T} \tag{5.6}$$

In maximization step, we use gradient ascent method to increase the value of objective Q function in Eq. 5.1 and search the best parameter λ. Since the first term of objective Q function depends on initial state probability vector π and state-transition matrix A, and the second term only depends on the observation probability matrix B, we separate the objective Q function into two sub-functions and optimize them alternately, which will simplify the primal problem and reduce computational cost.

The prediction of discrete performance y_{T+1} at next time step can be formulated as searching a discrete value $p_{T+1} = k$ from the performance candidate set $\{k|k = 1, 2, ...\}$ to maximize the likelihood $P(y_{T+1}|y_1^T, s_T, \hat{\lambda})$ with given performance observation sequence y_1^T, the stress response s_{T+1} and well trained CCSP model $\hat{\lambda}$. The objective conditional probability is calculated by Eq. 5.7.

$$P(y_{T+1}|y_1^T, s_T, \hat{\lambda}) = \frac{P(y_{T+1}, y_1^T|s_T, \hat{\lambda})}{P(y_1^T|s_T, \hat{\lambda})} \tag{5.7}$$

Because y_{T+1} is independent of y_1^T, the objective probability $P(y_{T+1}|y_1^T, s_T, \hat{\lambda})$ is proportional to $P(y_{T+1}, y_1^T|s_T, \hat{\lambda})$. For each possible performance observation $y_{T+1} = k, k = 1, 2, ...$, the likelihood $P(y_{T+1} = k, y_1^T|s_T, \hat{\lambda})$ can be obtained by the forward pass algorithm and the one with maximum likelihood would be selected as the predicted output observation that is the predictive cognitive performance y_{t+1} at next time stamp.

$$\text{precision}_i = \frac{TP_i}{TP_i + FP_{ij} + FP_{ij}}$$

$$\text{recall}_i = \frac{TP_i}{TP_i + FP_{ji} + FP_{ji}}$$

(a)

Prediction \ Truth	1	2	3
1	TP_1	FP_{12}	FP_{13}
2	FP_{21}	TP_2	FP_{23}
3	FP_{31}	FP_{32}	TP_3

1 : Increasing
2 : Decreasing
3 : Constant

(b)

5.7 Evaluation

In this section, we provide the result of experimental evaluation on cognitive performance prediction. We first introduce the metrics adopted for the evaluation. Then, we discuss two supervised learning schemes to validate our proposed CCSP model and comparing the result with several baseline methods.

5.7.1 Evaluation Metric

In this evaluation, the target classes are represented as $i = 1, 2, 3$ that refers to different states of cognitive performance: increasing, decreasing, and remain constant. We define the true positive and false positive corresponding to different truth and prediction classes as shown in Figure 5.8b. The precision and recall of each class are calculated with the equation as shown in Figure 5.8a.

where $j = 1, 2, 3$ and $j \neq i$. We also use F1-score as a combined metrics to evaluate performance in predicting each class label i, which is defined in Eq. 5.8.

$$F1_i = 2 \cdot \frac{\mathrm{precision_i} \cdot \mathrm{recall_i}}{\mathrm{precision_i} + \mathrm{recall_i}} \tag{5.8}$$

5.7.2 Prediction Analysis

In this section, we discuss two supervised learning schemes for predicting cognitive performance. The first scheme is the person independent prediction using 10-fold cross-validation. The other one is the n-step ahead prediction which utilizes the data at the early trails as a training set and predicts the future performance in the later trails. Note that we do not train an individual model to test person dependent prediction because each participant only has 12 data points collected in the neutral session and another 12 data points collected in the stress session, which is too small to train an effective model.

Among the three cognitive tasks, we selected the performance of ATMT to evaluate the prediction result. The reason is that the correctly connected numbers (or correct count) of ATMT have the most balanced distribution as shown in Section 5.5. Also, the performance of ATMT is less affected by the skill of the participant and therefore not related to the level of education. Particularly, we measure the amount of correct count within a given time. The precision of the ATMT task is defined as the correct count divided by both correct and incorrect count. In our dataset, we notice a small incorrect count that recorded an average 0.9558 precision with a standard deviation of 0.0722. It means that adopting average value to predict the precision already achieve negligible root-mean-square error. Therefore, we focus on the prediction of the change of correctly count. The average change of correct count over all participants is -0.035, and the standard deviation is 3.4. Therefore we define the three classes by the difference of correct count as $>= 3$ for increasing, $<= -3$ for decreasing, > -3, and < 3 for remain constant.

Table 5.2: Comparison of results in person independent prediction defined by the differences of correct count in ATMT task.

Model	Precision	Recall	F1-Score
KNN	40.38%	35.97%	37.45%
NN	31.92%	50.14%	38.12%
RF	32.28%	**55.97**%	40.69%
SVM	32.28%	**55.97**%	40.69%
CCSP	**42.86**%	43.19%	**42.03**%

Table 5.3: Comparison of results in person independent prediction defined by the normalized differences of correct count in ATMT task.

Model	Precision	Recall	F1-Score
KNN	46.76%	44.31%	45.06%
NN	42.10%	62.15%	49.94%
RF	42.02%	**64.44**%	**50.73**%
SVM	42.02%	**64.44**%	**50.73**%
CCSP	**50.34**%	51.39%	50.24%

Person Independent Prediction

For the person independent prediction, we divide the subject's data into 10 partitions randomly. Each partition has two to three subject's data in both stress and neutral session. For each fold, one of the partitions is selected as the testing set and the rest of the data is used as the training set. The weighted average of F1-score, precision, and recall are then calculated and averaged over the 10 folds. We choose the number of hidden states by an exhaustive search that minimizes the training error for the CCSP.

The result in Table 5.2 shows the prediction performance on the differences of correct count in the ATMT task. The RF and SVM achieved the same result, while the KNN and NN models also achieved a similar result. Generally, CCSP achieved better performance comparing with the baseline methods. It has significantly higher precision suggesting that it is more accurate to predict the correct classes. The RF and SVM were highly affected by the skewed distribution that predicting most samples as constant, which is the dominant class and therefore recorded a higher recall rate. In order to reduce the individual differences, we also measure the performance on predicting the normalized differences of correct count in ATMT. The differences in the correct count are normalized within each session for each participant. The result in Table 5.3 shows that the performance is significantly better, and again the CCSP model achieved the highest precision. However, the F1-Score is slightly lower than the RF and SVM.

N-Step Ahead Prediction

The N-step ahead prediction aims to predict the future n trails performance given the previous performance. Since each participant has 12 trails in each session, we divide the dataset by the index of trails. For example, 1-step ahead prediction leverage the first 11 trails data of all participants as a training set and predicts the performance of the last trail. In this evaluation, we adopt the same approach in the last section to determine the number of states used in CCSP.

The result is shown in Table 5.4 and the best performance of each evaluation are in bold. It suggests that the performance of CCSP consistently outperforms the other baseline methods. Again to mitigate the effect of individual differences in performance variation, we evaluated the prediction result on the normalized difference of performance and the results are shown in Table 5.5. Significantly better performance can be observed compared with the raw performance differences and the performance is also consistently superior to the other baseline methods.

Table 5.4: F1-Score in predicting the differeces of correct count in N-step ahead prediction.

Model	1-step	2-step	3-step	4-step	5-step	6-step	7-step	8-step	9-step
KNN	37.76%	45.04%	34.69%	35.45%	37.26%	40.39%	39.06%	**39.30**%	39.15%
NN	42.74%	42.47%	43.53%	7.08%	39.93%	38.43%	36.06%	34.20%	13.75%
RF	42.21%	42.21%	41.92%	**41.99**%	39.76%	38.29%	36.12%	34.19%	34.53%
SVM	42.21%	42.21%	42.21%	42.21%	39.93%	38.43%	36.12%	34.14%	34.53%
CCSP	**50.85**%	**47.83**%	**44.86**%	41.42%	**41.61**%	**42.30**%	**39.60**%	39.20%	**41.20**%

Table 5.5: F1-Score in predicting the normalized differeces of correct count in N-step ahead prediction.

Model	1-step	2-step	3-step	4-step	5-step	6-step	7-step	8-step	9-step
KNN	41.16%	44.44%	38.57%	43.00%	45.68%	46.88%	44.91%	45.99%	**48.60**%
NN	49.86%	51.70%	**52.51**%	52.87%	52.19%	49.68%	47.23%	35.40%	9.82%
RF	49.28%	51.70%	**52.51**%	52.92%	52.19%	49.68%	47.23%	45.41%	45.84%
SVM	49.28%	51.70%	**52.51**%	52.92%	52.19%	49.68%	47.23%	45.41%	45.84%
CCSP	**59.69**%	**56.73**%	48.87%	**53.77**%	**55.70**%	**52.46**%	**49.88**%	**51.90**%	47.77%

5.8 Discussion and Conclusion

Continuous stress and mental fatigue are important factors affecting cognitive performance. In the CHI community, stress detection using ubiquitous technology has been well achieved in the past few years. However, it remains at the level of sensing and yet applied for real-world applications. Our work combines psychological theories and models to create the composite CSP model considering the dynamic relationship between stress, fatigue, and performance. We then transform it into a data-driven computational model for cognitive performance prediction. The model has achieved better prediction accuracy compared with the state-of-the-art machine learning techniques. The model is trained and validated using data collected from a rigorous laboratory experiment carried out by 28 healthy participants. Because there are numerous studies that examined the relationship between these variables, our work focuses on the quantification of such relation from the collected data. The resultant model is capable to predict the change of cognitive performance given a series of physiological measurements as an approximation of stress. It is the first step towards the goal to apply stress detection on modeling the impact of stress on human behavior.

By analyzing the collected data, we replicated the results of previous studies, confirmed that physiological features are statistically significant in terms of predicting self-reported stress and fatigue. Although ambient noise as an external stressor may not significantly affect physiological stress responses, we observe that performance in searching task are generally higher while participants are hearing a noise in the background. This finding is also supported by previous studies where noise positively associated with alertness and selectivity - the situation that having higher priority for the allocation of resources to multiple tasks [55].

The result of cognitive performance prediction obtained from the proposed CCSP model shows the importance of leveraging domain knowledge in model design, which has been overlooked in most of the previous work. Although black-box approaches such as DNN also achieve relatively good results, our simple but interpretable CCSP model obtains the best prediction performance which is superior to the others.

Cognitive performance prediction with ubiquitous sensing devices has many potential applications. Organization management can apply this technique to optimize overall productivity with workflow management. For instance, more work can be allocated to the worker inducing stress that improves its productivity. It can also reduce the workload of those workers experiencing mental fatigue and allow those workers to recover from overload. Some occupations are stressful in nature such as caregivers, nurses, and drivers. Cognitive performance prediction can be applied to prevent accidents caused by excessive stress and fatigue. Individuals such as students and employees can utilize the prediction result to improve productivity by scheduling working or study plans dynamically.

This work has paved the way for future research towards extending the application of stress detection to understanding human behavior. However, it still has several limitations and potentials for future work. First of all, this study focused on the dynamic of performance in a prolonged cognitive task. It is a micro view of the dynamic process, therefore, overlooked the effect of macro attributes such as the cognitive rhythm [12]. It is valuable to extend the CCSP model considering the cognitive rhythm as well as the energy recovery process in a longer period. Second, the work could be extended by increasing the scale of participants where new observations and conclusions can be drawn from a larger sample. Third, the ambient noise as an artificial stressor may not be the best choice in studying the effect on performance. Other stress induction methods that would not affect the evaluation of performance can be applied, where different effects on performance could be examined. Finally, this laboratory study as proof of concept provided important insight on predicting behavioral changes by measuring psychophysiological attributes. Future work could extend it to field study to capture cognitive dynamics in a more realistic environment.

Chapter 6

Conclusion

Ubiquitous monitoring of stress is an important problem with great impact, especially for smart health applications. In this thesis, we focused on the measure of repetitive activities, stress, and its impact on cognitive performance. For measuring repetitive activities, the key challenge is the variation incurred by individual differences and the diversity of the activities. Meanwhile, the proposed method should be efficient enough to operate on mobile devices. For the prediction of stress impacts, the key challenge is the limited number of annotated data, which makes it difficult to approximate a model to the underlying relationship. Especially most of the data-driven models are data-hungry.

To address the issue of diversity in human activities, we proposed to measure the repeating physical motion to capture repetitive daily activities that are related to health. Particularly, we divided the problem into two sub-problems. The first is to detect repetitive activity using wearable sensors. The data generated are typically time-varying measures. We hereby study a more general problem that detects repeating patterns in multivariate time series. We proposed an efficient and robust algorithm to detect whether and where repeating patterns occurred in the time series. Second, we aim to locate and recognize the pattern that is being repeated from the detected segments of the time series. It helps to count the number of repetitions of the activity, and be able to extract the pattern so as to classify the activity. The proposed approach can effectively detect and identify repeating patterns in multivariate time series. Beyond measuring the repetitive activity, we demonstrated that the method can also be applied in many other applications such as respiration monitoring using wireless signals.

The second issue related to the difficulty to obtain annotated data for stress impact prediction. We first provided a preliminary study to show that positive stress can be recognized using data collected from smartphones. On this basis, we confirmed stress symptoms have relations with an impact on cognitive performance. By leveraging domain knowledge in psychology and physiology, we proposed a computational model that can continuously predict cognitive performance by monitoring physiological signals as an estimation of stress.

As discussed in section 1.3, stress detection has been studied extensively in which different potential measures were investigated for both biomedical and behavioral symptoms. However, the computational way for stress impact monitoring is under-examined. It is crucial to measure and predict the impact of stress to provide more effective stress management. In the future, we will investigate more aspects of the stress impacts to improve the stress intervention method. Also, more attention will be paid to the discovery of behavioral symptoms of stress. This can potentially be achieved by applying knowledge from psychology, as well as to inspire new directions back to the psychology field.

In summary, we investigated the measurement of stress symptoms and the prediction of stress impacts. We identified the challenges and research issues for stress monitoring and proposed an effective framework to address the issues aforementioned.

The proposed algorithms for repeating pattern is a general approach that is also applicable for other problems with multivariate time series. The prediction of stress impact demonstrated the potential of better stress management using a computational approach and offering promising direction for future work.

Bibliography

[1] Apple watch series 5. https://www.apple.com/apple-watch-series-5/. Accessed Jul. 20, 2020.

[2] Fitbit official site. https://www.fitbit.com/. Accessed Jul. 20, 2020.

[3] Garmin international. https://www.garmin.com/. Accessed Jul. 20, 2020.

[4] Human behavior. https://en.wikipedia.org/wiki/Human_behavior/. Accessed Jul. 20, 2020.

[5] Human behavior: The complete pocket guide. https://imotions.com/blog/human-behavior/. Accessed Jul. 20, 2020.

[6] Polar h7 heart rate sensor. http://www.polar.com/hk-en/products/accessories/H7_heart_rate_sensor. Accessed Mar. 16, 2016.

[7] Rescuetime : Time management software for staying productive and happy in the modern workplace. http://www.rescuetime.com/. Accessed Jan. 30, 2016.

[8] Understanding the stress response. https://www.health.harvard.edu/staying-healthy/understanding-the-stress-response. Accessed Jul. 20, 2020.

[9] Workplace stress. http://www.stress.org/workplace-stress/. Accessed Jul. 20, 2020.

[10] Stress and health disparities report, 2016. http://www.apa.org/pi/health-disparities/resources/stress-report.aspx/. Accessed Jul. 20, 2020.

[11] Zahraa S Abdallah, Mohamed Medhat Gaber, Bala Srinivasan, and Shonali Krishnaswamy. Activity recognition with evolving data streams: A review. *ACM Computing Surveys (CSUR)*, 51(4):1–36, 2018.

[12] Saeed Abdullah, Elizabeth L Murnane, Mark Matthews, Matthew Kay, Julie A Kientz, Geri Gay, and Tanzeem Choudhury. Cognitive rhythms: unobtrusive and continuous sensing of alertness using a mobile phone. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 178–189, 2016.

[13] Phil Adams, Mashfiqui Rabbi, Tauhidur Rahman, Mark Matthews, Amy Voida, Geri Gay, Tanzeem Choudhury, and Stephen Voida. Towards personal stress informatics: comparing minimally invasive techniques for measuring daily stress in the wild. In *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, pages 72–79, 2014.

[14] American Psychological Association. Stress in america: Coping with change. Stress in America™ Survey, 2017.

[15] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *Proceedings of ESANN*, 2013.

[16] Donald S Baim, Wilson S Colucci, E Scott Monrad, Harton S Smith, Richard F Wright, Alyce Lanoue, Diane F Gauthier, Bernard J Ransil, William Grossman, and Eugene Braunwald. Survival of patients with severe congestive heart failure treated with oral milrinone. *Journal of the American College of Cardiology*, 7(3):661–670, 1986.

[17] Oresti Banos, Rafael Garcia, Juan A Holgado-Terriza, Miguel Damas, Hector Pomares, Ignacio Rojas, Alejandro Saez, and Claudia Villalonga. mhealthdroid: a novel framework for agile development of mobile health applications. In *International workshop on ambient assisted living*, pages 91–98. Springer, 2014.

[18] P. Barralon, N. Vuillerme, and N. Noury. Walk detection with a kinematic sensor: Frequency and wavelet comparison. In *Proceedings of IEEE EMBS*, 2006.

[19] Yoshua Bengio and Paolo Frasconi. Input-output hmms for sequence processing. *IEEE Transactions on Neural Networks*, 7(5):1231–1249, 1996.

[20] H. Benson and R.L. Allen. How much stress is too much? *Harvard Business Review*, 58(5):86–92, 1979.

[21] Nicholas Blouin, John Deaton, Erin Richard, and Paul Buza. Effects of stress on perceived performance of collegiate aviators. *Aviation Psychology and Applied Human Factors*, 4(1):40–49, 2014.

[22] Andrey Bogomolov, Bruno Lepri, Michela Ferron, Fabio Pianesi, and Alex Sandy Pentland. Daily stress recognition from mobile phone data, weather conditions and individual traits. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 477–486. ACM, 2014.

[23] Maarten AS Boksem and Mattie Tops. Mental fatigue: costs and benefits. *Brain research reviews*, 59(1):125–139, 2008.

[24] Agata Brajdic and Robert Harle. Walk detection and step counting on unconstrained smartphones. In *Proceedings of ACM Ubicomp*, 2013.

[25] A. J. Camm, M. Malik, J. T. Bigger, G. Breithardt, S. Cerutti, R. J. Cohen, P. Coumel, E. L. Fallen, H. L. Kennedy, R. E. Kleiger, F. Lombardi, A. Malliani, A. J. Moss, J. N. Rottman, G. Schmidt, P. J. Schwartz, and D. H. Singer. Heart rate variability: standards of measurement, physiological interpretation and clinical use. *Circulation*, 93(5):1043–1065, 1996.

[26] N.V. Chawala, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer. Smote: Synthetic minority over-sampleing technique. *J. Artif. Intell. Res*, 16:321–357, 2002.

[27] Kaixuan Chen, Lina Yao, Dalin Zhang, Bin Guo, and Zhiwen Yu. Multi-agent attentional activity recognition. pages 1344–1350, 08 2019.

[28] Tong Chen, Peter Yuen, Mark Richardson, Guangyuan Liu, and Zhishun She. Detection of psychological stress using a hyperspectral imaging technique. *IEEE Transactions on Affective Computing*, 5(4):391–405, 2014.

[29] George P Chrousos and Philip W Gold. The concepts of stress and stress system disorders: overview of physical and behavioral homeostasis. *Jama*, 267(9):1244–1252, 1992.

[30] Maria Cornacchia, Koray Ozcan, Yu Zheng, and Senem Velipasalar. A survey on activity detection and classification using wearable sensors. *IEEE Sensors Journal*, 17:386–403, 2016.

[31] Hugo Critchley and Yoko Nagai. Electrodermal activity (eda). In *Encyclopedia of Behavioral Medicine*, pages 666–669. Springer, 2013.

[32] Hoang Anh Dau and Eamonn Keogh. Matrix profile v: A generic technique to incorporate domain knowledge into motif discovery. In *Proceedings of ACM SIGKDD*, pages 125–134. ACM, 2017.

[33] E Ron De Kloet, Marian Joëls, and Florian Holsboer. Stress and the brain: from adaptation to disease. *Nature reviews. Neuroscience*, 6(6):463, 2005.

[34] Johanna M Doerr, Beate Ditzen, Jana Strahler, Alexandra Linnemann, Jannis Ziemek, Nadine Skoluda, Christiane A Hoppmann, and Urs M Nater. Reciprocal relationship between acute stress and acute fatigue in everyday life in a sample of university students. *Biological psychology*, 110:42–49, 2015.

[35] David Eilam, Rony Izhar, and Joel Mort. Threat detection: Behavioral practices in animals and humans. *Neuroscience & Biobehavioral Reviews*, 35(4):999–1006, 2011.

[36] Akane Sano et al. Recognizing academic performance, sleep quality, stress level, and mental health using personality traits, wearable sensors and mobile phones. In *Wearable and Implantable Body Sensor Networks (BSN), 2015 IEEE 12th International Conference on*, pages 1–6. IEEE, 2015.

[37] Raihana Ferdous, Venet Osmani, and Oscar Mayora. Smartphone app usage as a predictor of perceived stress levels at workplace. In *2015 9th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*, pages 225–228. IEEE, 2015.

[38] Yoren Gaffary, David Antonio Gómez Jáuregui, Jean-Claude Martin, and Mehdi Ammi. Gestural and postural reactions to stressful event: Design of a haptic stressful stimulus. In *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*, pages 988–992. IEEE, 2015.

[39] Hua Gao, Anil Yüce, and Jean-Philippe Thiran. Detecting emotional stress from facial expressions for driving safety. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 5961–5965. IEEE, 2014.

[40] Enrique Garcia-Ceja, Venet Osmani, and Oscar Mayora. Automatic stress detection in working environments from smartphones' accelerometer data: a first step. *IEEE journal of biomedical and health informatics*, 20(4):1053–1060, 2015.

[41] B. Gerald and L. Paul. Can smartphones detect stress-related changes in the behaviour of individuals? In *Pervasive Computing and Communications Workshops, 2012 IEEE International Conference on*, pages 423–426. IEEE, 2012.

[42] Shaghayegh Gharghabi, Chin-Chia Michael Yeh, Yifei Ding, Wei Ding, Paul Hibbing, Samuel LaMunion, Andrew Kaplan, Scott E Crouter, and Eamonn Keogh. Domain agnostic online semantic segmentation for multi-dimensional time series. *Data Mining and Knowledge Discovery*, 33(1):96–130, 2019.

[43] Giorgos Giannakakis, Dimitris Grigoriadis, Katerina Giannakaki, Olympia Simantiraki, Alexandros Roniotis, and Manolis Tsiknakis. Review on psychological stress detection using biosignals. *IEEE Transactions on Affective Computing*, 2019.

[44] Earl F. Glynn, Jie Chen, and Arcady R. Mushegian. Detecting periodic patterns in unevenly spaced gene expression time series using lomb–scargle periodograms. *Bioinformatics*, 22:310–316, 2006.

[45] Xiaonan Guo, Jian Liu, and Yingying Chen. Fitcoach: Virtual fitness coach empowered by wearable mobile devices. In *Proceedings of IEEE INFOCOM*, 2017.

[46] Darakhshan J Haleem, Saida Haider, Tahira Perveen, Muhammad Abdul Haleem, et al. Serum leptin and cortisol, related to acutely perceived academic examination stress and performance in female university students. *Applied psychophysiology and biofeedback*, 40(4):305–312, 2015.

[47] Nils Y Hammerla, Reuben Kirkham, Peter Andras, and Thomas Ploetz. On preserving statistical characteristics of accelerometry data using their empirical cumulative distribution. In *Proceedings of the 2013 international symposium on wearable computers*, pages 65–68, 2013.

[48] Peter A Hancock and HC Ganey. From the inverted-u to the extended-u: The evolution of a law of psychology. *Journal of Human Performance in Extreme Environments*, 7(1):3, 2003.

[49] Ralf Hansmann, Stella-Maria Hug, and Klaus Seeland. Restoration and stress relief through physical activities in forests and parks. *Urban forestry & urban greening*, 6(4):213–225, 2007.

[50] Tian Hao, Guoliang Xing, and Gang Zhou. Runbuddy: A smartphone system for running rhythm monitoring. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 133–144. ACM, 2015.

[51] Harish Haresamudram, David V Anderson, and Thomas Plötz. On the role of features in human activity recognition. In *Proceedings of the 23rd International Symposium on Wearable Computers*, pages 78–88, 2019.

[52] Jennifer A Healey and Rosalind W Picard. Detecting stress during real-world driving tasks using physiological sensors. *IEEE Transactions on intelligent transportation systems*, 6(2):156–166, 2005.

[53] Roselinde Henderson, Hannah Snyder, Tina Gupta, and Marie Banich. When does stress help or harm? the effects of stress controllability and subjective stress response on stroop performance. *Frontiers in Psychology*, 3:179–194, 2012.

[54] Javier Hernandez, Pablo Paredes, Asta Roseway, and Mary Czerwinski. Under pressure: sensing stress of computer users. In *Proceedings of the 2014 SIGCHI Conference on Human Factors in Computing Systems*, pages 51–60. ACM, 2014.

[55] Robert Hockey. *Stress and fatigue in human performance*, volume 3. John Wiley & Sons Inc, 1983.

[56] Robert Hockey. Compensatory control in the regulation of human performance under stress and high workload: A cognitive-energetical framework. *Biological psychology*, 45(1):73–93, 1997.

[57] Karen Hovsepian, Mustafa al'Absi, Emre Ertin, Thomas Kamarck, Motohiro Nakajima, and Santosh Kumar. cstress: towards a gold standard for continuous stress assessment in the mobile environment. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'15)*, pages 493–504. ACM, 2015.

[58] Bing Hu, Yanping Chen, and Eamonn Keogh. Time series classification under more realistic assumptions. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 578–586. SIAM, 2013.

[59] Qianyi Huang, Yan Mei, Wei Wang, and Qian Zhang. Battery-free sensing platform for wearable devices: The synergy between two feet. In *Proceedings of IEEE INFOCOM*. IEEE, 2016.

[60] Stella-Maria Hug, Terry Hartig, Ralf Hansmann, Klaus Seeland, and Rainer Hornung. Restorative qualities of indoor and outdoor exercise settings as predictors of exercise frequency. *Health & place*, 15(4):971–980, 2009.

[61] Julian Huxley. *On living in a revolution*. Chatto and windus, 1944.

[62] Y. İşler and M. Kuntalp. Combining classical hrv indices with wavelet entropy measures improves to performance in diagnosing congestive heart failure. *Computers in Biology and Medicine*, 37(10):1502–1510, 2007.

[63] Natasha Jaques, Sara Taylor, Asaph Azaria, Asma Ghandeharioun, Akane Sano, and Rosalind Picard. Predicting students' happiness from physiology, phone, mobility, and behavioral data. In *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 222–228. IEEE, 2015.

[64] Mohammed Waleed Kadous. Learning comprehensible descriptions of multivariate time series. In *In the Proc. of ICML*, volume 454, page 463, 1999.

[65] Eugenijus Kaniusas. *Biomedical signals and sensors I: Linking physiological phenomena and biosignals*. Springer Science & Business Media, 2012.

[66] Roman Kupriyanov and Renad Zhdanov. The eustress concept: problems and outlooks. *World Journal of Medical Sciences*, 11(2):179–185, 2014.

[67] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter*, 12:74–82, 2011.

[68] Yongjin Kwon, Kyuchang Kang, and Changseok Bae. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications*, 41(14):6067–6074, 2014.

[69] Martin Lang, Jan Krátkỳ, John H Shaver, Danijela Jerotijević, and Dimitris Xygalatas. Effects of anxiety on spontaneous ritualized behavior. *Current Biology*, 25(14):1892–1897, 2015.

[70] Richard S Lazarus. From psychological stress to the emotions: A history of changing outlooks. *Annual review of psychology*, 44(1):1–22, 1993.

[71] I-Min Lee and David M Buchner. The importance of walking to public health. *Medicine & Science in Sports & Exercise*, 40(7):S512–S518, 2008.

[72] Iulia Lefter, Gertjan J Burghouts, and Léon JM Rothkrantz. Recognizing stress using semantics and modulation of speech and gestures. *IEEE Transactions on Affective Computing*, 7(2):162–175, 2016.

[73] Chun-Tung Li, Jiannong Cao, and Tim MH Li. Eustress or distress: an empirical study of perceived stress in everyday college life. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 1209–1217. ACM, 2016.

[74] Chun-Tung LI, Jiannong Cao, Xue Liu, and Milos Stojmenovic. msimpad: Efficient and robust mining of successive similar patterns of multiple lengths in time series. *ACM Transactions on Computing for Healthcare*, 1, 2020.

[75] Hong Li, Gregory D Abowd, and Thomas Plötz. On specialized window lengths and detector based human activity recognition. In *Proceedings of the 2018 ACM international symposium on wearable computers*, pages 68–71, 2018.

[76] Harris R Lieberman. Cognitive methods for assessing mental energy. *Nutritional neuroscience*, 10(5-6):229–242, 2007.

[77] Huijie Lin, Jia Jia, Quan Guo, Yuanyuan Xue, Qi Li, Jie Huang, Lianhong Cai, and Ling Feng. User-level psychological stress detection from social media using deep neural network. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 507–516, 2014.

[78] Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing sax: A novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15:107–144, 2007.

[79] Michele Linardi, Yan Zhu, Themis Palpanas, and Eamonn Keogh. Matrix profile x: Valmod - scalable discovery of variable-length motifs in data series. *Proceedings of ACM SIGMOD*, 2018.

[80] Xuefeng Liu, Jiannong Cao, Shaojie Tang, Jiaqi Wen, and Peng Guo. Contactless respiration monitoring via off-the-shelf wifi devices. *IEEE Transactions on Mobile Computing*, 15(10):2466–2479, 2015.

[81] Hong Lu, Denise Frauendorfer, Mashfiqui Rabbi, Marianne Schmid Mast, Gokul T Chittaranjan, Andrew T Campbell, Daniel Gatica-Perez, and Tanzeem Choudhury. Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM conference on ubiquitous computing*, pages 351–360, 2012.

[82] Wei Lu, Fugui Fan, Jinghui Chu, Peiguang Jing, and Su Yuting. Wearable computing for internet of things: A discriminant approach for human activity recognition. *IEEE Internet of Things Journal*, 6(2):2749–2759, 2018.

[83] Gale M Lucas, Jonathan Gratch, Stefan Scherer, Jill Boberg, and Giota Stratou. Towards an affective interface for assessment of psychological distress. In *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*, pages 539–545. IEEE, 2015.

[84] Yongqiang Lyu, Xiaomin Luo, Jun Zhou, Chun Yu, Congcong Miao, Tong Wang, Yuanchun Shi, and Ken-ichi Kameyama. Measuring photoplethysmogram-based stress-induced vascular response index to assess cognitive load and stress. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 857–866. ACM, 2015.

[85] Sebastian Madgwick. An efficient orientation filter for inertial and inertial/magnetic sensor arrays. *Report x-io and University of Bristol (UK)*, 2010.

[86] Takuya Maekawa, Daisuke Nakai, Kazuya Ohara, and Yasuo Namioka. Toward practical factory activity recognition: Unsupervised understanding of repetitive assembly work in a factory. In *Proceedings of ACM Ubicomp*, 2016.

[87] Alberto Malliani, Massimo Pagani, Federico Lombardi, and Sergio Cerutti. Cardiovascular neural regulation explored in the frequency domain. *Circulation*, 84(2):482–492, 1991.

[88] Dimitris Manousos, Galateia Iatraki, Eirini Christinaki, Matthew Pediaditis, Franco Chiarugi, Manolis Tsiknakis, and Kostas Marias. Contactless detection of facial signs related to stress: A preliminary study. In *2014 4th International Conference on Wireless Mobile Communication and Healthcare-Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH)*, pages 335–338. IEEE, 2014.

[89] Samuele M Marcora, Walter Staiano, and Victoria Manning. Mental fatigue impairs physical performance in humans. *Journal of applied physiology*, 106(3):857–864, 2009.

[90] G. Mark, Y. Wang, and M. Niiya. Stress and multitasking in everyday college life: an empirical study of online activity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 41–50. ACM, 2014.

[91] Gloria Mark, Shamsi T Iqbal, Mary Czerwinski, and Paul Johns. Bored mondays and focused afternoons: the rhythm of attention and online activity in the workplace. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3025–3034. ACM, 2014.

[92] Gloria Mark, Shamsi T Iqbal, Mary Czerwinski, Paul Johns, Akane Sano, and Yuliya Lutchyn. Email duration, batching and self-interruption: Patterns of email use on productivity and stress. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1717–1728. ACM, 2016.

[93] Gloria Mark, Melissa Niiya, Stephanie Reich, et al. Sleep debt in student life: Online attention focus, facebook, and mood. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 5517–5528. ACM, 2016.

[94] Pierre-François Marteau. Time warp edit distance with stiffness adjustment for time series matching. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):306–318, 2008.

[95] Gerald Matthews. *Human performance: Cognition, stress, and individual differences*. Psychology Press, 2000.

[96] Daniel J McDuff, Javier Hernandez, Sarah Gontarek, and Rosalind W Picard. Cogcam: Contact-free measurement of cognitive stress during computer tasks with a digital camera. In *Proceedings of the 2016 SIGCHI Conference on Human Factors in Computing Systems*, pages 4000–4004. ACM, 2016.

[97] Bruce S McEwen and Eliot Stellar. Stress and the individual: mechanisms leading to disease. *Archives of internal medicine*, 153(18):2093–2101, 1993.

[98] N. Meiran, Z. Chorev, and A. Sapir. Component processes in task switching. *Cognitive psychology*, 41(3):211–253, 2000.

[99] Mahtab Mirmomeni, Yousef Kowsar, Lars Kulik, and James Bailey. An automated matrix profile for mining consecutive repeats in time series. In *Proceedings of PRICAI*, 2018.

[100] Kei Mizuno and Yasuyoshi Watanabe. Utility of an advanced trail making test as a neuropsychological tool for an objective evaluation of work efficiency during mental fatigue. In *Fatigue science for human health*, pages 47–54. Springer, 2008.

[101] Dan Morris, T Scott Saponas, Andrew Guillory, and Ilya Kelner. Recofit: using a wearable sensor to find, recognize, and count repetitive exercises. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3225–3234, 2014.

[102] Jeremy N Morris and Adrianne E Hardman. Walking to health. *Sports medicine*, 23(5):306–332, 1997.

[103] Abdullah Mueen and Eamonn Keogh. Extracting optimal performance from dynamic time warping. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2129–2130, 2016.

[104] Abdullah Mueen, Suman Nath, and Jie Liu. Fast approximate correlation for massive time-series data. In *Proceedings of ACM SIGMOD*, 2010.

[105] NPR, Robert Wood Johnson Foundation, and Harvard School of Public Health. The burden of stress in america, 2014.

[106] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9:62–66, 1979.

[107] Panos M. Pardalos and Nisha Desai. An Algorithm for Finding a Maximum Weighted Independent Set in an Arbitrary Graph. *International Journal of Computer Mathematics*, 38:163–175, 1991.

[108] Matthew Pediaditis, Giorgos Giannakakis, Franco Chiarugi, Dimitris Manousos, Anastasia Pampouchidou, Eirini Christinaki, Galateia Iatraki, Eleni Kazantzaki, Panagiotis G Simos, Kostas Marias, et al. Extraction of facial features as indicators of stress and anxiety. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3711–3714. IEEE, 2015.

[109] Valentina Perciavalle, Marta Blandini, Paola Fecarotta, Andrea Buscemi, Donatella Di Corrado, Luana Bertolo, Fulvia Fichera, and Marinella Coco. The role of deep breathing on stress. *Neurological Sciences*, 38(3):451–458, 2017.

[110] Xia Qingxin, Atsushi Wada, Joseph Korpela, Takuya Maekawa, and Yasuo Namioka. Unsupervised factory activity recognition with wearable sensors using process instruction information. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, volume 3, pages 1–23. ACM New York, NY, USA, 2019.

[111] Judith G Rabkin and Elmer L Struening. Life events, stress, and illness. *Science*, 194(4269):1013–1020, 1976.

[112] Nastaran Mohammadian Rad, Seyed Mostafa Kia, Calogero Zarbo, Twan van Laarhoven, Giuseppe Jurman, Paola Venuti, Elena Marchiori, and Cesare Furlanello. Deep learning for automatic stereotypical motor movement detection using wearable sensors in autism spectrum disorders. *Signal Processing*, 144:180–191, 2018.

[113] Zafar Rafii and Bryan Pardo. Repeating pattern extraction technique (repet): A simple method for music/voice separation. *IEEE transactions on audio, speech, and language processing*, 21(1):73–84, 2012.

[114] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of ACM MobiCom*, 2012.

[115] Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers*, pages 108–109. IEEE, 2012.

[116] Akane Sano and Rosalind W Picard. Stress recognition using wearable sensors and mobile phones. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 671–676. IEEE, 2013.

[117] Hillol Sarker, Matthew Tyburski, Md Mahbubur Rahman, Karen Hovsepian, Moushumi Sharmin, David H Epstein, Kenzie L Preston, C Debra Furr-Holden, Adam Milam, Inbal Nahum-Shani, et al. Finding significant stress episodes in a discontinuous time series of rapidly varying mobile sensor data. In *Proceedings of the the 34th Annual Conference on Human Factors in Computing Systems*, CHI '16, pages 4489–4501. ACM, 2016.

[118] Neil Schneiderman, Gail Ironson, and Scott D Siegel. Stress and health: psychological, behavioral, and biological determinants. *Annual Review of Clinical Psychology*, 1:607–628, 2005.

[119] H. Selye. The stress syndrome. *AJN The American Journal of Nursing*, 65(3):97–99, 1965.

[120] Hans Selye. A syndrome produced by diverse nocuous agents. *Nature*, 138(3479):32–32, 1936.

[121] Hans Selye. Stress and the general adaptation syndrome. *British medical journal*, 1(4667):1383, 1950.

[122] Hans Selye. Stress without distress. In *Psychopathology of human adaptation*, pages 137–146. Springer, 1976.

[123] Cornelia Setz, Bert Arnrich, Johannes Schumm, Roberto La Marca, Gerhard Tröster, and Ulrike Ehlert. Discriminating stress from cognitive load using a wearable eda device. *IEEE Transactions on information technology in biomedicine*, 14(2):410–417, 2010.

[124] N. Sharma and T. Gedeon. Objective measures, sensors and computational techniques for stress recognition and classification: A survey. *Computer methods and programs in biomedicine*, 108(3):1287–1301, 2012.

[125] Nandita Sharma, Abhinav Dhall, Tom Gedeon, and Roland Goecke. Modeling stress using thermal facial patterns: A spatio-temporal approach. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 387–392. IEEE, 2013.

[126] Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul J.M. Havinga. A survey of online activity recognition using mobile phones. *Sensors*, 15:2059–2085, 2015.

[127] Mihaela Sorostinean, François Ferland, and Adriana Tapus. Reliable stress measurement using face temperature variation with a thermal camera in human-robot interaction. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 14–19. IEEE, 2015.

[128] Sujesha Sudevalayam and Purushottam Kulkarni. Energy harvesting sensor nodes: Survey and implications. *IEEE Communications Surveys & Tutorials*, 13(3):443–461, 2010.

[129] David Sun, Pablo Paredes, and John Canny. Moustress: detecting stress from mouse motion. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 61–70, 2014.

[130] Feng-Tso Sun, Cynthia Kuo, Heng-Tze Cheng, Senaka Buthpitiya, Patricia Collins, and Martin Griss. Activity-aware mental stress detection using physiological sensors. In *International Conference on Mobile Computing, Applications, and Services*, pages 211–230. 2010.

[131] Samuel J Vine, Liis Uiga, Aureliu Lavric, Lee J Moore, Krasimira Tsaneva-Atanasova, and Mark R Wilson. Individual reactions to stress predict performance during a critical aviation incident. *Anxiety, Stress, and Coping*, 28(4):467–477, 2015.

[132] Michail Vlachos, Philip Yu, and Vittorio Castelli. On periodicity detection and structural periodic similarity. In *Proceedings of SIAM International Conference on Data Mining*, 2005.

[133] Michail Vlachos, Philip S. Yu, Vittorio Castelli, and Christopher Meek. Structural periodic measures for time-series data. *Data Mining and Knowledge Discovery*, 12:1–28, 2006.

[134] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, pages 3–14, 2014.

[135] Alan T Welford. Stress and performance. *Ergonomics*, 16(5):567–580, 1973.

[136] Jacqueline Wijsman, Bernard Grundlehner, Hao Liu, Julien Penders, and Hermie Hermens. Wearable physiological sensors reflect mental stress state

in office-like situations. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 600–605. IEEE, 2013.

[137] Lei Xie, Xu Dong, Wei Wang, and Dawei Huang. Meta-activity recognition: A wearable approach for logic cognition-based activity sensing. In *Proceedings of IEEE INFOCOM*, 2017.

[138] Jiong Yang, Wei Wang, and P.S. Yu. Mining asynchronous periodic patterns in time series data. *IEEE Transactions on Knowledge and Data Engineering*, 15:613–628, 2003.

[139] Kung-Jiuan Yang, Tzung-Pei Hong, Yuh-Min Chen, and Guo-Cheng Lan. Projection-based partial periodic pattern mining for event sequences. *Expert Systems with Applications*, 40:4232–4240, 2013.

[140] Yanni Yang and Jiannong Cao. Robust rfid-based respiration monitoring in dynamic environments. In *IEEE International Conference on Sensing, Communication and Networking (IEEE SECON)*, 2020.

[141] Lina Yao, Feiping Nie, Quan Z Sheng, Tao Gu, Xue Li, and Sen Wang. Learning from less for better: semi-supervised activity recognition via shared structure discovery. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 13–24, 2016.

[142] Chin-Chia Michael Yeh, Helga Van Herle, and Eamonn Keogh. Matrix profile iii: The matrix profile allows visualization of salient subsequences in massive time series. *Proceedings of IEEE ICDM*, 2016.

[143] Chin-Chia Michael Yeh, Nickolas Kavantzas, and Eamonn Keogh. Matrix profile vi: Meaningful multidimensional motif discovery. *Proceedings of IEEE ICDM*, 2017.

[144] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 1317–1322. Ieee, 2016.

[145] Robert M Yerkes and John D Dodson. The relation of strength of stimulus to rapidity of habit-formation. *Journal of comparative neurology*, 18(5):459–482, 1908.

[146] Xiao Yu, Qing Li, and Jin Liu. Scalable and parallel sequential pattern mining using spark. *World Wide Web: Internet and Web Information Systems*, 22:295–324, 2019.

[147] Quan Yuan, Jingbo Shang, Xin Cao, Chao Zhang, Xinhe Geng, and Jiawei Han. Detecting multiple periods and periodic patterns in event time sequences. In *Proceedings of ACM CIKM*, 2017.

[148] Jing Zhai and Armando Barreto. Stress detection in computer users based on digital signal processing of noninvasive physiological variables. In *2006 international conference of the IEEE engineering in medicine and biology society*, pages 1355–1358. IEEE, 2006.

[149] Yan Zhu, Chin-Chia Michael Yeh, Zachary Zimmerman, Kaveh Kamgar, and Eamonn Keogh. Matrix profile xi: Scrimp++: Time series motif discovery at interactive speeds. In *Proceedings of IEEE ICDM*, 2018.